

Title: SipSplit - the SCORE part extractor
Date: (first issue 1992), last updated May 30, 2005
Document version: 6.0
Program conformity: 5.12

Table of contents

1. Introduction
2. Definition of terms
3. Input
4. Output
5. Limitations
6. Warranty and intellectual property
7. Marketing
8. Installation
9. Description of the keyword commands in the profile file SIPSPLIT.PRO

1. Introduction

From personal experience and judging from the reports on the net, PAGE has a number of persistent instabilities in the part extraction feature. An extraction program is available called SipSplit which offers superior features for part extraction.

SipSplit is one of the many utilities in the SIP family.

SipSplit claims to replace PAGE as a part extractor, and the following description will make clear in which situations the use of this utility is preferred or justified. The page layout capabilities of PAGE are excellent, and PAGE will still be needed for page layout.

- PAGE makes many errors when extracting multivoice parts. It has no way of properly distinguishing rests in the upper or lower voice. In fact it interprets two vertical rests as the lower one. SipSplit does not drop items. Note that PAGE also removes the instrument id.
- SipSplit can extract the upper or lower voice in a multivoice score. Use of the SEP command in SCORE is not considered a comfortable or reliable working method and does not properly process vertical rests and unstemmed notes.
- PAGE makes errors like duplication of extracted material, or dropping bars without diagnostic warnings. PAGE will sometimes change a tie with p8=-2 to p8=0 and then alter p3 and p6 to reproduce the endpoint offsets of p8=-2. SipSplit does not suffer from this instability.
- PAGE uses instrument id's to extract parts. SipSplit has the option to either use the instrument id, or to look at a specific staff in each system.
- To define input files in PAGE is not very well explained in the reference manual, and the method is not user-friendly. Defining input files for extraction in SipSplit is more relaxed than in PAGE. Select files by the Windows common dialog box. The input file suffixes need not be MUS, nor do input files need to have consecutive names.
- When partial input is requested, PAGE uses bar numbers which is usually a trial and error method. SipSplit counts the number of systems to skip and/or consider. In addition, SipSplit can consider the marks set in Par12 of barline items (PAGE does not seem to recognize these). This effectively means that you can extract from any bar to any other in one series of files.
- Texts (and numbers, and symbols) to be passed to other parts (Par6=1xx or 3xx) or suppressed (Par6=2xx) work as in PAGE. SipSplit has added features. It can distinguish top, middle, and bottom text. Also, text of any given type of font can be excluded from being passed.
- Passing cues is more elaborate in SipSplit than in PAGE. The clef of the cue is taken along, and the original clef is reset in the extracted part. Also the alphabetic instrument identification is automatically generated above the cue, while the size of notes, beams, rests, and thickness of slurs can be set to a user selected size.
- A PAGE error was reported on an invisible staff. SipSplit handles such a staff as any other.
- Reporting is copied to a file which can be inspected offline. In abnormal situations, extensive diagnostic messages clarify the cause in plain text.
- Voice separation recognizes the 'à 2' and likewise performer instruction and acts accordingly.
- Both PAGE and SipSplit work with invisible information from MUS files (instrument id's, cue information, barline marks, text and symbol pass info). This invisible information is summarized as part of the cross-reference program SipXref.

- SipSplit, when instructed to recognize French scoring will generate a rests staff for the instrument which was missing in the system.
- PAGE creates version 1 or version 2 formats when its input does not have the standard file trailer. SipSplit always generates files in current SCORE format (version 3).
- SipSplit can decompose a file more intelligently than PAGE.

The philosophy between PAGE and SipSplit from a control point of view is quite different. PAGE is interactive, meaning that the user is prompted for all options. Even when using PAGE repetitively, the same questions must be answered over and over again. Strictly speaking, PAGE is a mixture of interactive and batch processing, since extraction also needs a control file. SipSplit is a pure batch process. The DOS user prepares his options in a text file called a profile, and only changes it when the circumstances change, ie. for another instrument or project. In Windows, this text file is generated automatically with the selected options and the file is transparent for the user.

The number of SipSplit keywords described here may be impressive, however most users will never need to set most of them, since default values are assumed. At the end of this document there is a “survival guide” which gives the absolute minimum to successfully use SipSplit. It must be accepted that if SipSplit offers more extraction features than PAGE, there are also more elements over which the user has control.

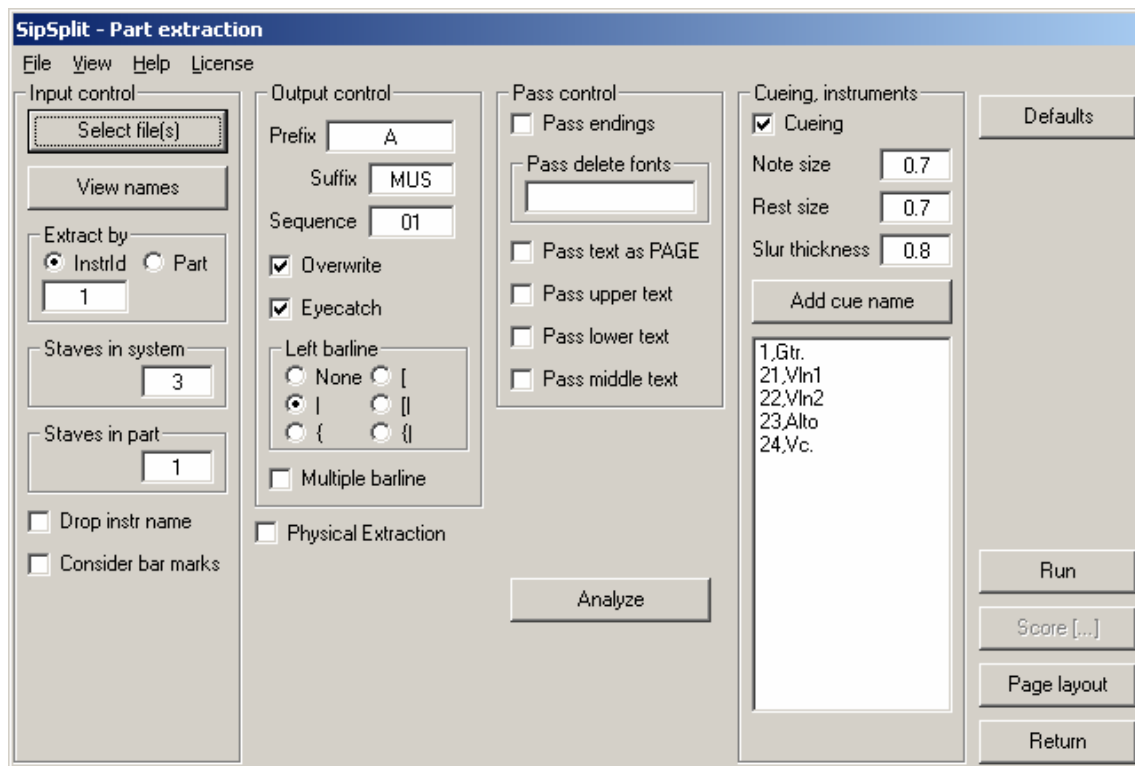
Additional features for Windows users are two programs which are included with SipSplit:

- whole rest reduction
- decomposition in a fixed number of bars per output system (SipDecom)

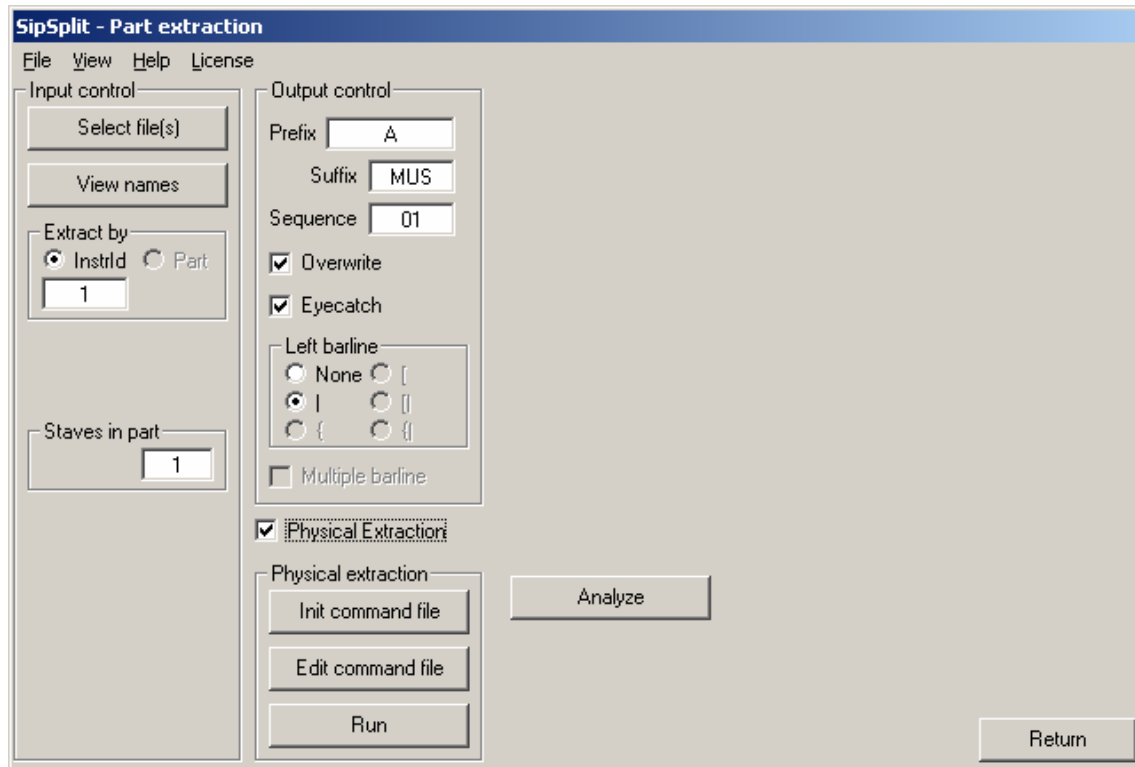
Whole rest reduction replaces consecutive whole rest bars by one long rest bar with a number. Optionally this can be an American style rest bar. There is a difference with PAGE. While PAGE acts on rest bars which have any item (such as a text) as an interruption of consecutive rests, SipSplit will only act on

- change of clef
- change of key signature
- change of time signature
- barline other than the normal single line

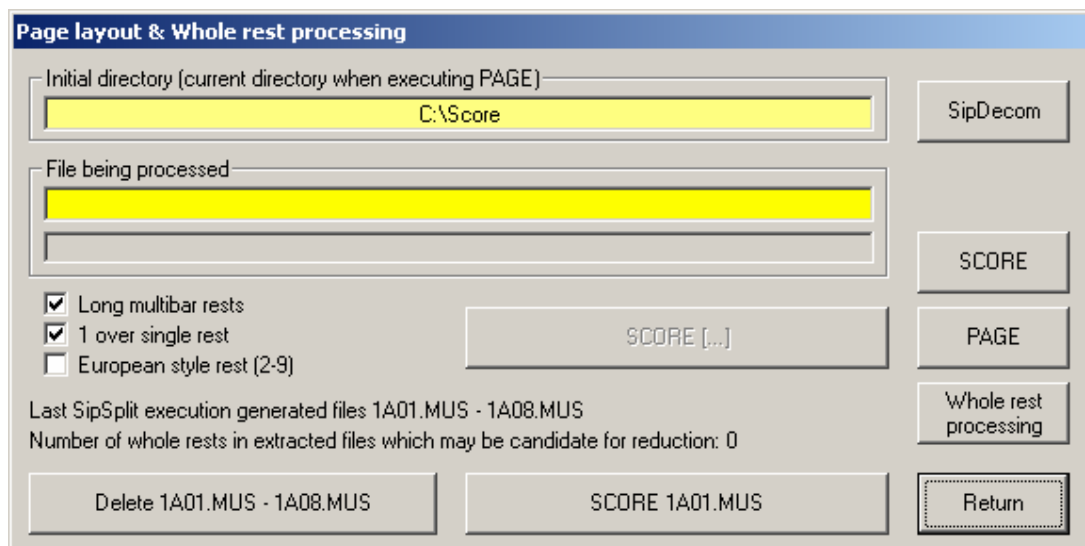
SipDecom is separately described in the document “The graphical user interface for SIP”. The panel is self-explanatory while there is an online help file description.



When the 'physical separation' is selected, the panel looks like this:



Once extraction has run, the user can select the 'Page layout' function shown here:



2. Definition of terms

It is assumed that each page of music is a file. When there are more files to define one actual page, this is of no influence to SipSplit. Files created by the SCORE save command ('sa', 'sm'), or output by PAGE or SipSplit for that matter, will work properly.

3. Input

The process of splitting files into parts is controlled by keyword commands. These keyword commands define input, output, and options. All keyword commands have defaults, only input file(s) must be provided in the Names file. Windows users select parameters by intuitive buttons and input fields. Files are defined by the select files function in the Control Center.

The format of the profile file SIPSPLIT.PRO is described in chapter 9. In Windows, this file is automatically created with the options selected.

4. Output

Output files are in the MUS format and can immediately be input by the 'g' SCORE command and read by PAGE. Each resulting file is generated as a single number 1 staff, or if a multistaff instrument is (or two or more adjacent instruments at the same time are) extracted, 1 and up. The output filename structure is nAx.MUS where n is the instrument id or the number of the part (one to four digits), x is a sequence number from 01 to 99 or 001 to 999. "A" is a string of from one to five alphabetic characters defined by the user. MUS is the default suffix, it may be set by the user.

The resulting files each receive an eye catcher. This is the name of the file in text, and allows the user to identify the extracted part (and voice) once it has been processed in a later stage. In the final stages of editing this eye catcher is to be deleted by hand. The eye catcher option may be turned off. The eye catcher serves an additional purpose. The order of the items in the generated MUS files is staff, barlines, other items in the order they were encountered, eye catcher, text items passed from other staves. Hence the user can easily distinguish the original text from the passed text.

The license number in MUS files output by SipSplit is by default set to 3109999. This may optionally be set to any value by the user. During processing, SipSplit creates a report file called SIPSPLIT.PRN which is to be consulted after each execution for diagnostics. It displays automatically at execution termination in Windows.

Diagnostic messages are output for any situation which is unacceptable for further processing, or to warn the user about questionable input. These messages contain sufficient information to locate the cause. There are information messages, warnings, errors, and abnormal terminations.

5. Limitations

maximum	number of	per
250	output staves (files)	program execution
99	instrument names	program execution
99	parts	system
30	barlines	staff
99	staves	page (input file)
500	stemmed notes	page (input file)
500	unstemmed notes	page (input file)
300	cues	page (input file)

Overflow of any of these limits is properly diagnosed.

When separating voices, SipSplit cannot decide to which voice belong grace notes. This is because with few exceptions, grace note stems are always up. Therefore grace notes are always copied. Sensitive to voice separation are notes, rests, beams, and slurs. Use the FLN command of SCORE to normalize stem directions. Note that the FL commands make no exception for grace notes. Flip command behave erratically on displaced notes under or above beamed groups. Text which is passed from one staff to another may not contain quote marks or apostrophes. If these occur in a pass text they will be replaced by spaces and the user receives a warning message. Holes in staff numbering between systems, as well as within systems are allowed when extracting by instrument id. However, when addressing staves by Part and StavesInSystem, holes in staff numbering are not acceptable. French scoring is acceptable.

6. Warranty and intellectual property

The program comes on an "as is" basis. Do not make copies for use by others. Remember that this is a small world. Error reporting and feature suggestions are welcome but no more than that. If you return the original diskette, there is a realistic possibility that you automatically get an upgrade or fix at no charge. Self-explanatory messages report abnormal situations. If a message cannot be understood it is probably an indication

that the input file is not in the MUS format. Any other situation may be reported to the author. Provide a copy of your input file if you desire a quality improvement of this product. Correspondence is via the net.

7. Ordering, delivery, pricing and payment

SipSplit costs US\$ 50.-. For further details contact the author.

8. Installation

Consult the file README on the delivery medium for installation details, installation verification, features not documented here, and sample files.

9. Description of the keyword commands in the profile file SIPSPLIT.PRO

Keyword commands control the functioning of SipSplit. If no keyword command file is present, or keyword commands are omitted from the file, defaults are assumed. Remember that Windows users need not know about this file since it is generated automatically.

Keyword commands start in column 1 of this text file. Records with "*" in column 1 are considered comment. All characters in keyword commands are interpreted as uppercase except where noted.

The keyword commands may not contain anything beyond the last significant character. Use the profile which comes on the installation diskette and adapt it to your liking. Note that as of release 5.8, a default profile is generated when SipSplit is executed for the first time.

9.1 Basic keyword commands

ExtractBy=InstrId or Part

This parameter decides whether the extraction is by instrument id or part. Either one is used, the other is ignored.

InstrumentId=1

Defines the instrument to be extracted. This value corresponds with Par9 of staff (Code8) items. Any number from 0 to 9999 may be provided. (SipSplit can extract instrument 0 while PAGE cannot). The default instrument id is 1.

Part=1

This parameter defines the staff number in a system to be selected. Note that parts are counted top to bottom (unlike SCORE staff numbering). The default part is 1, the top staff of the system.

StavesInSystem=3

The System parameter defines the number of staves in a system. The default is 3.

StavesInPart=1

It is possible to extract multistaff music such as for harp, piano, organ etc as well as vertically adjacent instruments in one execution. By default, one staff is extracted. When multistaff music is to be extracted, define the number of staves for that instrument with this parameter. The InstrumentId or Part keyword command refer to the bottom staff of the multistaff selection. The value is set to 1 when voice separation is in effect.

ConsiderBarMarks=No

By default, Par12 in barline (Code14) items is ignored. If set to Yes, this parameter will ignore all information until it finds the first barline item with the mark set. It will also ignore all information following the second barline item with the mark set. Note that marks are not considered in skipped systems (see the Skip parameter in chapter 9.4 (Input filename selection)). Note that to extract a small portion for a thematic index, there is the program SipThema. To extract a music sample for an article the program SipExtra is better suited for this.

DropInstrName=Yes

Since PAGE has the option to delete items left of the staff, this keyword command was added. It is however more specific for a text left of a staff and must be from vertical positions zero to twelve to qualify. If someone can prove that an equivalent of the PAGE option is needed, one will be considered.

9.2 Passing information from other staves

PassEndings=Yes

First and second endings are by default passed in the same way as PAGE. PAGE looks at Par5 which can have a 1xx or 2xx indicator. To suppress the passing of endings set this parameter to No.

PassDeleteFont=

Provide the number of the font which is not to be passed to another staff. More than one font may be specified if separated by commas. Fonts are defined by numeric values from 0 thru 99. The number of digits is not significant. An empty string terminated by a comma is interpreted as font 00. For example, *PassDeleteFont=4,25* removes all texts with fonts *_04*, *__04*, *_25*, and *__25*.

PassAsPage=Yes

SipSplit reacts to texts (Code16), numbers (Code10) and symbols (Code9, Code11) as PAGE when Par6 has the 1xx, 2xx, or 3xx value set. If set to No, the texts will be passed based on their vertical position (Par4) according to the following three parameters.

PassUpperText=Yes (Considered if PassAsPage=No)

To pass text from the top part of a system to the other parts in the same system (titles, movements, ...). The default is Yes. Text is considered upper-text if on vertical position 12 or higher.

PassLowerText=Yes (Considered if PassAsPage=No)

To pass text from the bottom part of a system to the upper parts in the same system (da Capo al Fine, ...). The default is Yes. Text is considered lower-text if it is on vertical position zero or lower.

PassMiddleText=No (Considered if PassAsPage=No)

Text in between vertical positions zero and 12 is by default not passed in an extraction. By setting this parameter to Yes, the default can be overridden.

9.3 Voice separation (feature suppressed since version 5.12)

The separation of voices is much more elaborate than in PAGE. Sensitive to voice separation are notes, rests, beams, and slurs. When *SingleVoice=No*, notes are considered belonging to the upper voice if they have upstems or if they are unstemmed and above an upstem note. Notes for the lower voice are processed likewise. Unstemmed notes which are not vertically connected to stemmed notes are extracted with a run time message. For orchestral scores to separate the upper and lower voices is controlled by the parameter *SingleVoice=Yes* which is the default. Separation is semi intelligent.

Voice=No

By default, SipSplit will not separate voices. By setting this parameter to Upper or Lower, it will extract only the indicated voice. Stems will not be flipped. Grace notes are always copied. When separation is in effect, the *StavesInPart* value is set to 1.

NormalizeRest=Yes

When voice separation is in effect, rests will by default be moved to the vertical center of the staff (*Par4=0*). To maintain the original vertical position, set this to No.

SingleVoice=Yes

When set (it is the default), voice extraction assumes there are two single voice instruments notated on one staff. It will separate the voices while passing and adjusting slurs and ties. Unstemmed notes are assumed to be significant, so a stemmed note with an attached unstemmed one is considered two voices, the lower and the upper. Unstemmed notes receive a stem and copy the notemark from the stemmed partner. When there are upstem and downstem notes, the stem direction decides voice extraction. It is important to note that slurs and beams can be connected to notes if they are horizontally closer than 1.99 steps, and that ties can be connected to notes if they are vertically closer than 2 steps. Slurs and ties are flipped if necessary. Adjustment is comparable to the SCORE "cv" command. Slurs receive a *Par8* of -1 while ties are always made horizontal.

SipSplit recognizes voice separation commands a2, 1., 2., Div. and Unis. while any string can be defined for these commands. The last command on a staff is carried over to the next page. Commands are considered only if *SingleVoice=Yes*. The command must be horizontally atr or left of the section to be in effect. Sizes are not considered.

Commands can be given as Code9 items (*P5=3XXX*) or as strings. SCORE type notation for accented characters is supported, so ">>a 2" would mean "à 2". The definition must include the font prefix.

a2=

SipSplit will find out which is the upper voice when Voice=Upper and which is the lower voice when Voice=Lower.

Example: a2=_00>>a2

a2(1)=

To recognize the upper voice override, define the string for instrument 1.

Example: a2(1)=_001.

a2(2)=

To recognize the upper voice override, define the string for instrument 2.

Example: a2(2)=_002!0

Div=

The string to recognize divisi is defined by this command.

Example: Div=_00div.

Unis=

Unisono is recognized by the string defined with this command.

Example: Unis=3053 (where CODE9.DAT has defined 3053 as __00unis.)

9.4 Input filename selection

Input filenames are read from the Names file. Files are processed in that order.

Normally, all systems on a page are considered. To skip systems on top of a page, use the Skip keyword command. To consider a restricted number of systems, use the Count keyword command. For the Skip and Count to work properly, staves must be consecutively numbered, bottom to top. Multiple occurrences of the same staff number leads to unpredictable results.

The Skip keyword command refers to the first page, the Count keyword command to the last.

Skip=0 (Only considered when ExtractBy=Part)

By default no systems on the first page are skipped.

Count=99 (Only considered when ExtractBy=Part)

By default all systems on the last page are considered. If zero, this is considered as 99.

Assume that piece B is to be selected from 3 pages of 4 systems:

page10	page11	page12
A	B	B
A	B	B
B	B	B
B	B	C

(input files are page10, page11, and page12)

Skip=2

Count=3

Likewise, a middle portion from a page can be addressed:

page20

A
B
B
C

(input file is page20)

Skip=1

Count=2

9.5 Passing cues

SCORE is used to set cue instrument parameters in Par13 of whole rests, where this Par13 is the instrument identifier which refers to the staff in the same system which has that identifier as Par9 in the staff (Code8) item. SipSplit offers some additional features as documented by the parameters. Unlike PAGE does, the cue is not flipped up, since this is not always desirable. The cue receives an alphabetic instrument name (when defined) and if the clef of the cued instrument is different, a begin and end cue size clef will be generated. A full rest is generated on the bottom (at -8). Note that when the instrument name starts with GU or GT (Guitar, Gtr.), the rest is put at -20. (See the CueName parameter).

Cueing=No

If cueing is desired, this parameter must be set to Yes.

CueName=

By default, if an alphabetic instrument name was encountered left of a leftmost barline, it will be attached to the instrument number it finds in that staff. The name remains attached across pages. Names starting and ending with a digit are ignored. When a cue is passed to a staff, the instrument name is generated above the first of a series of cue bars. With this parameter, the user can override the name by an appropriate instrument name such as an abbreviation. Example:

CueName=21,Vln1.

where 21 is the instrument id for the first violin, and "Vln1." the text to be generated above a cue by that instrument. Blanks in a name need to be coded as the percent sign (%), while the comma (,) and the equal sign (=) are invalid characters. Upper and lower case will be considered. There can be from 1 to 99 definitions, in any order. A font may be specified as part of the instrument name. If omitted _00 is assumed.

CueNoteSize=0

By default notes and beams are made cue size by setting the 1xx value of Par4. When a non-zero value is specified, that becomes the note size (Par15) or beam size (Par18).

Note that when the 1xx indicator was set, the note's parameters 14 and 15 are zeroed. This is necessary in those cases where Par15 was a non-zero value.

CueRestSize=0

By default rests are made cue size by setting the 1xx value of Par4. When a non-zero value is specified, that becomes the note size (Par15).

CueSlurThickness=0

By default, the slur and tie thickness is standard, but by setting a non-zero value, that value becomes the thickness (Par13).

Use SCORE editing for cue flipping (FLx), adjusting clefs to before barlines (LJ), center whole rests (CWn), and normalize cue stem lengths (ADn).

9.6 Output filename keyword commands

The structure of the output filename is nAx.MUS and the generated text item which serves as an eye catcher is nAx. In this name, n represents 1 to 4 digits, and it is the instrument id or the Part parameter value, whichever was selected. The alpha part of the name can be from 1 to 5 letters. This is controlled by parameter Prefix. The x represents a sequence number. By default this is two digits starting with 01, but if it is expected that more than 99 staves are extracted, one would rather have this starting at 001. The parameter which controls the length of the sequence number is Sequence. Conflicts may arise from erroneous settings (eg. 12HAYDN123.MUS, filename length error). These conflicts are properly diagnosed and if necessary the process is aborted. The filename may not be longer than 8 characters.

EyeCatch=Yes

The eye catcher is the filename without the .MUS extension. Generation may be suppressed by setting this option to No.

When voice separation is in effect, '(lower)' or '(upper)' is appended to the eye catcher.

Licence=3109999

The output .MUS file by default receives a dummy license number. The user may wish to have his own number here. Note that PAGE output has no license number (See the VER command of SCORE).

Prefix=A

This keyword command defines the alphabetic part of the output filename(s). There must be from one to five alphabetic characters. The default is 'A'.

Sequence=01

With Sequence, the length of the 'x' part in the filename is defined. Also it specifies the first sequence number. 001 would mean a length of three for the sequence number, 30 would mean a two-digit number starting at thirty. The default is 01.

MultipleBarLine=No

When the StavesInPart keyword command indicates a value higher than one, the barlines in the output will go across the staves like piano notation. This is a vertical notion. If each line is to be individually divided by barlines (stacking), set this option to No.

LeftBarLine=No

The leftmost barline in outputfiles is by default dropped. Various options can be given to force leftmost barlines: Line, Brace, Accolade, BraceLine, and AccoladeLine. In case MultiBarLine=No, each of these options will generate a single bar line, There cannot be a brace or accolade on a single staff system.

Overwrite=Yes

If set to No, and an output file exists, the user will be prompted for overwrite or termination. By default, an existing output file will be overwritten.

10. Environment

SipSplit runs on minimal DOS (640kb), Windows 3.1x, Windows98, 95 and NT systems.

11. Survival guide

Experience with PAGE for extraction helps but is not mandatory. Start by understanding the beginning of chapter 9 about the profile text file. Then check the following points. Primarily this is for DOS users, while Windows users have the intuitive panel settings.

Issue:	Study parameter:
How are input files specified?	InputFiles
You have the instrument parameter (Par9) set in each staff	InstrumentId
You do not have the instrument parameter set in each staff	Part/StavesInSystem
You want to do partial extraction by skipping complete systems from the input	Skip/Count
You want to do partial extraction, that is beginning and ending somewhere in the horizontal middle of systems	ConsiderBarMarks
You want to extract a part with more than one staff	StavesInPart

This will get you going. Other keyword commands will become familiar when you have specific needs.

Version 4.0 updates

This update is the Beta test qualification version. Compared to version 3.0 only the performance in time has been improved.

Version 5.0 update and acknowledgements

Ideas coming from users and members in the SCORE forum have been implemented and some quality improvements were made. Many changes were made to this user guide. Enhancement requests are taken seriously. We wish to thank the following people:

Philip Griffin for suggestions on user friendliness. The StavesInSystem parameter was suggested in favor of the misleading System parameter. The latter is still valid for upward compatibility.

Paul Sarcich for suggesting the OutputSuffix parameter.

Philip Carlsen for his ideas on cue element sizes. This resulted in the implementation of the parameters CueNoteSize, CueRestSize, and CueSlurThickness.

Peter Simcich suggested to make voice separation for single voice instruments more intelligent. This was a major rewrite for version v5.2. His request that the Instrument Id value can be greater than 99 was honored in v5.3. It was extended to 9999.

Tom Widlar suggested that SipSplit react on the 'à 2' voice separation instruction. Implemented in v5.3.

Sample SIPSPLIT.PRO (for DOS users)

```
ExtractBy=InstrId
InstrumentId=1
Sequence=01
StavesInSystem=7
*Part=1
OutputSuffix=MUS
StavesInPart=1
ConsiderBarMarks=No
MultipleBarLine=Yes
MultipleBarRest=No
DropInstrName=Yes
PassAsPage=Yes
*PassLowerText=Yes

*PassUpperText=Yes
*PassMiddleText=Yes
PassEndings=Yes
*
Skip=0
Count=99
*
Overwrite=yes
*LeftBarLine=BraceLine
EyeCatch=yes
Cueing=Yes
CueName=1,Cb.
CueName=2,Vc.
CueName=3,Va.
CueName=4,Vn.%II
CueName=5,Vn.%I
CueName=6,Gtr.
CueName=7,Perc.
CueNoteSize=0.75
CueRestSize=0.6
CueSlurThickness=0.6
```

About v5.1

One error message was modified to a warning. When a file did not contain an instrument to be extracted, SipSplit would abort. This was modified so that a warning is issued, but SipSplit continues with the next file. Also note that file selection is now through a common dialog box (Windows).

About v5.2

A new parameter was added called ExtractBy. Values can be InstrId or Part. The reason for this change was that the instrument number was reset to 1 at each execution, which turned out not to be handy. The parameter decides whether the extraction is by instrument id or part. Either one is used, the other is ignored.

Another new parameter is SingleVoice=Yes. Voice separation was completely rewritten for v5.2. Some warnings with little practical use have been suppressed.

About v5.3

A few changes were made to increase stability. Also note that the Instrument Id can now be up to 9999. The new feature 'à 2' was added to the voice separation capabilities.

About 5.7

Support for French scoring was introduced (user request). Many improvements were made tot the voice separation.

About 5.8

Includes whole rest compression. Consecutive rests are combined to one with the correct number over it. There is a difference with PAGE: SipSplit will only interrupt compression on a barline other than the standard or when the time, clef or key changes. PAGE interrupts on any other item it finds in a bar.

Rest compression is a separate run. It gives the user the opportunity to make changes to the extracted parts before doing a new layout.

Cueing did not work together with voice separation. This version supports the combination.

An internal table, storing cue information, was increased in size.

About 5.9 – 5.11

These are stability releases. It fixes problems in barline generation. The Multibar option is now functioning properly. French scoring improvements.

About 5.12

The voice separation was suppressed as the quality turned out to be too low for the average user.

It will be revised in version 6 which will be Windows rather than DOS based.

The physical separation feature was added.

The 'Analyze feature' was added to provide a list of instrument number in the selected files.

[end]