

**NAME**

**autos**p – preprocessor to generate note-spacing commands for MusiXTeX scores

**SYNOPSIS**

**autos**p [-v | --version | -h | --help]

**autos**p [-d | --dotted] [-l | --log] *infile* [.aspc | .tex ] [*outfile* [.tex]]

This program makes it easier to create MusiXTeX scores by converting (non-standard) commands of the form `\notes ... \en` into one or more conventional note-spacing commands (`\notes \Notes \NOTes ...`), determined by the actual note values, with `\sk` spacing commands inserted as necessary. The coding for an entire measure can be entered one part at a time, without concern for note-spacing changes within the part or spacing requirements of other parts.

For example, if applied to

```
\notes\qa J\qa K&\ca l\qa m\ca n\en
```

**autos**p generates

```
\Notes\qa J\sk\qa K\sk&\ca l\qa m\sk\ca n\en
```

Typically, an `\notes` command is expanded into several conventional note-spacing commands.

If the *infile* argument has .tex extension, **autos**p will process conventional note-spacing commands (but not `\vnotes` or `\znotes`) similarly. This is intended to correct spacing in an extracted single-instrument part, but may not produce satisfactory output when applied to a conventional multi-instrument MusiXTeX score.

If the *infile* argument has neither .aspc nor .tex extension, input is taken from *infile*.aspc if that file exists, or from *infile*.tex otherwise.

For `\notes` commands (and, for a .tex file, all conventional note-spacing commands), line breaks and spaces may precede note segments, allowing more flexible source formatting; the line breaks and spaces will be elided from the output.

For example,

```
\notes
\ibl0K0\qb0K\nbb10\qb0{.K}\tbb10\tbb10\tql0L&
\ibb11m{-2}\qb1{.m}\tbb11\tbb11\qb11\tql1k\en
```

is acceptable and generates

```
\notes\ibl0K0\qb0K&\ibb11m{-2}\qb1{.m}\en
\nnotes\sk&\tbb11\tbb11\qb11\en
\notes\nbb10\qb0{.K}&\tql1k\en
\nnotes\tbb10\tbb10\tql0L&\sk\en
```

If no *outfile* argument is provided, output goes to *infile*.tex if the *infile* name has extension .aspc, or to standard output otherwise.

If the **-l** ( **--log** ) option is used, a log *infile.alog* is generated.

If the **-d** ( **--dotted** ) option is used, *dotted* beam notes of the form `\qb{n}{.p}` are *not* given extra space; it is assumed that the subsequent note will be shifted by a `\roff`-like command or a spacing command such as `\qsk` or `\hqsk`. Commands of the form `\qlp{p}`, `\qlpp{p}`, ..., `\qpb{n}{p}` and `\qppb{n}{p}` are always spaced as indicated.

If there is a single staff, consecutive whole-bar rest bars are merged into a multi-bar rest. Bar-centered rests can be coded using the standard `\def\atnextbar` notation but also the non-standard command `\Cpause` in a note segment generates a bar-centered rest.

Spacing commands `\sk` and `\hsk` (but not `\qsk`, `\hqsk` or `\qqsk`) in the source are discarded. A note segment can be completely empty, but if a note segment should start with or contain a "space," the note-value of that space must be made explicit with a command of the form `\ha{*}`, `\qa{*}`, `\ca{*}`, etc.

All other conventional MusiXTeX commands are output exactly as given in the input.

## OPERATION

**autosp** determines the spacing for ordinary notes from the note commands themselves; for example,

+ `\qa`, `\qu`, `\ql`, `\qp` result in `\NOtes`;

+ `\ca`, `\cu`, `\cl`, `\ds` result in `\Notes`;

and so on.

The spacing for *beamed* notes is determined by the beam multiplicity, so that `\ib...` results in `\Notes`, `\ibb...` results in `\notes`, etc.

Collective coding of note sequences (including accidentals and dots) is handled by expanding the sequence into a sequence of individual note commands.

## LIMITATIONS

**autosp** assumes that `&` and `|` (rather than `\nextinstrument` and `\nextstaff`) are used to separate instruments and staves.

Appoggiaturas and grace notes are recognized by the use of `\tinynotesize`; note-spacing of `1.45\elemskip` is used. If this isn't suitable and can't be corrected with a small skip, a `\vnotes` command with any desired spacing can be used.

**autosp** supports *x-tuplets* introduced using `\xtuplet{x}` and *triplets* introduced using any of the following commands (regardless of any re-definition of `\txt` or `\tuplettxt`):

```
\triolet
\uptrio
\downtrio
\uptuplet
\downtuplet
```

However, MusiXTeX notation does not specify the *intended* duration of an x-tuplet. **autosp** assumes that an x-tuplet is to be played in  $(x-1)/x$  of the apparent x-tuplet duration. So, for example, a triplet in eighths is assumed to be played in the time of one quarter note. If this assumption isn't valid, the x-tuplet must be coded explicitly using a `\vnotes` command; see the first measure of `barsant2.aspc` for an example of a non-

standard x-tuplet: a 5-tuple of 64th notes with an intended duration of *six* 64ths. **autosp** cannot deal with simultaneous x-tuplets in multiple staves unless the x values and note durations are identical.

User-defined macros are not processed or expanded.

All staves are assumed to have the same meter; see `kinder2.aspc` for an example of how to work around this.

**autosp** may not be effective for music with more than one voice in a single staff. It might be advisable to use a separate staff for each voice, to avoid `\anotes` when necessary, or to omit certain voices initially and add them into the resulting TeX file.

## EXAMPLES

See files `quod2.aspc`, `kinder2.aspc`, `geminiani.aspc` and `barsant2.aspc` for scores suitable for input to **autosp**.

## SEE ALSO

`msxlint(1)`

`musixdoc.pdf`

## AUTHOR

This program and manual page were written by Bob Tennent <rdt@cs.queensu.ca>.