

**NAME**

`autosp` – preprocessor to generate note-spacing commands for MusiXTeX scores

**SYNOPSIS**

`autosp` [`-v` | `--version` | `-h` | `--help`]

`autosp` [`-d` | `--dotted`] [`-l` | `--log`] *infile* [.aspc] [*outfile* [.tex]]

**DESCRIPTION**

This program makes it easier to create MusiXTeX scores by converting (non-standard) commands of the form `\notes ... \en` (or `\an ... \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 \qa m\ca n\en
```

`autosp` generates

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

Typically, an `\notes` command generates several conventional note-spacing commands.

If the *infile* argument does not have an .aspc extension, input is taken from *infile.aspc* if that file exists.

If the *outfile* argument does not have a .tex extension, output is sent to *outfile.tex*. If no *outfile* argument is provided, output will go to *infile.tex* (or to *infile.ltx* if a `\documentclass` declaration is encountered).

For `\notes` 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,

```
\an
\ibl0K0\qb0K\bb10\qb0{.K}\tbbb10\tbb10\tq10L&
\ibb11m{-2}\qb1{.m}\tbbb11\tbb11\qb11\tq11k\en
```

is acceptable and generates

```
\notesp\ibl0K0\qb0K&\ibb11m{-2}\qb1{.m}\en
\notes\sk&\tbbb11\tbb11\qb11\en
\notesp\bb10\qb0{.K}&\tq11k\en
```

`\notes\tbbb10\tbb10\tq10L&\sk\en`

If the `-l` (`--log`) option is used, a very detailed 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 `\defatnextbar` notation but the non-standard command `\Cpause` (or `\Cp`) in a note segment also generates a bar-centered rest.

Spacing commands `\sk` and `\bsk` in the source are discarded, but not "small" skips `\hsk`, `\qsk`, `\tqsk`, `\hqsk` or `\qqsk`, or the small "backward" skips `\hbsk`, `\bqsk`, `\btsk`, or `\bhsk`. Moreover, non-standard commands `\QQsk`, `\HQsk`, `\TQsk` and `\Qsk` in the source generate "global" skips; i.e., the effect of `\qqsk`, `\hqsk`, `\tqsk` or `\qsk`, respectively, in *every* staff. These ensure that staves remain synchronized if additional spacing is needed in any staff(s).

Global skips may also be obtained within collective-coding sequences by using up to four successive commas to get the effects of `\QQsk`, `\HQsk`, `\TQsk` or `\Qsk`, respectively. Global skips `\tqsk` (or, for double-flats, `\qsk`) are automatically inserted before accidentals (`'^'`, `'_'`, `'='`, `'<'`, `'>'`) on collective-coding notes (except when the preceding note is "virtual"; i.e., a skip). If this automatic additional spacing is *not* wanted in some context, it may be avoided by replacing the accidental in the collective-coding sequence by any of the *explicit* accidental commands: `\sh`, `\fl`, `\na`, `\smallsh`, `\bigsh`, etc. If the automatic spacing is insufficient, the spacing may be increased by adding sufficient commas or using a conventional notes command instead of `\an`.

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{*}`, `\qa{*}`, `\ca{*}`, etc.

From version 2017-06-14, the effects of `\TransformNotes` calls are implemented by the **autosp** pre-processor. This enables use of `musixlyr` in **autosp** scores; `musixlyr.tex` is incompatible with the `musixtnt.tex` implementation of `\TransformNotes`.

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.

**autosp** scales notes by the relevant instrument-size value.

## 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}` or `\xxtuplet{x:y}` and *triplets* introduced using any of the following commands (regardless of any re-definition of `\txt` or `\tuplettxt`):

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

**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 suitable `\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.

In some polyrhythmic scores, the `\txt` numeral may be displaced, even if the notes themselves are correctly spaced. In these cases, it is possible to suppress the normal output of `\txt` by using the non-standard commands `\Triolet` (no arguments) or `\Xtuplet{k}` and placing a numeral at the correct location using `\zcn` (i.e., `\zcharnote`).

**autosp** can deal with simultaneous x-tuplets in multiple staves provided the x values and total note durations are identical.

In some baroque scores, particularly by J.S. Bach, a beamed sixteenth note is vertically aligned with the third note of a triplet of eighth notes in another staff (implying that they should be sounded simultaneously); e.g.,

```
\ibl0L0\qb0{.L}\tqq10L
```

would be played as if notated

```
\uptrio{b}10\ql L\hroff{\cl L}
```

The following coding will align the beamed sixteenth note with the third note of a triplet in another staff:

```
\ibl0L0\qb0{.L}\hbsk\tqq10L
```

and, similarly, for triplets of sixteenth notes:

```
\ibbu0J0\qb0{.J}\hbsk\inqqu0J\qb0{.J}\hbsk\tqqqu0J
```

Generally, user-defined macros are not processed or expanded; however, definitions of the form

```
\def\atnextbar{\znotes ... \en}
```

generate definitions that do take account of `\TransformNotes`.

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**. The program **tex2aspc** can be used to convert "legacy" MusiXTeX scores to `.aspc` format.

## SEE ALSO

**msxlint(1)** **tex2aspc(1)**

`musixdoc.pdf`

## AUTHOR

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