NAME

fixmsxpart − corrects note spacing in a single-staff MusiXTeX part

SYNOPSIS

fixmsxpart [**−v | −−version | −h | −−help**]

fixmsxpart [**−d** | **−−debug**] *infile*[**.tex**] [*outfile*[**.tex**]]

Converts a single-staff MusiXTeX part (possibly derived from a multi-instrument score and as a result having irregular note spacing) to a single-staff part with proper spacing determined by the notes themselves.

If *outfile* is not specifed, standard output is used.

USAGE

Generating a Single-Instrument Part

To generate a single-instrument part from (a copy of) the MusiXTeX source for a multi-instrument score, add

```
\input musixtnt
```

to the preamble, set

```
\instrumentnumber1
```

and use the \TransformNotes macro defined in musixtnt.tex to discard all but one part. For example, the following line placed after \startpiece (but before any note commands) would be appropriate for a *four*-instrument score (arguments #2, #3, #4, and #5, separated by three &s), and will result in a part for the *second* of these (#3):

```
\TransformNotes{#2&#3&#4&#5}{#3}
```

Argument #1 is a scaling parameter and should not be modified. **It is essential that every \znotes, \notes, \Notes, \NOtes, etc. command in the score match the macro pattern exactly;** insufficient (or too many) note segments will result in lost text and possibly compilation failure; see **msxlint**(1). It is assumed by \TransformNotes that notes commands are terminated by \en (rather than \enotes).

Some additional manual changes to the source will be necessary:

+ adjustments of \setname1, \setclef1, \setsign1 , \setmeter1 and \setstaffs1 commands, as necessary;
+ ensuring that tempo and roadmap markings (**D.C.**, **Fine**, etc.) are in the appropriate instrument segment;

Finally, if the modified score is compiled and viewed, it may be seen that horizontal spacing designed for *multiple* instruments often produces bad spacing for a *single* instrument. This can be corrected manually (and very tediously) but it is what **fixmsxpart** was designed to fix (much more conveniently).

Correcting Horizontal Spacing Using fixmsxpart

The \notes \Notes \NOtes \NOTes... commands in a part derived from a multi-instrument score are unreliable, and so **fixmsxpart** determines the spacing for ordinary notes by the note commands themselves; for example,

+ \qa, \qu, \ql, \qp result in \NOtes;
+ \ca, \cu, \cl, \ds result in \Notes;

and so on. Spacing commands \sk and \hsk in the input are discarded (but \qsk, \hqsk and explicit uses of

\off{...} are preserved).

fixmsxpart determines the spacing for *beamed* notes by the beam multiplicity: \ib... results in \Notes, \ibb... results in \notes, etc. Some beam-termination commands (\tb...) are problematic because the immediately-following beam note \qb... must be spaced according to the spacing *before* the \tb... command. This is handled by setting a variable *new_beaming* when processing the \tb... command and only updating *beaming* to *new_beaming* when processing the subsequent beam note. This isn't needed for the abbreviations \tqb, \tqh, \ztqb and \ztqh which combine beam-termination and specification of the final beam note.

Pointed notes are *not* given extra space automatically. They should be given extra space manually in the score by using either a \roff-like command on the subsequent note or the \qsk or \hqsk spacing commands.

Additional features of **fixmsxpart** transformation:

+ Successive whole-bar rests are accumulated into a multi-bar rest, with appropriate adjustment of the bar number.
+ \alaligne and \alapage commands are replaced by \bar.
+ \mulooseness, \eject, \linegoal, \song{top | bottom}, \group{top | bottom} and \akkoladen commands are commented out.
+ \instrumentnumber... commands become \instrumentnumber1.
+ \nostartrule is added to the preamble.
+ Simple slurs or ties initiated by \slur or \tie are normalized by setting the *n* parameter to 1 on the assumption that the new note spacing is correct.

LIMITATIONS

Only single-staff instrumental parts are supported. User-defined macros are not expanded. A few esoteric MusiXTeX commands and constructions are not supported.

SEE ALSO

msxlint(1)

musixdoc.pdf

AUTHOR

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