

Text file syntax

The text file must be placed in the same directory as the imposition file, and it must have the same name as the imposition file, except ending with a `.txt` extension.

Syntax

The text strings must be arranged in a hierarchy by imposition object.

Configure one line per object.

Each line of the text file must be formatted as:

```
<Object> <ID> <String 1>; <String 2>; ... <String N>
```

where,

- `<Object>` is:
 - Signature or Sg
 - Sheet or Sh
 - Surface or Su
 - Page or Pg

Objects are case-insensitive.

- `<ID>` is a numeric identifier or a Preps-style alpha surface identifier. The surface identifiers Front or Fr and Back or Ba are also supported. Pages are identified by their run-list position.

More syntax rules

- Strings must be separated by semicolons (and therefore cannot contain semicolons).
- Objects should be listed in hierarchical order. In other words, signature 1 should be listed before any of the objects in that signature. The order of hierarchy is:
 - Signatures
 - Sheets
 - Surfaces
 - Pages
- If a higher level object is not listed, default values are assumed.
- The file must not contain duplicate objects, and objects must be listed in their hierarchical order unless defaults are being assumed for higher-level objects. If duplicates are found or objects are listed out of order, the output task will fail with an error message listing the problem line. The error must be corrected before the task is resubmitted.

See [Example: text-file-based marks](#).