

Music Braille in OpenSheetMusicDisplay — User Guide

OpenSheetMusicDisplay · User Documentation · July 2026

This guide explains the **braille music feature** of OpenSheetMusicDisplay (OSMD) for everyone who wants to use it: what it is, whom it helps, and how to work with it — no programming knowledge required.

Try it right now, free, in your browser — nothing to install:

Public demo: <https://opensheetmusicdisplay.org/wp-content/uploads/sites/2/osmd-braille-demo/index.html>

If you are a software developer and want to build the feature into your own application, read the [developer documentation](#) instead — this guide is about using it.

Contents

1. What is it?
2. Who is it for, and how does it help?
3. How does it work?
4. 3.1 Your first braille score
5. 3.2 Choosing music — samples and your own files
6. 3.3 Reading the braille output
7. 3.4 The translation table
8. 3.5 The braille options: keyboard, ensemble, lyrics, facsimile
9. 3.6 Using a screen reader and a braille display
10. 3.7 Getting the braille out: copying, embossing
11. 3.8 Tips and troubleshooting
12. Current limitations
13. Links and further reading

1. What is it?

The braille feature turns **digital sheet music into braille music notation — instantly and automatically**. You open a piece of music (any file in the widely used **MusicXML** format), and next to the printed score OSMD shows the same music as braille: a sequence of braille characters that a blind musician can read on a refreshable braille display, enlarge on screen, copy into other programs, or emboss on paper.



OSMD Demo

OSMD Version: 2.0.0

Select a sample:

test Braille ScaleSimple.musicxml

Open file...

Show

Previous

Reset

Hide

Next

☐ Follow Cursor
 ☐ Incremental rendering (load on scroll)

Show bounding box for:

None

Show debug information:

☐ Skyline
 ☐ Bottomline

Braille options:

☒ Show classical score above braille
 ☐ Enable braille lyrics
 ☐ Enable facsimile mode
 ☐ Bar-over-bar (keyboard)
 ☐ Ensemble (instrument abbreviations)

Debug controls:

Re-render

Clear

Transpose by Semitone(s):

Transpose

Page size:

endless

Create PDF

Dark mode

test_Braille_ScaleSimple

Music



Music Braille Score



Braille Debug Output

nonfacsimile mode

#	Braille	Meaning	Measure
0	⠠⠠⠠⠠⠠⠠	time 4/4	1
1	⠠⠠	octave 4	1
2	⠠⠠	quarter/64th C4	1
3	⠠⠠	quarter/64th D4	1
4	⠠⠠	quarter/64th E4	1
5	⠠⠠	quarter/64th F4	1
6	⠠⠠⠠⠠⠠⠠	barline	1
7	⠠⠠	quarter/64th G4	2
8	⠠⠠	quarter/64th A4	2

Braille music is not an invention of this project — it is the standardized notation blind musicians have used for over 150 years, today defined by the *Music Braille Code* (2015 edition, Braille Authority of North America). Like literary braille it is written in cells of six dots, but the signs mean musical things: each note sign combines pitch and rhythm in a single cell, small prefix signs give the octave, a blank cell is a barline. The OSMD braille feature produces exactly this standard notation, and it also offers the standard's special layouts for piano music, ensemble scores, and songs with lyrics.

What makes it useful is the *automatic* part: braille music has traditionally been produced by specially trained transcribers, which is slow and expensive — only a small fraction of all sheet music has ever been transcribed. Here, the transcription happens in about a second, for any MusicXML file, every time.

You can use the braille feature in three ways: in the **free public demo** (link above), in the free **OSMD WordPress plugin** — which brings braille output to scores embedded on WordPress websites — and in any application whose developers build it in with the OSMD library.

What it is not: it does not read scanned paper or PDF files (the music must be in MusicXML format — see [section 3.2](#) for how to get that), and for professionally published braille editions a human transcriber's editorial judgment is still valuable. Think of it as an always-available transcriber for everyday needs: practicing, teaching, checking, exploring.

2. Who is it for, and how does it help?

Blind and visually impaired musicians. You can read any MusicXML score, immediately, without waiting for a transcription. The demo works in an ordinary web browser with your screen reader; the braille output sits under its own heading so it is easy to jump to, and every button is labeled. Reading works on a refreshable braille display, by touch reading an embossed printout, or visually enlarged on screen. An "Open file..." button lets you load your own music without needing drag-and-drop. *By which means:* a free public web

demo (link above), usable with NVDA, JAWS and other screen readers, plus the open-source OSMD library behind it.

Music teachers — including sighted teachers of blind students. Turn the week's teaching material into braille in seconds, so a braille-reading student works from the same piece as everyone else. The built-in **translation table** explains every braille sign in plain words ("quarter/64th C4", "2 sharps", "right hand"), so you can verify the braille and even use the demo to *learn* braille music notation yourself — it doubles as an interactive textbook. *By which means:* the demo's side-by-side view of print score, braille, and sign-by-sign translation.

Braille music transcribers and accessibility services. Get a complete, standard- conforming first transcription automatically — including the Music Braille Code's bar-over-bar keyboard format, ensemble format with instrument abbreviations, and vocal format with word lines — and spend your time on editorial polish instead of note-by-note work. The output is plain Unicode braille text that can be copied into any braille editing workflow. *By which means:* standard-compliant automatic transcription (Music Braille Code 2015) with copyable text output.

Choirs, music schools, families, self-learners. Songs with lyrics are supported: the words appear as braille text lines paired with the melody lines, verse by verse. *By which means:* the "braille lyrics" option and ready-made song samples.

Websites that publish sheet music. With the free [OSMD WordPress plugin](#), WordPress sites — score archives, teaching blogs, choir and parish pages — can embed scores together with their braille version, no programming required. *By which means:* the free OSMD WordPress plugin with built-in braille output.

Developers of music software. Every application that displays sheet music with OSMD can offer braille output with a few lines of code — see the [developer documentation](#).

3. How does it work?

3.1 Your first braille score

1. Open the demo: <https://opensheetmusicdisplay.org/wp-content/uploads/sites/2/osmd-braille-demo/index.html>
2. A sample piece loads automatically. The printed score appears in the middle of the page, and the braille version appears under the heading "**Music Braille Score**".
3. That's it — every time you pick a different piece or change an option, the braille updates immediately.

For screen reader users: the page has a clear heading structure. Jump to the level-2 heading "**Music Braille Score**" (for example with NVDA's **H** key or the elements list) — the braille music follows directly after it, and the translation table follows under the level-3 heading "**Braille Debug Output**".

3.2 Choosing music — samples and your own files

Select a sample:

test Braille Accidentals.musicxml

▼

Open file...

Samples. The dropdown list at the top starts with about thirty small **braille test pieces**, each named after the notation it demonstrates — `test_Braille_ScaleSimple` (a plain scale, the best starting point), `test_Braille_Accidentals`, `test_Braille_ChordsSimple`, `test_Braille_Repeats`, and so on. After them come complete pieces of classical repertoire: Clementi sonatinas (piano), Bach, Beethoven's song *An die ferne Geliebte*, Mozart, a string quartet, and *Land der Berge* (the Austrian national anthem, with three verses of lyrics).

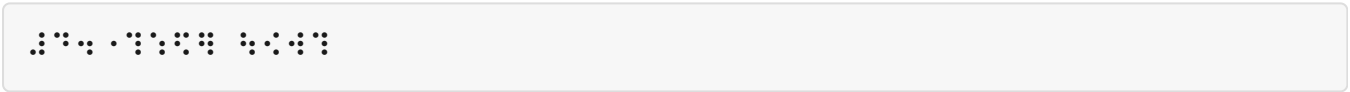
Your own music. Use the **"Open file..."** button (or drag a file anywhere onto the page) to load your own `.xml`, `.musicxml` or `.mxl` file. Where to get MusicXML files:

- **Notation programs** can export it: in the free [MuseScore](#) use *File* → *Export* → *MusicXML*; Sibelius, Finale, Dorico, capella and others have equivalent export commands.
- **Online score libraries** (including the MuseScore community site and the OpenScore editions of public-domain works) offer MusicXML downloads.
- **Scanning apps** (music OCR, e.g. PhotoScore, ScanScore, Audiveris) can turn printed sheet music into MusicXML — quality depends on the scan.

3.3 Reading the braille output

The braille under "Music Braille Score" is genuine Unicode braille text — not a picture. You can read it directly on a braille display, copy and paste it, or zoom your browser to enlarge it.

A tiny example so you know what you are looking at. A C major scale in quarter notes, two measures of 4/4, looks like this in braille:



Reading left to right:

Braille	Meaning
<code>.:.</code>	time signature 4/4
<code>.</code>	octave mark: the next note is in the 4th octave (the octave of middle C)
<code>⠠⠠⠠⠠</code>	quarter notes C, D, E, F — each sign is pitch <i>and</i> rhythm in one cell
(space)	the barline — in braille music a measure ends with a blank cell
<code>⠠⠠⠠⠠</code>	quarter notes G, A, B, C

A few things braille music does differently from print, in case you are new to it: there are no staff lines and normally no clefs — small **octave marks** like `.` say how high a note is, and they are only written when the reader could be in doubt. **Chords** are written as one note plus interval signs. **Piano music** marks the hands with `⠠⠠` (right hand) and `⠠⠠` (left hand). You do not need to know any of this to *produce* braille with the demo — but if you want to learn to *read* it, the translation table below is your friend, and [section 5](#) lists the official manuals.

3.4 The translation table

Under the braille, the heading **"Braille Debug Output"** introduces a table with one row per braille sign: the sign itself, its meaning in plain words, and the measure it belongs to.

Music Braille Score

Braille Debug Output

nonfacsimile mode

#	Braille	Meaning	Measure
0	⠠⠠⠠⠠	time 4/4	1
1	⠠	octave 4	1
2	⠠⠠	quarter/64th C4	1
3	⠠⠠	quarter/64th D4	1
4	⠠⠠	quarter/64th E4	1
5	⠠⠠	quarter/64th F4	1
6		barline	1
7	⠠⠠	quarter/64th G4	2
8	⠠⠠	quarter/64th A4	2
9	⠠⠠	quarter/64th B4	2
10	⠠⠠	quarter/64th rest	2

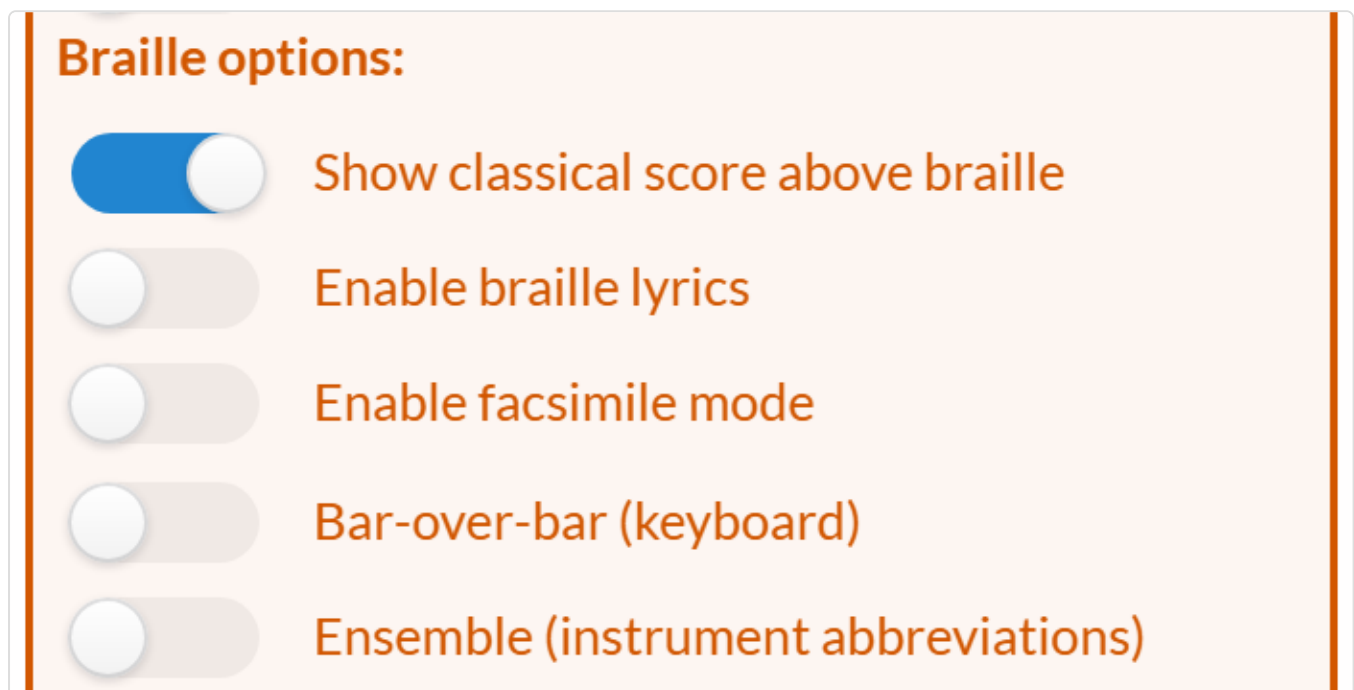
This table is useful in three ways:

- **Verification** — a sighted teacher or parent who does not read braille can check what the student's braille says, sign by sign, measure by measure.
- **Learning** — read the braille with one hand and the meanings with the other (or with your screen reader): the demo becomes a self-explaining braille music course.
- **Trust** — if a sign surprises you, the table tells you exactly what the software meant by it.

(The name "debug output" is a leftover from development — it is simply a translation of every sign.)

3.5 The braille options: keyboard, ensemble, lyrics, facsimile

By default the demo produces **standard braille music** ("nonfacsimile" format): the music flows measure by measure, and pieces with two staves — like piano music — are written as one line per staff with hand signs. The left control panel, under "**Braille options:**", holds five toggle switches:



Show classical score above braille — on by default. Switch it off to see *only* the braille, without the printed (classical) score above it. This is more than a cosmetic choice: with the printed score switched off, the demo skips drawing it entirely when a piece is loaded, so **large scores load noticeably faster** — the braille conversion itself is quick; it is the visual score rendering that takes time on big pieces. (The one exception is facsimile mode, which needs the visual layout internally — see below.) Switch it back on at any time; the printed score reappears immediately.


OSMD Demo

OSMD Version: 2.0.0

Select a sample:
test Braille ScaleSimple.musicxml
Open file...

100%

Cursor controls:

Show
Previous
Reset

Hide
Next

☐ Follow Cursor
☐ Incremental rendering (load on scroll)

Show bounding box for:

None

Show debug information:

☐ Skyline
☐ Bottomline

Braille options:

☐ Show classical score above braille
☐ Enable braille lyrics
☐ Enable facsimile mode
☐ Bar-over-bar (keyboard)
☐ Ensemble (instrument abbreviations)

Debug controls:

Re-render
Clear

Transpose by Semitone(s):

0
Transpose

Page size:

endless

Create PDF

Dark mode

Music Braille Score

Braille Debug Output

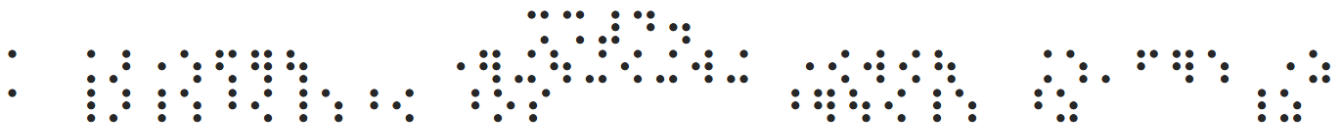
nonfacsimile mode

#	Braille	Meaning	Measure
0	⠠⠠⠠⠠⠠⠠	time 4/4	1
1	⠠⠠	octave 4	1
2	⠠⠠	quarter/64th C4	1
3	⠠⠠	quarter/64th D4	1
4	⠠⠠	quarter/64th E4	1
5	⠠⠠	quarter/64th F4	1
6	⠠⠠	barline	1
7	⠠⠠	quarter/64th G4	2
8	⠠⠠	quarter/64th A4	2
9	⠠⠠	quarter/64th B4	2
10	⠠⠠	quarter/64th rest	2

The other four toggles choose special braille layouts defined by the Music Braille Code:

Bar-over-bar (keyboard) — the classic format for piano music. Right-hand and left-hand lines are placed directly above each other, measure starts aligned, with the measure number at the left margin and the key and time signature centered above as a heading. Try it with `test_Braille_BarOverBar` or a Clementi sonatina.

Music Braille Score



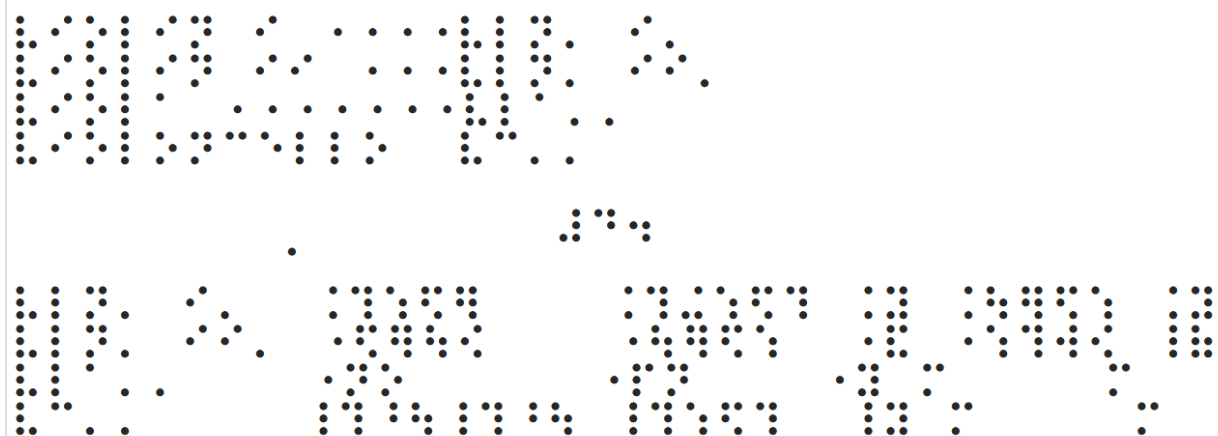
Braille Debug Output

nonfacsimile mode

#	Braille	Meaning	Measure
0	⠠⠠	2 sharps	0
1	⠠⠠⠠	time 4/4	0
2		heading separator (newline)	0
3	⠠	measure 1	1
4	⠠⠠	right hand	1
5	⠠	octave 4	1
6	⠠	quarter/64th D4	1
7	⠠	quarter/64th E4	1
8	⠠	quarter/64th F#4	1
9	⠠	quarter/64th G4	1
10		alignment padding	1
11		barline	1
12	⠠	octave 4	2
13	⠠	quarter/64th F#4	2
14	⠠	3rd interval	2
15	⠠	quarter/64th G4	2
16	⠠	3rd interval	2
17	⠠	quarter/64th A4	2
18	⠠	3rd interval	2
19	⠠	quarter/64th B4	2
20	⠠	3rd interval	2
21		barline	2
22	⠠	octave 4	3
23	⠠	quarter/64th A4	3
24	⠠	quarter/64th B4	3

Ensemble (instrument abbreviations) — for scores with several instruments, for example a string quartet. The output starts with an instrument list (full names and their braille abbreviations), then each group of measures shows one line per instrument, each line starting with its abbreviation. Instruments that only rest are omitted from that group — as the standard prescribes. Try it with `test_Braille_Ensemble_Quartet` .

Music Braille Score



Braille Debug Output

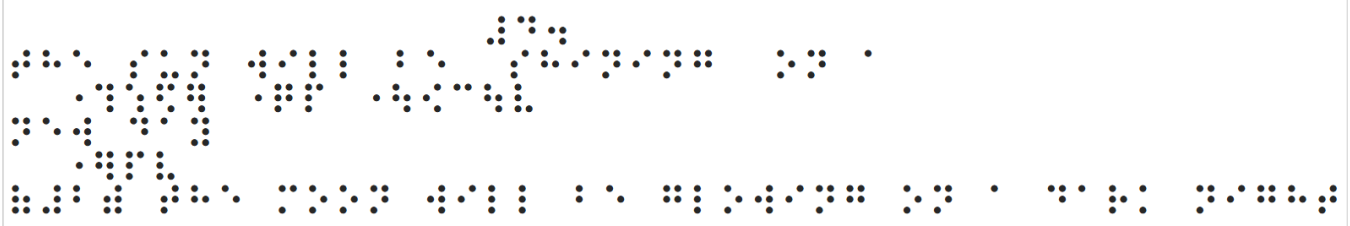
nonfacsimile mode

#	Braille	Meaning	Measure
0		instrument list entry	0
1		instrument list entry	0
2		instrument list entry	0
3		instrument list entry	0
4		instrument table separator	0
5		time 4/4	0
6		heading separator (newline)	0
7		measure 1	1
8		measure number separator (newline)	1
9		instrument	1
10		octave 5	1
11		quarter/64th C5	1
12		quarter/64th D5	1
13		quarter/64th E5	1
14		quarter/64th F5	1
15		alignment padding	1
16		barline	1
17		octave 5	2
18		quarter/64th C5	2
19		3rd interval	2
20		quarter/64th D5	2
21		quarter/64th E5	2
22		quarter/64th C5	2
23		barline	2

24	.	octave 5	3
25	::	whole/16th C5	3
26		barline	3
27	:	octave 5	4
28	::	quarter/64th G5	4

Enable braille lyrics — for songs. The words are written as literary braille on their own line, with the melody on the music line below; each syllable belongs to the next note. Additional verses follow at the end as continuous text. Try it with `test_Braille_Lyrics_Simple` , *An die ferne Geliebte*, *Das Veilchen*, or *Land der Berge*.

Music Braille Score



Braille Debug Output

nonfacsimile mode

#	Braille	Meaning	Measure
0	:::	time 4/4	0
1		heading separator (newline)	0
2	:::	lyric: the	1
3	:::	lyric: sun	1
4	:::	lyric: will	1
5	::	lyric: be	1
6	:::	lyric: shi	2
7	:::	lyric: ning	2
8	:::	lyric: on	3
9	.	lyric: a	3
10	.	octave 4	1
11	::	quarter/64th C4	1
12	::	quarter/64th D4	1
13	::	quarter/64th E4	1
14	::	quarter/64th F4	1
15		barline	1
16	.	octave 4	2
17	::	half/32nd F4	2
18	::	half/32nd E4	2
19		barline	2
20	.	octave 4	3
21	::	quarter/64th G4	3
22	::	quarter/64th A4	3

Enable facsimile mode — braille that mirrors the printed page. Normally braille music deliberately ignores print layout; in facsimile mode each braille line corresponds to one printed line (system) of the score, and print-specific signs such as clefs and ottava brackets are included. This is useful when a braille reader and sighted musicians work from the same printed edition and want to talk about "the second line". The sample `test_Braille_Facsimile_option` switches this mode on automatically.

The screenshot shows the OSMD Demo interface. At the top, there's a logo and 'OSMD Demo' text. Below it, a version number 'OSMD Version: 2.0.0' is visible. A 'Select a sample:' dropdown menu is set to 'test_Braille_Facsimile_option.musicxml', with an 'Open file...' button next to it. The main title 'test_Braille_Facsimile_option' is centered. Below the title, a musical score is displayed on a staff. To the left of the score, there's a sidebar with various controls. The 'Cursor controls' section includes buttons for 'Show', 'Previous', 'Reset', 'Hide', and 'Next', along with checkboxes for 'Follow Cursor' and 'Incremental rendering (load on scroll)'. The 'Show bounding box for:' dropdown is set to 'None'. The 'Show debug information:' section has checkboxes for 'Skyline' and 'Bottomline'. The 'Braille options:' section includes checkboxes for 'Show classical score above braille', 'Enable braille lyrics', 'Enable facsimile mode' (which is checked), 'Bar-over-bar (keyboard)', and 'Ensemble (instrument abbreviations)'. The 'Debug controls:' section has 'Re-render' and 'Clear' buttons. The 'Transpose by Semitone(s):' section has a '0' input and a 'Transpose' button. The 'Page size:' dropdown is set to 'endless'. At the bottom of the sidebar, there are buttons for 'Create PDF' and 'Dark mode'. The main area also shows a 'Music Braille Score' section with a 'Braille Debug Output' table. The table has columns for '#', 'Braille', 'Meaning', and 'Measure'. The data rows are as follows:

#	Braille	Meaning	Measure
0	♩	treble clef	1
1	♯	1 sharp	1
2	♩	time 3/4	1
3	♩	octave 4	1

Because facsimile mode mirrors the *rendered* layout, it is the one mode that always needs the visual score to be computed internally — if you have "Show classical score above braille" switched off, the demo still does that work behind the scenes (the printed score simply stays hidden), so facsimile does not get the faster loading described above.

Two practical notes: pick the option that matches the piece (the ensemble format on a piano piece, or lyrics on an instrumental piece, will produce technically valid but odd-looking results), and switch options off again before exploring other pieces — everything re-renders instantly, nothing is lost.

3.6 Using a screen reader and a braille display

The demo was built and tested with screen reader use in mind (NVDA on Windows, and the same techniques work with JAWS and others):

- **Finding the braille:** jump by headings. "**Music Braille Score**" is a level-2 heading; the braille text follows immediately. The translation table has its own level-3 heading, "**Braille Debug Output**".
- **Reading on a braille display:** the output is ordinary Unicode braille text, so your screen reader sends it to the display like any text — move through it with your usual reading commands. The line width of the special formats (40 cells) matches standard braille displays and embossers, and because every cell is the same width on a display, the vertical alignment of the keyboard and ensemble formats is exact.
- **All controls are labeled:** the sample list, the "Open file..." button (which exists precisely because drag-and-drop is not usable without a mouse), the four braille option toggles, zoom, and dark mode.

- **Loading your own file:** activate "Open file...", pick the file in the standard file dialog, and the braille updates.
- **Tip: switch off the printed score.** If you work purely with braille, disable "Show classical score above braille" in the braille options — pieces load faster (see [section 3.8](#)) and the page holds only the content that matters to you.

3.7 Getting the braille out: copying, embossing

- **Copy and paste.** Select the braille text and copy it like any text — into a document, a note-taking app, or a braille editor. It stays braille wherever Unicode text is supported.
- **Embossing on paper.** Braille production software (for example braille editors used by transcription services) can take Unicode braille text and send it to an embosser; some tools may first convert it to the older ASCII braille (BRF) encoding. The 40-cell line width of the formatted layouts fits standard braille paper.
- **Keeping it.** Save the copied braille as a text file — it remains readable on any device with braille support, independent of OSMD.

3.8 Tips and troubleshooting

- **The braille columns look slightly misaligned on screen.** That is a limitation of ordinary screen fonts, not of the braille: some browsers draw braille characters with tiny width differences. On a braille display or embossed page the alignment is exact. A dedicated braille font (such as SimBraille) also fixes the visual alignment.
- **No braille appears for my file.** Make sure it is a MusicXML file (`.xml` , `.musicxml` , `.mxl`). If the demo shows an error message, the file could not be read — try re-exporting it from your notation program. Test with a built-in sample first to confirm everything else works.
- **Which sample shows feature X?** The `test_Braille_...` names say what they demonstrate: Accidentals, ChordsSimple, TiesSimple, SlursSimple, Repeats, Voltas, Dynamics, Ornaments, DaCapo, Ottava, MultiStaff_Piano, and more.
- **Large scores load faster without the printed score.** Most of the waiting time on a big piece is spent drawing the printed score — the braille conversion itself is quick. If you only need the braille, switch off "**Show classical score above braille**" in the braille options ([section 3.5](#)): the visual rendering is then skipped entirely and pieces load noticeably faster. The exception is facsimile mode, which always needs the visual layout internally and therefore takes the usual time even with the printed score hidden.
- **Low vision:** the demo has a **Dark mode** button and zoom controls; browser zoom (`Ctrl` + `+`) enlarges the braille text itself.

4. Current limitations

The feature covers the everyday range of music notation — notes, rests, accidentals, key and time signatures, chords, multiple voices, ties, slurs, dynamics, articulations, ornaments, repeats and endings, D.C./D.S. directions, piano/ensemble/vocal layouts, and facsimile mode. Things it does not do yet:

- **Grace notes and tuplets** (e.g. triplets) are written as ordinary notes — the notes are all there, but without the special grace-note and triplet signs.

- Some rare layout refinements of the braille standard (splitting one measure across lines, braille-specific repeat abbreviations, specialized choral layouts) are not applied; the music is always written out in full, which is correct, just sometimes longer than a human transcriber would make it.
- Lyrics are transcribed in uncontracted (Grade 1) braille, first voice.

The complete, regularly updated list is in the [developer documentation](#).

5. Links and further reading

- **The public demo:** <https://opensheetmusicdisplay.org/wp-content/uploads/sites/2/osmd-braille-demo/index.html>
- **OpenSheetMusicDisplay:** <https://opensheetmusicdisplay.org> — the open-source sheet music renderer this feature belongs to (GitHub: <https://github.com/opensheetmusicdisplay/opensheetmusicdisplay>).
- **OSMD WordPress plugin** — embed scores with braille output on your own WordPress site: <https://wordpress.org/plugins/opensheetmusicdisplay/>.
- **Developer documentation** of the braille feature: [README.md](#) — for integrating it into your own application or continuing its development.
- **Learning braille music:**
- *Music Braille Code 2015*, Braille Authority of North America (BANA) — the official standard this feature implements; available free of charge at brailleauthority.org.
- *New International Manual of Braille Music Notation* (ed. B. Krolick, 1996), World Blind Union.
- **MuseScore** (free notation software with MusicXML export): <https://musescore.org>.

Development of this feature was supported by a **netidee** grant (Internet Foundation Austria). Progress history with many output examples is documented in the project's development issue tracker.