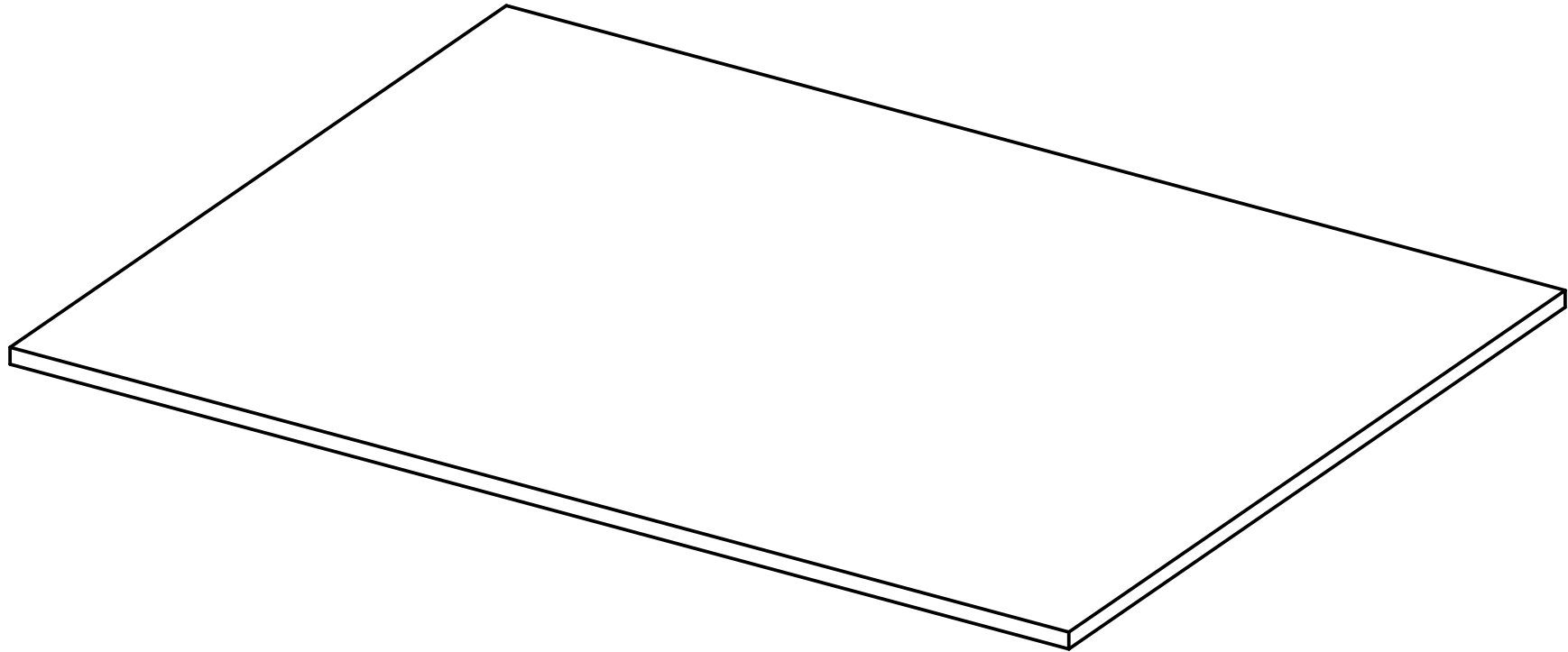


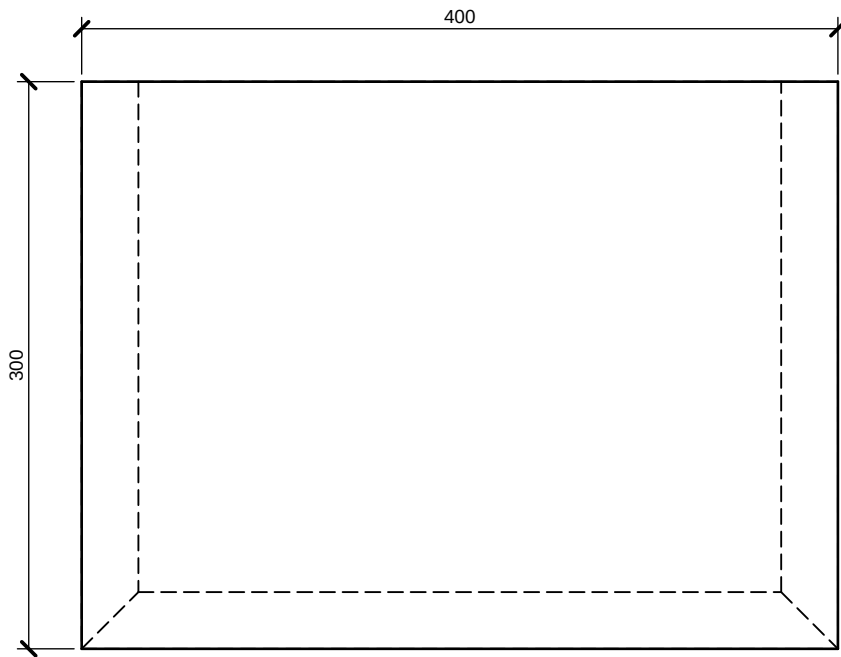
Tapered panel



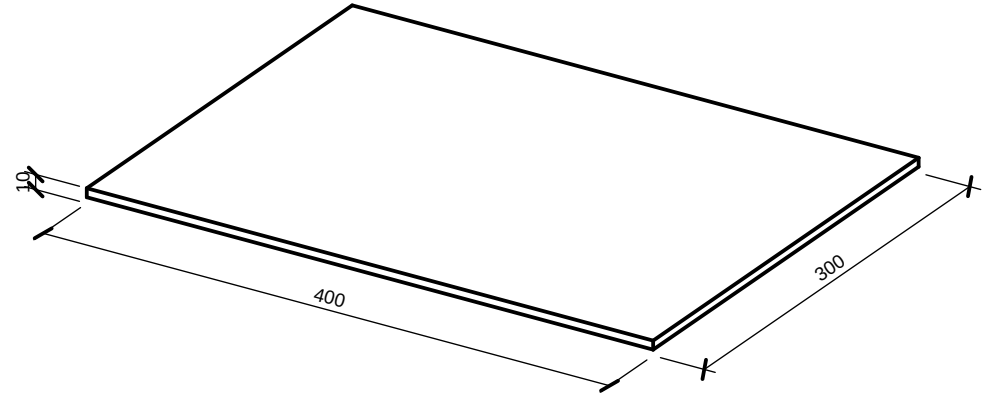
parts	1
overall	400 x 300 x 10 (W x D x H)
material	oak
date	2026-10-25
made with	woodwright 0.9.0

Contents

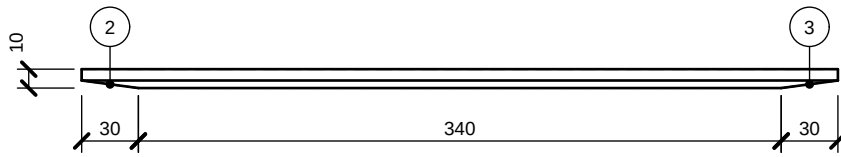
Drawings	3
1 board	3
Cut list	4
Appendix: source	5



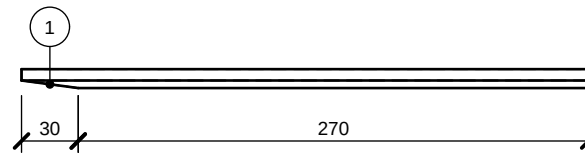
TOP FACE



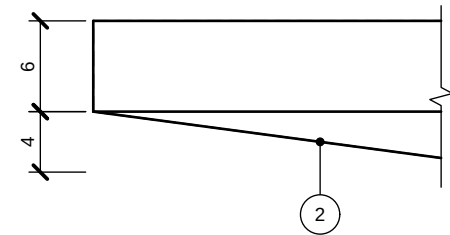
ISOMETRIC



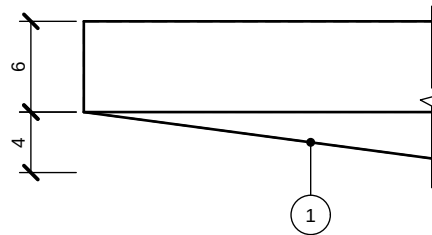
FRONT EDGE



RIGHT END



FRONT EDGE: 2, 3 2:1



RIGHT END: 1 2:1

- 1 chamfer 30 x 4 mm on bottom_face_front_edge_arris
- 2 chamfer 30 x 4 mm on bottom_face_left_end_arris
- 3 chamfer 30 x 4 mm on bottom_face_right_end_arris

part	board
of	tapered_panel
make	1
cut size	400 x 300 x 10
material	oak
scale	1:4, mm

Cut list

board 400 x 300 x 10 mm oak

- chamfer 30 x 4 mm on bottom_face_front_edge_arris
- chamfer 30 x 4 mm on bottom_face_left_end_arris
- chamfer 30 x 4 mm on bottom_face_right_end_arris

Appendix: source

Made with woodwright 0.9.0, build123d 0.13.0, Python 3.14.4.

To make these plans again:

```
uvx --python 3.14 woodwright==0.9.0 export cuts.tapered_panel
```

cuts.tapered_panel, in woodwright/examples/cuts.py

```
def tapered_panel(
    length: float = 400,
    width: float = 300,
    taper_width: float = 30,
    edge_thickness: float = 6,
) -> Assembly:
    """A drawer bottom with three edges tapered on the underside to fit grooves.

    The underside is planed at a slope `taper_width` wide, leaving the edge
    as thick as the groove is wide. The back edge stays square: it goes
    under the back of the drawer.
    """
    asm, board = _board("tapered_panel", THIN_OAK, length=length, width=width)
    amount = board.thickness - edge_thickness
    for at in (board.front_edge, board.left_end, board.right_end):
        board.cut(Chamfer(board.aris(board.bottom_face, at), size=taper_width, size_b=amount))
    return asm
```