ADDALY · THE AI CLASSROOM

Vectors, Logos and Things That Get Printed

Draw curves that hold at any size, then hand a press a file it will not send back.

---

8 modules · 73 lessons · 29 figures · 8 case studies · 56,934 words · about 10 hours

Dr Mustafa Mun
Author and editor

ABOUT THIS BOOK

Vectors, Logos and Things That Get Printed, published by Addaly, 2026.
Author and editor: **Dr Mustafa Mun**.

The manuscript was drafted with the assistance of a large language model and was edited and published under his name. That is stated plainly here rather than left to be discovered, because a reader is entitled to know how a book they are trusting was made.

This book is the printed form of the course of the same name at addaly.com/learn/vector-and-print. The website is the living version and is corrected first; where the two differ, the website is right.

Free to read, free to download, free to print, and free to teach from. There is no paid edition and there never will be. If you paid for this, somebody has sold you something that was given away.

Every diagram in it is drawn from data by code, not generated as a picture, so the numbers on the page are the numbers in the argument. Answers to every exercise, test and examination question are at the back, with a note on why each wrong option attracts people — those notes are the teaching, not the marking.

Contents

1. The thing that is actually in the file

Read the geometry inside a vector file by hand, predict which operations are lossless and which silently rasterise, explain a size or line-weight discrepancy by naming the unit or rendering rule that caused it, and choose vector or raster for a job from its output device rather than from habit.

1. Why your Photoshop logo cannot go on a hoarding
2. What the `d` attribute is saying
3. Points, millimetres, and two different inches
4. A path has no thickness
5. Why the hole will not appear
6. Clipping hides; it does not remove
7. Where vector quietly becomes pixels
8. What auto-trace is doing, and when to redraw instead
9. Case study: The hoarding, the WhatsApp logo and the three-day deadline
10. Module test

2. Drawing curves people will pay for

Draw, correct and combine any shape with the pen, node, boolean, offset and transform tools, name the node type, handle or continuity failure responsible when a curve goes lumpy, and rebuild an existing mark from a photograph or scan as exact geometry rather than as an approximation.

1. Your curves are lumpy because you used too many points
2. Most good logos are circles and rectangles cut into each other
3. The pen tool is four gestures
4. The kink you can see but cannot find
5. Booleans, and the mess they leave behind
6. Offsets, and the spike that appears at a sharp corner
7. Transforms are arithmetic, so type the numbers
8. Grids and snapping that help instead of fighting
9. Rebuilding a mark that only exists as a photograph
10. Case study: The star-shaped label and the twenty-six millimetre spike
11. Module test

3. Type, which is also vector

Set, space and convert type in a vector file without breaking it — choose a typeface you are licensed to use for the job in hand, correct spacing at the pair and the paragraph level, handle a complex script correctly, and decide per job whether to outline, embed or subset, stating the consequence of each to the printer.

1. Why the printer asked you to outline the fonts
2. What is actually inside a font file
3. The licence, which is the part that bites
4. Kerning a wordmark by hand
5. Setting a paragraph somebody can read
6. Type on a curve, and why it splays
7. Devanagari, Arabic, and the scripts that break
8. Figures, dashes and the characters nobody checks
9. Variable fonts, and what a press does with one
10. Case study: The consent form, the Friday deadline and the reader who was away
11. Module test

4. Colour that survives the press

Specify a colour so the sheet that comes off the press matches what was agreed — choose between process, spot and duotone, read a gamut warning and pick a rendering intent, allow for dot gain and trapping, and state which of screen, proof and press you are trusting and why.

1. Two different physical processes
2. Lab, Delta E, and what "close enough" means
3. Profiles, and the dialog that decides everything
4. What happens to a colour the press cannot print
5. Why your 20% tint printed at 30%
6. A spot colour is a plate, and a plate is money
7. Getting a whole design out of two inks
8. Trapping, and the white line around everything
9. Proofing when you cannot afford a proofing system
10. Case study: The grape carton, the fifth plate and two printers in two cities
11. Module test

5. Designing a mark

Take a logo from brief to delivered asset pack — interrogate the brief against the surfaces it will appear on, choose and draw a mark that holds in one colour at 16 pixels and on a hostile background, define its lockups and clear space, and hand over files plus a usage sheet a stranger can work from.

1. If it does not work in black at 16 pixels, it is not a logo
2. The questions to ask before you draw anything
3. Five kinds of mark, and what each one costs
4. Thumbnails, and why you never show a concept you would not ship
5. Real construction, and the golden-ratio story
6. Lockups, clear space and the smallest it may ever be
7. Reversed, on a photograph, and in one thread
8. Names, trademarks and who owns the file
9. The folder you hand over
10. Case study: The peacock the uniforms could not hold
11. Module test

6. Documents that survive a late change

Build a multi-page document that survives a late change — masters, a style family, a grid derived from the body text, linked images at verified effective resolution, an automatic contents list and a data merge — and specify a bound job with the right page count, spine width and gutter allowance.

1. Eighty pages, one font change, six hours of your life
2. Choosing a layout tool, honestly
3. Deriving a grid from the type, not from a template
4. Masters, running heads and why folios drift
5. Threaded text, overset, and copy that does not fit
6. Linked images, and the number that actually matters
7. Tables, and why tabs are not one
8. Contents lists and cross-references that renumber themselves
9. Five hundred different cards from one template
10. Page counts, creep and the spine
11. Case study: Fourteen hundred certificates and a question mark in every name
12. Module test

7. The handover

Export a file a press will run without a phone call — bleed and boxes correct, the right PDF/X flavour, fonts and images accounted for — and verify every one of those claims from the exported PDF itself using free tools, then specify the stock and finishing and record the approval.

1. The blue you chose on screen cannot be printed
2. What a RIP actually does with your file
3. PDF/X, and what each flavour forbids
4. The five rectangles inside a PDF
5. Preflighting your own PDF, free
6. Reading separations, and the fifth plate you did not order
7. Paper, and everything done after printing
8. Approvals, and who pays for the reprint
9. CDR is the most common file in Indian print and the least portable
10. The covering note, and the file a shop can open
11. Case study: Six at night, plates at eight, and a die-line set to knock out
12. Module test

8. Machines that cut, stitch and burn

Prepare a vector file for a plotter, needle, laser, router, screen or large-format printer — stating the minimum feature the material can hold, the kerf, pull or choke the process introduces, and the file format, layer convention and spot-colour name the operator's software expects.

1. The paid vector work is mostly machines, not posters
2. A vinyl job from file to wall
3. Print and cut, and the marks the machine looks for
4. Embroidery, which does not use vectors at all
5. Laser cutting, kerf, and the material you must never use
6. CNC routing, tool radius, and the letters on a shop board
7. Screen printing, separations and the underbase
8. DTF, DTG, sublimation and heat transfer
9. Signage, viewing distance and the seams
10. Case study: Eighteen hundred shirts and a sew-out nobody wanted to pay for
11. Module test

Vectors, Logos and Things That Get Printed: final examination

The paper that opens the next course. Answers to everything follow it.

MODULE 1

The thing that is actually in the file

Before any drawing, the raw material. What a vector file stores, in what units, and how a renderer turns it back into something you can see. Path data read by hand, the coordinate system and the two different "dots per inch" that fight each other, strokes that are instructions rather than shapes, winding rules, clipping, and the places where vector artwork quietly becomes pixels without telling you.

> **BY THE END OF THIS MODULE YOU CAN**
>
> Read the geometry inside a vector file by hand, predict which operations are lossless and which silently rasterise, explain a size or line-weight discrepancy by naming the unit or rendering rule that caused it, and choose vector or raster for a job from its output device rather than from habit.

Why your Photoshop logo cannot go on a hoarding

THE ONE THING TO KEEP

A raster file records pixels at one fixed size; a vector file records geometry with no size at all, which is why only the second can be redrawn for a hoarding.

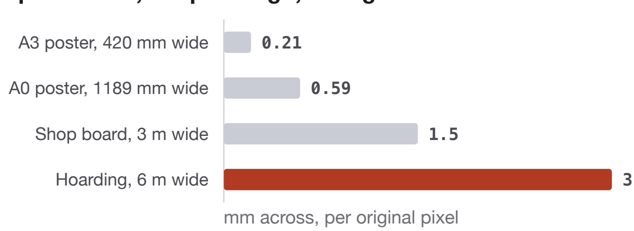
The phone call every beginner gets

You made a logo in Photoshop. It looked sharp. You sent it to the printer for a six-metre hoarding and got a message back: *send vector*. You sent the same PNG at 300 dpi, larger. Same message.

Nothing is wrong with your design. The file is the wrong kind of thing.

A raster file — PNG, JPG, PSD, a Photoshop document — stores a fixed grid of coloured squares. A 2000 × 2000 PNG stores exactly four million colour values and no other information about the picture. Enlarge it to six metres and each of those squares becomes about three millimetres wide. Software can guess what belongs between them, and modern upscalers guess well, but a guess is not the edge you drew. On a hoarding, an edge that was one pixel wide is now a soft three-millimetre smear.

One pixel of a 2,000-pixel logo, enlarged



A 2,000 x 2,000 PNG holds four million colour values and nothing else. Enlarging does not add any; it makes each one bigger. At six metres an edge that was one pixel wide is a three-millimetre smear, which is what the printer is looking at when they ask for vector.

A vector file stores something else entirely: instructions. Not "these pixels are black" but "a closed path starting here, curving to there with this tangent, filled with this colour". Nothing in that description has a size. When the file reaches the printer, the RIP — the software driving the press or the large-format machine — reads the instructions and draws them fresh at whatever resolution that machine actually has. Six metres or sixteen pixels, it is drawn cleanly both times.

What the instructions look like

You can read them. An SVG is plain text:

```
<svg viewBox="0 0 100 100" xmlns="http://www.w3.org/2000/svg">
  <circle cx="50" cy="50" r="40" fill="#c0392b"/>
  <path d="M35 50 L65 32 L65 68 Z" fill="#ffffff"/>
</svg>
```

That is a play button. It is about 200 bytes and it is sharp at any size, because there is no size in it. The same mark as a PNG large enough for a hoarding would be tens of megabytes and still soft.

Open any SVG on your machine in a text editor. Seeing the coordinates is the fastest way to stop thinking of vector as "a better kind of picture" and start thinking of it as a description of geometry.

Where vector is the wrong tool

This is not a hierarchy where vector wins. Vector is bad at photographs, and the failure is worth understanding because people meet it by accident.

Auto-trace — Image Trace in Illustrator, Path > Trace Bitmap in Inkscape, PowerTRACE in CorelDRAW — converts a raster into paths by finding colour boundaries. On a flat logo scanned from a business card, it works. On a photograph it produces thousands of overlapping blobs, because a photograph has no flat regions. A traced portrait can carry 60,000 nodes. The file is enormous, it will not redraw smoothly on your screen, and it can time out on the press's RIP. Print shops lose jobs to this every week.

So: photographs stay raster, and belong in image editing. Logos, type, diagrams, maps, icons, signage, packaging die-lines and anything that must be cut, engraved or stitched are vector.

Resolution independence also does not mean quality independence. A clumsy curve stays clumsy at six metres — it becomes a very large clumsy curve. Vector removes one failure mode. It does not remove drawing.

What to install, including if you have no money

- **CorelDRAW** is the tool of the Indian print trade, of signage shops, of garment and screen-printing units across South and Southeast Asia. If you want paid work at a local press, this is the file they will hand you. It is a paid product; Corel has sold both a subscription and a one-off perpetual licence, so check what is on offer now.
- **Adobe Illustrator** is what agency job ads ask for. Subscription only.
- **Inkscape** is free, open source, runs on Windows, macOS and Linux, and is a genuinely capable vector editor. It draws, edits nodes, does booleans, does type on a path, exports SVG, PDF, EPS and DXF. It is weaker on colour management for press and slower on very heavy files. Nothing else in this course is out of its reach.
- **Affinity Designer** was for years a one-off purchase, which made it the obvious escape from subscriptions. After Canva acquired Serif, the Affinity apps were relaunched in 2025 with free access and some features behind a paid tier — confirm the current terms before you plan around it.
- On a phone, real node editing is unpleasant on a small screen, but it is possible: Infinite Design on Android, Linearity Curve on iPhone. Use a phone to review a PDF and mark it up. Build on the largest screen you can reach, including a cyber café machine.

Install Inkscape today if you have nothing. The concepts in the next eight lessons are the same in all four programs; only the menus move.

One thing to check right now

Open the logo you are proudest of. If the only file you have is a PNG or a JPG, you do not have a logo yet — you have a picture of one. Redrawing it as vector is the work of the next four lessons, and it is the difference between a file a shop can print and a file they will email you back about.

BEFORE YOU MOVE ON

A client sends their logo as a 2000 × 2000 PNG saved at 300 dpi and asks for it on a six-metre hoarding. The print shop refuses the file. What is the actual problem?

A. The PNG is in RGB and the hoarding needs CMYK, so a colour-space conversion is what is blocking the job.

B. 300 dpi is below the resolution large-format printing requires, and the file should be re-exported at a higher dpi.

C. The file stores a fixed grid of four million pixels; enlarged to six metres each pixel becomes a visible block, and no resampling can add detail that was never recorded. A vector file would be redrawn at the machine's own resolution.

D. PNG compression discarded fine detail, so sending the same artwork as an uncompressed TIFF would give enough information to enlarge.

2 · 9 min

What the `d` attribute is saying

THE ONE THING TO KEEP

An SVG is text you can read and edit, and the `d` attribute is a short list of pen instructions in document coordinates — which is why a 240-byte logo and a 1.9 MB logo can look identical on screen.

Open it in a text editor

Right-click any SVG, open it with Notepad, TextEdit or `nano`, and look. There is no magic in there. There is a list of instructions.

```
<path d="M 20,80 C 20,20 80,20 80,80 Z" fill="#1d4ed8"/>
```

Read it out loud. **M**ove the pen to 20,80. Draw a **C**ubic curve whose first handle points to 20,20, whose second handle points to 80,20, ending at 80,80. **Z** closes the path back to the start.

That is the whole vocabulary you need, and there are only a handful of commands:

- `M x,y` — move without drawing. Starts a new subpath.
- `L x,y` — straight line to a point. `H x` and `V y` are the horizontal and vertical shorthands.
- `C x1,y1 x2,y2 x,y` — cubic Bézier: two handle positions, then the end point. This is by far the most common.
- `S x2,y2 x,y` — smooth cubic. Reuses the previous curve's outgoing handle, mirrored. This is how exporters compress a smooth run of curves.
- `Q` and `T` — quadratic Bézier, one handle. TrueType fonts are built from these; PostScript and SVG artwork mostly is not.

- `A rx,ry rot large-arc sweep x,y` — elliptical arc. Rarely written by hand, common in output from charting libraries.
- `Z` — close the subpath.

A capital letter means **absolute** coordinates. A lowercase letter means **relative** to where the pen already is. `l 10,0` means "ten units right of here". Exporters use lowercase because the numbers come out shorter.

The coordinate frame

Two attributes decide what those numbers mean.

```
<svg viewBox="0 0 100 100" width="40mm" height="40mm">
```

`viewBox` sets the internal coordinate system: this drawing is 100 units across. `width` and `height` say how large to render it. Change the width to `400mm` and nothing inside the file changes — the same 100 units are simply stretched over a larger area. **This is the whole of resolution independence, in one attribute.**

If `width` and `height` are missing, the renderer uses the viewBox and falls back to 96 units per inch. That default is the cause of a great deal of confusion, and it is the next lesson.

Why two identical-looking logos differ by 8,000×

Export the same mark two ways and you will see something like this:

```
play-drawn.svg        240 bytes    3 path elements
play-traced.svg    1,912,406 bytes  412 path elements
```

Three things inflate a file, and all three are visible in the text:

1. **Node count.** A traced logo has hundreds of anchor points where a drawn one has eight. Every anchor is six numbers.

2. **Decimal precision.** Illustrator and Inkscape happily write `47.238471950000003`. Fifteen significant figures on a drawing that will be cut at half-millimetre accuracy is sixty bytes of nothing.
3. **Embedded rasters.** An `<image xlink:href="data:image/png;base64,...">` means somebody placed a photograph inside what they called a vector file. This is the single most common reason a "vector logo" fails when the printer enlarges it.

Search the file for `base64`. If you find it, you do not have vector artwork.

Cleaning up, free

`svgo` and `scour` both strip precision, merge paths and remove editor metadata:

```
npx svgo --precision=2 logo.svg -o logo.min.svg
```

Two decimal places on a 100-unit viewBox is accuracy to one hundredth of the drawing's width. Nothing you print can hold more.

The part nobody tells you: batch editing

Because it is text, you can do things no menu offers. A client changes their brand blue and you have forty icons:

```
sed -i 's/#1d4ed8/#0f766e/g' icons/*.svg
```

Forty files, one second, no application open. You can also grep an entire folder for `base64` to find the fake vectors, or for `font-family` to find the files where somebody forgot to outline the type.

This is not a trick for programmers. It is the single most useful thing about the format, and it is why SVG is also the only vector format that diffs sensibly in version control — you can see, line by line, what changed between two versions of a drawing.

Watch out for `transform`

A path can sit inside nested groups, each with its own `transform="translate(...) scale(...) rotate(...)"`. The numbers in `d` are then not the numbers on your ruler. If coordinates look wrong, look upward at the groups. Most editors have an "ungroup and bake transforms" step; in Inkscape, Edit ▸ Preferences ▸ Behaviour ▸ Transforms ▸ *Store transformation: Optimized* pushes transforms down into the path data where you can read them.

BEFORE YOU MOVE ON

Two SVG files of the same simple logo look identical on screen. One is 240 bytes; the other is 1.9 MB. Opening the large one in a text editor shows 412 `<path>` elements and coordinates written to fifteen decimal places. What does that tell you about the large file, and what will go wrong with it?

A. It was saved at a higher resolution setting, so it will print more sharply but will take much longer to render on the press.

B. It is uncompressed, so gzipping it will make the two files equivalent.

C. It was auto-traced, so it carries hundreds of redundant anchors that will wobble when enlarged or cut.

D. It contains CMYK colour values, which are longer than hex and cannot be compressed.

3 · 9 min

Points, millimetres, and two different inches

THE ONE THING TO KEEP

A vector file stores unitless numbers plus a rule for how many of them make an inch — and because PostScript says 72 and the web says 96, a file that does not state its units can open at the wrong size in perfectly working software.

The two inches

There are two competing definitions of a "unit" in graphics, and both are correct in their own world.

- **The PostScript point.** 72 points to the inch. This is the unit of typesetting, of PDF, and of everything that goes to a press. A 12 pt type size is one sixth of an inch of em, and a 3 mm bleed is 8.504 pt.
- **The CSS pixel.** 96 to the inch. This is the unit of the browser and therefore of SVG when no units are given.

Neither has anything to do with the dots on a printer. **A vector file has no dpi at all.** When somebody asks for a "300 dpi logo", they are asking a question with no answer; the answer they need is a finished size in millimetres.

The arithmetic that catches everyone

Inkscape before version 0.92 used 90 units per inch. From 0.92 onwards it uses 96, to match the SVG standard. Old files that stored bare numbers with no unit suffix were therefore reinterpreted.

Draw a 100 mm square in Inkscape 0.91. The file stores:

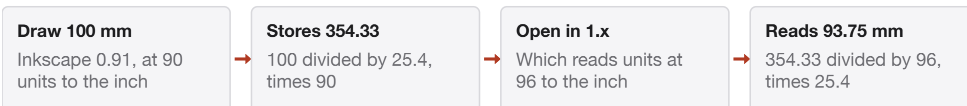
```
100 mm ÷ 25.4 mm/inch × 90 units/inch = 354.33 units
```

Open it in Inkscape 1.x, which reads those same 354.33 units at 96 to the inch:

```
354.33 ÷ 96 × 25.4 = 93.75 mm
```

The square is now 93.75 mm. Nothing is corrupt. Two programs simply disagreed about what a unit is, and the file never said. Inkscape 1.x offers a scaling fix on open for exactly this reason, and if you inherit old artwork you should measure a known dimension before trusting anything.

A 100 mm square that opens at 93.75 mm



Nothing is corrupt at any stage. The file stored a bare number and never said what a unit was, so two programs answered the question differently. Measure one known edge after every import.

The same class of error explains why an EPS from a 1990s archive sometimes opens at 133% in a modern application, and why a DXF for a cutting machine can arrive in inches when the shop works in millimetres. **Always measure one known edge after importing.**

Set the document up properly, once

Decide the unit from the output, not the software default.

- **Print of any kind** — millimetres. Set the document units to mm and the page to trim size. A visiting card is 90 × 54 mm in India, 85 × 55 mm in most of Europe, 89 × 51 mm in the US. Ask; do not assume.
- **Screen and icons** — pixels, and a viewBox that is a round number.
- **Signage and cutting** — millimetres, and build at 1:1 if the machine can take it. Large-format software usually can. If you must work at a scale, put it in the filename: `hoarding-1-to-10.svg`.

Where to set it:

- **Inkscape**: File ▸ Document Properties. Set *Display units* and the page size, and check *Scale* is 1.
- **Illustrator**: File ▸ Document Setup ▸ Units, and Edit ▸ Preferences ▸ Units for the rulers.

- **CorelDRAW**: the units dropdown on the property bar with nothing selected.
- **Scribus**: File ▸ Document Setup ▸ Document ▸ Units.

Stroke weights live in the same trap

A 1 pt rule is 0.353 mm. A 1 px rule at 96 dpi is 0.265 mm. A 0.5 pt rule is 0.176 mm, which is about as thin as offset litho reliably holds. If you draw in pixels and print in millimetres, every hairline in the job is 25% thinner than you think it is.

Get in the habit of quoting line weights in millimetres when you talk to a press. "Point-two-five millimetre keyline" is unambiguous. "Half a point" invites the question "half of whose point?"

The one number worth memorising

25.4. Millimetres to the inch. Every other conversion in this subject is that number with a 72 or a 96 beside it:

- mm → pt: `× 72 ÷ 25.4` (so 1 mm = 2.835 pt)
- mm → px at 96: `× 96 ÷ 25.4` (so 1 mm = 3.780 px)
- pt → px: `× 96 ÷ 72` (so 1 pt = 1.333 px)

A 3 mm bleed is 8.5 pt or 11.34 px. An A4 page is 210 × 297 mm, 595 × 842 pt, 794 × 1123 px. You will use these enough to stop looking them up.

One more habit worth forming: when a client or a shop sends a size, write back with the millimetres. "You said A5 — that is 148 × 210 mm trim, and I will supply 154 × 216 with 3 mm bleed." Half the size disputes in print are two people using the same word for different measurements, and one sentence settles it before any drawing happens.

BEFORE YOU MOVE ON

A 100 mm square drawn in Inkscape 0.91 is opened in Inkscape 1.x and now measures 93.75 mm. The file is not damaged and both programs are behaving correctly. What happened?

A. The older version saved in points and the newer one reads millimetres, so the square was converted using 72 rather than 25.4.
B. The file stored 354.33 bare units, which 0.91 wrote at 90 per inch and 1.x reads at 96 per inch.
C. Inkscape 1.x defaults to a 93.75% document scale, which can be reset in Document Properties.
D. SVG cannot store millimetres, so any physical dimension is approximated and drifts slightly on every save.

4 · 9 min

A path has no thickness

THE ONE THING TO KEEP

A stroke is a render-time instruction, not an object, so its caps, joins, miter limit and scaling behaviour can change the printed shape — and a "hairline" means whatever the output device's thinnest line is, which on a platesetter is thin enough to disappear.

The line you drew does not exist

Draw a line in any vector program and give it a 2 mm stroke. You now believe there is a 2 mm-wide object on the page. There is not. There is a mathematical path — a curve of zero width — and an instruction attached to it that says *when rendering, sweep a 2 mm-wide brush along me*.

Nothing downstream is obliged to obey that instruction the way you expect, and the places where it goes wrong are the places this lesson exists for.

The parameters that actually change the shape

A stroke carries more than a width:

- **Cap** — what happens at an open end. `butt` stops exactly at the endpoint. `round` and `square` extend it by half the stroke width beyond. A 10 mm line with a 4 mm round cap measures 14 mm. This is why a dimension you carefully set comes out long.
- **Join** — what happens at a corner. `miter` extends both edges until they meet, `round` arcs across, `bevel` cuts the corner flat.
- **Miter limit** — the escape hatch on the above. At a sharp angle, a mitred join produces a spike that can run far beyond the corner; the limit is the ratio at which the renderer gives up and bevels instead. SVG and PDF default to 4, which cuts in at about 29°. A logo with a very sharp point can therefore render with a spike in one program and a flat corner in another, and both are following the specification.
- **Dash array** — a list of on/off lengths, in stroke units, plus an offset. Dashes are generated at render time; they are not objects.
- **Alignment** — centre, inside or outside. Illustrator, Affinity and CorelDRAW offer all three. **SVG has only centre**, so an inside-aligned stroke exported to SVG shifts by half its width. Inkscape approximates the other two with a Live Path Effect, which does not survive export to most other programs.

Hairlines, and the line that vanishes on press

CorelDRAW and several plotter drivers offer a width called **hairline**. It is not a measurement. It means *the thinnest line this output device can produce*.

- On a 600 dpi office laser, that is 1/600 inch = **0.042 mm**. Visible, slightly weak, fine.
- On a 2400 dpi platesetter, that is 1/2400 inch = **0.011 mm**. That is thinner than a press can hold ink. The line breaks into dots, or disappears.

So a hairline keyline proofs perfectly on the printer beside your desk and is missing from five thousand cartons. It is one of the classic print failures, and the mechanism is entirely in the definition of the word.

Use real numbers instead. Practical minimums, worth asking your supplier to confirm:

- Offset litho, positive line on white: **0.1 mm** (about 0.25 pt)
- Offset litho, reversed out of a solid: **0.15 mm**, because ink spreads into the gap
- Digital toner and inkjet: around **0.08 mm**
- Screen print: **0.3 mm**, coarser on textile
- Vinyl cutting: **1–2 mm**, because a human has to weed it
- Embroidery: **1.2 mm** minimum satin width

Scale strokes, or do not

Every program has a preference deciding whether a stroke scales with the object. Both behaviours are useful and the wrong one is expensive.

- **On** is what you want for a logo. Shrink the mark to 10 mm and the outline shrinks with it.
- **Off** is what you want for a technical drawing or a die-line, where 0.25 mm means 0.25 mm at any zoom.

Illustrator: Preferences ▸ General ▸ *Scale Strokes & Effects*. Inkscape: the four toggle buttons at the right end of the selection tool's control bar — the first one governs stroke width. CorelDRAW: *Scale outline with object* in Object Properties.

The failure is always the same direction. Someone designs a mark at 200 mm with a 2 mm outline, scales it to a 15 mm badge with scaling off, and the outline is now proportionally thirteen times too heavy. It looks like a blob and nobody can say why.

When to make the stroke real

`Stroke to path` converts the instruction into geometry: a closed outline you can boolean, cut, engrave or stitch. It is destructive — the centreline is gone, and so is the ability to change the width by typing a number.

Do it when the file is leaving for a machine, and keep the editable original. Do not do it early, because every later size change then means redrawing.

BEFORE YOU MOVE ON

A carton design proofs correctly on a 600 dpi office laser, with a visible thin keyline around each panel. Five thousand printed cartons come back with no keyline at all. The artwork was set to CorelDRAW's "hairline" width. What happened?

A. The press ran out of registration and the keyline fell outside the trim on every sheet.
B. Hairline means the device's thinnest line: 0.042 mm on the laser, 0.011 mm on the platesetter, below what a press can hold.
C. Hairline is 0.25 pt, and 0.25 pt is below the 0.35 pt minimum for offset litho.
D. The keyline was a stroke rather than an outlined object, and strokes are discarded whenever a PDF is flattened for press output.

---

5 · 9 min

Why the hole will not appear

THE ONE THING TO KEEP

A shape with more than one loop needs a rule for what counts as inside: non-zero winding depends on the direction each subpath is drawn, even-odd only on how many crossings there are, and the same geometry prints differently under each.

One path, several loops

A path can contain more than one subpath — the letter "O" is two loops, a die-line with a window is two loops, a five-pointed star drawn in one continuous stroke is one loop that crosses itself. In every case the renderer has to answer a question that is genuinely not obvious: **for any point on the page, is it inside the shape or outside it?**

There are two answers in general use, and they disagree.

Non-zero winding

Fire an imaginary ray from the point out to infinity. Every time the ray crosses a subpath, add 1 if that subpath is running left-to-right across the ray, subtract 1 if it is running right-to-left. If the total is anything other than zero, the point is inside.

The word doing the work is **direction**. Two circles drawn the same way — both clockwise — give a winding number of 2 in the inner region, which is not zero, so it fills. You get a disc. Reverse the inner circle and the counts cancel to zero: a hole.

Even-odd

Count crossings and ignore direction. Odd is inside, even is outside. Two nested circles always give 2 crossings in the middle, which is even, which is outside. You always get a hole.

Two circles, one inside the other

	Non-zero	Even-odd
Same direction	**Solid disc** winding number 2	**Hole** two crossings
Inner reversed	**Hole** winding number 0	**Hole** two crossings

Non-zero winding depends on the direction each subpath is drawn; even-odd only counts crossings. Same geometry, two printed results, neither program broken. Reversing the inner loop is the portable fix, because direction travels with the file and a fill-rule attribute can be dropped on import.

Why this matters more than it sounds

Take a five-pointed star drawn as one self-intersecting path — the one you draw without lifting the pen. The central pentagon has a winding number of 2.

- Under **non-zero**, 2 ≠ 0, so the pentagon fills. You get a solid star.
- Under **even-odd**, 2 crossings is even, so the pentagon is outside. You get a star with a pentagonal hole in the middle.

Same geometry, same file, two completely different printed results. Neither program is broken.

Both SVG and PDF default to non-zero. In PDF the two operators are literally `f` and `f*`; in SVG it is `fill-rule="nonzero"` or `"evenodd"`. Illustrator's default is non-zero; Inkscape's default is non-zero; CorelDRAW's is even-odd for some legacy objects, which is one reason CDR imports fill strangely.

The fixes, in each program

- **Reverse the inner subpath.** Inkscape: select the node-tool, click a node on the inner loop, Path ▸ Reverse (**Shift+R**). Illustrator: Attributes panel ▸ Reverse Path Direction. CorelDRAW: Shape tool, select a node on the subpath, then the reverse button on the property bar.
- **Switch the object to even-odd.** Inkscape: Fill & Stroke ▸ the two small buttons beside the fill swatch. Illustrator: Attributes panel ▸ Even-Odd Fill Rule.

Reversing is the more portable fix, because direction is stored in the geometry and travels with the file. The fill rule is an attribute that some importers drop.

See the direction

You cannot fix what you cannot see. Turn on direction markers:

- Inkscape: node tool, then Edit ▸ Preferences ▸ Tools ▸ Node ▸ show path direction on outlines.
- Illustrator: View ▸ Show Edges plus the Document Info panel, or simply select the path and look at where the start point sits.

- Any program: apply a dashed stroke with an arrow marker at the end. Crude and instant.

Compound path, group, clip — three different things

People use these interchangeably and they are not.

- A **compound path** is *one object* made of several subpaths, sharing one fill and one stroke, and subject to a winding rule. This is what a letter "O" or a die-line window is. Make one: **Ctrl+8** in Illustrator, Path ▸ Combine (**Ctrl+K**) in Inkscape, Object ▸ Combine (**Ctrl+L**) in CorelDRAW.
- A **group** is several independent objects moved together. Each keeps its own fill. Grouping two circles never produces a hole.
- A **clip** hides everything outside a shape. It does not remove it — which is the subject of the next lesson.

The practical consequence: if you send a die-line to a cutting machine as a *group* of two circles, the machine cuts both outlines and the middle falls out. As a *compound path* with the inner loop reversed, it is one object with a hole, and the operator's software reads it as one closed cut with an interior.

Where you will meet this again

A logo with counters that fill solid after import. A map where the lakes print as land. An envelope die-line whose window prints as a block. A vinyl cut where the inside of every "e" is cut as a separate piece. All the same rule.

BEFORE YOU MOVE ON

A five-pointed star is drawn as a single path without lifting the pen, so the path crosses itself and the central pentagon has a winding number of 2. It renders as a solid star in one application and as a star with a pentagonal hole in another. Why?

A. The second application imported the path as a group rather than a compound path, so the inner region became a separate object.
B. The path was left open in the second application, and open paths fill only their outermost loop.
C. One uses non-zero winding, where 2 counts as inside, and the other uses even-odd, where 2 crossings count as outside.
D. The second application converted the path to even-odd because the subpaths run in opposite directions.

6 · 8 min

Clipping hides; it does not remove

THE ONE THING TO KEEP

A clipping path hides geometry rather than deleting it, so hidden image data still bloats the file, still costs RIP time, and can still be recovered by anyone who opens the PDF.

The crop that is not a crop

Place a photograph in a layout, drag the frame in until only a face shows, export a PDF. Most people believe the rest of the photograph is gone. It is not. The frame is a **clipping path**: an instruction that says *render only the part of this object that falls inside this shape*. The object itself — every pixel of the original photograph, at its original size — is still in the file.

Three consequences follow, and the third one has ended careers.

File size. A 40 MB photograph cropped to a thumbnail is still 40 MB in the exported PDF unless you explicitly downsample or crop-on-export. This is why a twelve-page brochure ships at 480 MB.

RIP time. The press's raster image processor decodes the whole image and then throws most of it away, on every page, every time.

Recovery. Anyone with a PDF tool can release the clip and see what you hid. Newspapers have published redacted documents where the black rectangle was an object drawn *over* the text rather than the text being removed; the text was extractable with copy-and-paste. The same mechanism applies to a face cropped out of a photo, a price list with the competitor's column covered, a contract with a name masked.

> *If you need something to be gone, delete the pixels in a raster editor and place the edited file. A clip, a mask, a white box and a black box are all just drawings on top.*

Clip versus mask

- A **clip path** is geometric and binary: inside is shown, outside is not. Hard edges only.
- A **mask** uses a greyscale or luminance image: white shows, black hides, grey is partially transparent. This is how a soft fade into a background is made.

Applying them:

- **Inkscape**: select the object and the shape, Object ▸ Clip ▸ Set, or Object ▸ Mask ▸ Set. Release with the same menu.
- **Illustrator**: **Ctrl+7** for a clipping mask; the Transparency panel's *Make Mask* for an opacity mask.
- **CorelDRAW**: Object ▸ PowerClip ▸ Place Inside Frame. Extract contents to release.
- **Scribus**: an image frame is itself the clip.

CorelDRAW's PowerClip is worth a separate warning: it nests happily, and a file can end up with PowerClips inside PowerClips several levels deep, each one carrying full-size contents. A file that opens slowly for no visible reason usually has this.

Transparency groups and the knockout you did not ask for

Opacity has two scopes and the difference is visible.

- **Object opacity** applies to each object separately. Two overlapping 50% circles show a darker patch where they cross, because the second is composited onto the first.
- **Group opacity** composites the group internally at full strength, then fades the whole result. The overlap disappears.

That is the setting people hunt for when a logo made of two overlapping shapes develops a visible seam at 50% opacity. In Illustrator it is the Transparency panel's *Isolate Blending*; in SVG it is putting the opacity on the `<g>` rather than on each `<path>`.

A **knockout group** goes further: inside the group, later objects punch through earlier ones rather than blending with them. It is rarely needed and, when set by accident, produces artwork that looks correct on screen and prints with holes in it.

Before the file leaves

Three checks, thirty seconds:

1. **Release clips on anything going to a machine.** A cutting plotter or laser follows paths, and a clipped path still carries its hidden geometry — the blade will happily cut the part you thought was cropped away.
2. **Crop images properly for press.** Most PDF export dialogs have a *crop image data to frames* option. Turn it on and watch the file size fall. In Acrobat Pro the equivalent is the PDF Optimizer's *Discard cropped image data*; Ghostscript will not do it for you.
3. **Never use a clip, a mask or a covering rectangle for redaction.** Flatten to raster, or remove the content at source. If you must prove something is gone, export the page to a 300 ppi image, check it, and place that.

A quick way to see what a PDF is really carrying: `pdfimages -list job.pdf` lists every embedded image with its true pixel dimensions. If a picture you cropped to a 20 mm square is listed at 6000 × 4000 pixels, the crop was a clip and the rest of the photograph is still travelling with the file.

BEFORE YOU MOVE ON

A local paper crops a photograph in its layout so a bystander's face falls outside the picture frame, then exports a press PDF and posts the same PDF on its website. A reader recovers the face. What is the mechanism?

A. The PDF export used a low compression setting, so the discarded pixels were reconstructed by the reader's viewer.

B. The image frame is a clipping path, so the whole original image is still embedded and reappears when the clip is released.

C. PDF stores a thumbnail preview of every placed image at original dimensions, which viewers can extract.

D. The clip was applied to the frame rather than the image, and applying it to the image instead would have deleted the surplus pixels.

7 · 10 min

Where vector quietly becomes pixels

THE ONE THING TO KEEP

Shadows, blurs, meshes and blend modes are generated as pixels at a resolution the document sets — often 72 ppi by default — and anything transparent gets sliced into atomic regions when flattened, which is where the grey box and the hairline seam come from.

The effects that are not vector at all

Some things in a vector application are geometry. Some are pictures generated on the fly. The second group has a resolution, and if you do not set it, the software picks one for you — often badly.

Raster by nature, in every program:

- Drop shadow, outer glow, inner glow
- Gaussian blur and feather
- Most texture, grain and distortion filters
- **Gradient mesh** (Illustrator) and mesh fills (Inkscape, CorelDRAW)
- Anything under Illustrator's Effect ▸ Photoshop menu

Compositing, which is not raster but needs flattening in older workflows:

- Any object below 100% opacity
- Blend modes: multiply, screen, overlay, soft light
- Transparency inside a placed PDF or PSD

The 72 ppi trap

Illustrator has a document setting called **Raster Effects Resolution**. For years the default on a new document was *Screen (72 ppi)*. Every drop shadow, every feather, every blur in that document is generated at 72 ppi.

You will not see it. On screen at 100% it looks perfect, because your screen is roughly 72–110 ppi. It prints as a visibly chunky, stepped shadow.

Fix it on day one: Effect ▸ Document Raster Effects Settings ▸ **300 ppi** for print, and check it again whenever you open somebody else's file. Inkscape's equivalent is in Document Properties (default export dpi and filter quality); CorelDRAW sets it per-effect in the bitmap resolution dropdown.

What flattening actually does

A press workflow that cannot handle live transparency — PDF/X-1a, EPS, PostScript, older RIPs — must **flatten** it. The flattener does something surprising:

1. It finds every region where a transparent object overlaps something.
2. It divides the artwork into *atomic regions* — areas that can each be described with a single opaque appearance.
3. Regions that can be expressed as vectors stay vector. Regions that cannot — because a gradient is blending through a blur, say — are rasterised at the flattener's chosen resolution.
4. It reassembles the page from those pieces.

Two well-known artefacts come straight out of step 3 and 4.

The grey or white box behind a logo. The rasterised region is rectangular, it carries the background it was composited against, and if the background in the file is not the background on the paper, the rectangle shows.

The hairline seam. Two adjacent atomic regions share an edge. One is drawn as vector, one as a raster tile. Their anti-aliasing does not agree exactly along that edge, so a light line one pixel wide appears across your artwork — visible on screen at some zoom levels and, worse, sometimes on the printed sheet.

Illustrator's **Flattener Preview** panel (Window ▸ Flattener Preview) shows you which regions will rasterise before you export. Acrobat Pro's Output Preview has a transparency flattening view. Both are worth a look before a big job.

How to avoid the whole problem

- **Export PDF/X-4 where the shop can take it.** X-4 keeps transparency live all the way to the RIP, which does the flattening once with full knowledge of the device. No atomic regions in your file, no seams.
- **Ask.** "Do you want X-1a or X-4?" is one line in an email and it settles this.
- **Build flat where you can.** A shadow made from a solid shape at a lower tint is not transparency and never flattens. A "gradient" built as five flat steps in a screen-printed job is a deliberate decision, not a limitation.
- **Rasterise on purpose.** If an element genuinely needs a blur, rasterise it yourself at 300 ppi with a known background, and place it. You now control the resolution and the edge rather than discovering them later.

The honest limitation

Resolution independence is a property of *geometry*, not of the file extension. An `.ai` file whose hero image is a gradient mesh with a Gaussian blur over it is, for practical purposes, a raster file in a vector wrapper. It will not enlarge cleanly and it will choke a plotter.

The test is not what the file is called. It is what is inside: select everything, look at the Document Info or Objects panel, and see whether the count is paths or effects.

BEFORE YOU MOVE ON

A logo with a soft drop shadow is exported as PDF/X-1a. On press, a faint rectangle of slightly different white appears around the mark, and a thin light line runs across the shadow. What produced these two artefacts?

A. The shadow was drawn at 72 ppi, so it was upsampled by the press RIP, and the interpolation produced both the rectangle and the line running across it.
B. PDF/X-1a converts RGB to CMYK, and the shadow's alpha channel was converted into a light grey fill.
C. The flattener sliced the page into atomic regions, rasterised those containing the shadow, and left mismatched edges between them.
D. The press used a coarser line screen than the proof, so the shadow's halftone dots grouped into visible bands.

8 · 9 min

What auto-trace is doing, and when to redraw instead

THE ONE THING TO KEEP

Auto-trace thresholds, finds contours on a pixel staircase, guesses which steps are corners and fits curves within a tolerance — so wobbly straights, rounded corners and node explosions are all predictable consequences, and anything being enlarged or cut should be redrawn over the trace rather than shipped as one.

The algorithm, so you can predict it

Every tracer — Inkscape's Trace Bitmap, Illustrator's Image Trace, CorelDRAW's PowerTRACE, the `potrace` command line, and the paid web services — runs roughly the same four steps.

1. **Reduce to regions.** For a black-and-white trace this is a single brightness threshold. For a colour trace it is a quantisation to N colours.
2. **Find contours.** Walk the boundary between regions. The result is a polygon whose vertices sit on pixel corners, so a diagonal edge comes out as a staircase.
3. **Detect corners.** Decide which staircase steps are a genuine corner and which are sampling noise. This is a tolerance you can set.
4. **Fit curves.** Replace each run of steps between corners with Bézier segments, within an error tolerance.

Once you know this, the failures stop being mysterious.

- **Wobbly straight edges** come from step 4 smoothing a staircase that was actually a straight line. The fix is a higher resolution scan, not a different setting.
- **Rounded corners on a square logo** come from step 3 deciding your corner was noise. Lower the corner threshold.
- **Hundreds of nodes** come from a tight error tolerance in step 4 plus a noisy scan.
- **Blobs everywhere on a photograph** come from step 1: a photograph has no flat regions to find.

The settings that actually matter

In Inkscape (Path ▸ Trace Bitmap):

- **Brightness cutoff** for line art. Try 0.45, then move it and watch the preview.
- **Speckles** — discards features below N pixels. Set it to about 4 for a clean scan, 20 for a photographed business card.
- **Smooth corners** — lower keeps corners sharp.
- **Optimise** — merges adjacent curve segments. Higher means fewer nodes and more drift from the original.
- **Multiple scans ▸ Colours** for a flat colour logo. Tick **Stack scans** and **Remove background**.

Two things about multi-colour trace nobody mentions. The output is a **stack of opaque shapes**, largest at the back, each partly hidden by the one in front — not a jigsaw of adjacent regions. So recolouring the top shape reveals the one beneath rather than changing the visible colour, and deleting a "background" shape puts holes in everything. And the shapes overlap by a pixel or two, which is invisible on screen and shows as a doubled edge on a cutting plotter.

Centreline tracing is a different job

A hand-drawn signature, a technical line drawing, a map: these are *lines*, not filled regions. A normal trace gives you the outline of each stroke — two paths where you wanted one. Centreline tracing gives you a single path down the middle, which is what a plotter pen, an engraver or a CNC router wants.

Inkscape 1.x has *Centreline tracing (autotrace)* in the same dialog. On the command line, `autotrace –centerline` does it. Illustrator has no true centreline mode; the workaround is to trace normally and then use Object ▸ Path ▸ *Simplify* on the result, which is not the same thing and usually worse.

Preparing the scan is most of the win

Ten minutes in GIMP or Photopea before tracing beats an hour of settings afterwards.

1. Scan at **600 dpi minimum**, 1200 for fine line work. Photograph only if you must, and flatten the lighting.
2. **Levels** — drag the black and white points in until the background is genuinely white and the ink genuinely black.
3. **Despeckle** or a small median filter to kill paper texture.
4. **Threshold** to pure black and white yourself, so you control step 1 rather than the tracer.

When to redraw instead

Trace, and accept it, when: the artwork is a one-off, it will be used at roughly the size it was scanned, and nobody will cut, stitch or enlarge it. A signature on a certificate. A rubber-stamp texture. A decorative border.

Redraw, always, when: the mark is a logo, anything goes to a plotter, laser or embroidery machine, or the artwork will be enlarged. Redrawing a typical mark from a clean trace takes 30–60 minutes with primitives and the pen tool, and the difference is measurable — eight to fifty nodes instead of two hundred to four hundred, straight edges that are straight, and circles that are circles.

Use the trace as a locked template layer at 30% opacity and draw over it. That is the professional workflow, and it is the same one whether you paid for your software or not.

The paid vectorisers, honestly

Vector Magic, Vectorizer.AI, Adobe's and Recraft's tools produce noticeably better curve fitting than potrace on difficult sources, and they are genuinely faster. They do not change the underlying truth: the output has more nodes than a drawn mark, the corners are inferred rather than intended, and a plotter will still cut every wobble. Treat them as better tracers, not as a substitute for drawing.

BEFORE YOU MOVE ON

A shop logo is traced from a 300 dpi scan in Inkscape. The result looks acceptable at 100% on screen, but when cut in vinyl at 600 mm wide the long straight edges of the letterforms have a visible gentle waviness. Which step of the tracing pipeline produced that, and what actually fixes it?

A. Curve fitting smoothed a straight edge's pixel staircase into shallow Béziers; a better scan and a redraw fix it.

B. The speckle setting was too low, so paper noise was traced as tiny features along each edge.

C. The vinyl plotter's blade offset was set wrong, which adds a small periodic deviation along every long straight cut.

D. The trace was done in RGB rather than greyscale, so the threshold varied slightly along each edge.

CASE STUDY · MODULE 1

The hoarding, the WhatsApp logo and the three-day deadline

A four-person signage shop on the Mumbai road at Nashik, four days before a jewellery showroom opens.

Sundeep runs a signage shop with three staff and a small CNC router. On Wednesday afternoon a jewellery showroom books him for an opening on Sunday: a six-metre by three-metre flex hoarding for the approach road, and 1.2 metre acrylic letters for the fascia above the shutter.

He asks for the logo. What arrives on WhatsApp is a JPEG, 480 pixels wide, forwarded twice. The owner says it was made in 2019 by his nephew, who has since moved to Dubai. There is no other file, nobody knows what typeface the name is set in, and the owner is confident he has supplied a logo, because he can see it on his phone.

Two jobs, two different problems.

The flex hoarding will accept a raster file, because the large-format RIP will take whatever it is given. At six metres wide, each of those 480 pixels becomes 12.5 mm across. Even at the generous 30 ppi a hoarding seen from thirty metres can get away with, the file would need to be about 7,000 pixels wide. It is fifteen times short. An upscaler will invent detail, and the detail it invents will be a guess about somebody else's curves.

The acrylic letters are worse, because the router does not read pixels at all. It follows paths. Somewhere between the JPEG and the machine, somebody has to produce geometry.

Sundeep has two routes and he has to choose on Wednesday evening.

Trace it. Twenty minutes in PowerTRACE. The output is roughly 2,400 nodes of contour found along a staircase of pixels, with the corners rounded where the tracer decided a corner was noise. On the hoarding the wobble on each letter is about a centimetre. On the routed letters it is worse, because the bit follows every wobble at a constant feed rate and the edge comes out rippled where it should be straight. The traced file also becomes the showroom's logo for the rest of its life, because it will be the only vector anyone has.

Redraw it. Identify the typeface, rebuild the diamond mark from two circles and a rotated square, and draw the wordmark over a locked, dimmed template. Sundeep has done this before and it takes him between two and three hours. Those hours are not billed: the client believes he has supplied artwork, and Sundeep has not raised the question of charging for it.

The cost of redrawing is not really the three hours. It is that the flex file then misses Thursday night. The printer he uses has one machine, Friday is already full with municipal election work, and the next clear slot is Saturday morning. If anything goes wrong on Saturday there is no float before the Sunday opening.

The cost of tracing is nothing on Wednesday and something every year afterwards.

Sundeep's brother-in-law, who does the installations, has a view: nobody stands close enough to a hoarding to see a wobble, the showroom is paying for a sign and not for archaeology, and Sunday is Sunday. That argument is half right, which is what makes it hard. At thirty metres the wobble genuinely does not read. At the fascia, where customers wait for their car with their nose a metre from the acrylic, it does.

There is a third consideration that neither of them raises on Wednesday. Whoever produces the vector will be the only person in Nashik who has it, and the showroom will need it again for a visiting card, a bill book, a shutter sticker and a box for every purchase.

STOP HERE

Before turning the page: what would you have done, and what would it have cost if you were wrong? Write it down. The point of the next section is to compare.

WHAT HAPPENED

Sundeep redrew it, and what decided him was one measurement rather than an argument. He printed a 300 mm crop of the traced capital S at full size on the office inkjet, taped it to the counter, and stood at the door. The wobble was visible from six feet, which is roughly where a customer stands.

The redraw took two hours forty minutes: the mark was two circles and a square rotated 45 degrees, and the wordmark turned out to be a widely available face with the R leg redrawn. He sent the flex file at eleven that night, kept the Thursday slot, and routed the letters on Friday afternoon.

He also put a line on the invoice: "Logo reconstruction from supplied image, 2.5 hours." Itemised, with the traced version and the redrawn version printed side by side and stapled to it. The showroom paid it without comment.

Eighteen months later that file had been used for the visiting cards, two shutter stickers, a bill book, a Diwali hoarding and a wedding-season van wrap, and every one of those orders came back to Sundeep, because he was the person who had the file.

The client is certain he has supplied a logo. What is the fastest honest way to show him otherwise?

Print one letter from the supplied image at the size it will actually be used and put it in front of him. A 300 mm crop on an office printer, seen from where a customer stands, settles in ten seconds what an explanation of raster and vector will not settle in ten minutes. The argument is not that the file is low quality; it is that the file records pixels at one fixed size and the job needs geometry at another.

Suppose the job had been the hoarding alone, with no routed letters. Would tracing have been defensible?

Tracing would have been beside the point, because the hoarding could have taken the raster directly. The real problem there is resolution: 480 pixels across six metres is about fifteen times short even of the 20 to 40 ppi a hoarding needs, so the file is inadequate whether it is traced or not. The honest options would have been an upscale, disclosed as invented detail, or a redraw. And the redraw still earns its keep, because the artwork will be reused.

What should Sundeep keep when the job is finished, and why does it matter commercially?

The redrawn vector master, clearly named and archived with the date, plus an outlined PDF as a format-independent copy. Being the only person who holds a usable file makes him the default supplier for every later job, and it protects the client from the same problem recurring in three years when a different shop is asked for a sticker. Charging for the reconstruction openly, and itemising it, is what turns an unpaid favour into a service the client understands they bought.

MODULE 1 · TEST

Check what stuck

6 questions, one right answer each. This one is for you, not for a record: answers and the reason behind each are at the back. If two questions from the same module go wrong, re-read that module before the exam rather than after.

1 A 100 mm square drawn in Inkscape 0.91 measures 93.75 mm when the same file is opened in Inkscape 1.x. What happened?

A. The file was saved at 90 dpi and the newer version resampled it to 96 dpi
B. The viewBox was written wrongly by the older version and has now been silently corrected
C. The bare numbers in the file were read at 96 units to the inch instead of 90
D. Millimetres were converted using 72 points to the inch rather than 25.4

2 A keyline set to 'hairline' proofs perfectly on the 600 dpi office laser and is missing from five thousand printed cartons. What is the mechanism?

A. The PDF export dropped the stroke because the object had no fill
B. The press trapped the line away when it spread the lighter colour into the darker one
C. The line was set to overprint, so it printed beneath the background
D. Hairline means the device's thinnest line, which on a platesetter is 0.011 mm

3 A ring built from a large circle and a small one previews as a solid disc. Which fix is the more portable, and why?

A. Reverse the direction of the inner subpath
B. Move the inner circle to the top of the stacking order
C. Group the two circles rather than combining them
D. Set the object to even-odd fill in the application

4 A 40 MB photograph was cropped to a 20 mm square in the layout, and pdfimages lists it in the exported PDF at 6000 by 4000 pixels. What does that tell you?

A. The export downsampled the file but kept its original colour depth
B. The crop was a clipping path, so the whole photograph is still there
C. The image was embedded rather than linked, which doubles its size
D. PDF/X requires both a low-resolution preview and the full image to be stored

5 A drop shadow looks perfect on screen and prints visibly stepped and chunky. Which setting explains it?

A. The rendering intent, which was left on saturation
B. The image downsampling threshold in the PDF export
C. The line screen, which the RIP set too coarse for the stock
D. The document's raster effects resolution, left at 72 ppi

6 Auto-tracing a clean scan of a square logo returns rounded corners. Which stage of the tracer did it, and what do you change?

A. The threshold: raise the brightness cutoff so the ink is fully black
B. The corner detection: lower the corner threshold so corners survive
C. The curve fitting: loosen the error tolerance so fewer nodes are used
D. The speckle filter: raise it so small features are discarded

MODULE 2

Drawing curves people will pay for

The manual craft. The four gestures of the pen tool, what makes a join kink, how booleans multiply nodes and how to clean up after them, offsets and the miter spike, transforms as arithmetic rather than dragging, and the grid and snapping setup that helps instead of fighting you. Ends with the workflow for redrawing something that exists only as a photograph.

> BY THE END OF THIS MODULE YOU CAN
>
> Draw, correct and combine any shape with the pen, node, boolean, offset and transform tools, name the node type, handle or continuity failure responsible when a curve goes lumpy, and rebuild an existing mark from a photograph or scan as exact geometry rather than as an approximation.

---

9 · 10 min

Your curves are lumpy because you used too many points

THE ONE THING TO KEEP

Put anchor points only at a curve's extremes — top, bottom, left, right — and fix problems by adjusting handle length, never by adding another point.

The pen tool is not hard, it is misunderstood

Almost everyone learning vector drawing produces the same result: an outline that looks faceted, with flat stretches between small bumps, and a node count in the hundreds. They conclude the pen tool is difficult. It is not. They are placing points wherever the curve seems to change, and every point they place is another chance to break the curve.

Here is the mechanism. A path is made of **anchor points**. Between any two anchors the software draws a cubic Bézier curve, and the shape of that curve is set entirely by two **handles** — one leaving the first anchor, one arriving at the second.

Two properties matter:

- The **direction** of a handle is the direction the curve travels as it leaves that anchor. The curve is tangent to the handle.
- The **length** of a handle is how far the curve commits to that direction before it starts turning towards the other anchor.

That is the whole system. Every curve you have ever seen in a font, a logo or a map is this, repeated.

Why more points make it worse

Each anchor imposes a tangent constraint. If you place twenty anchors around a circle, you have declared twenty tangent directions, and unless every single one is exactly right, the curve visibly kinks at each. Four anchors declare four tangents, all of them easy to get right, and the software interpolates a perfect arc between them.

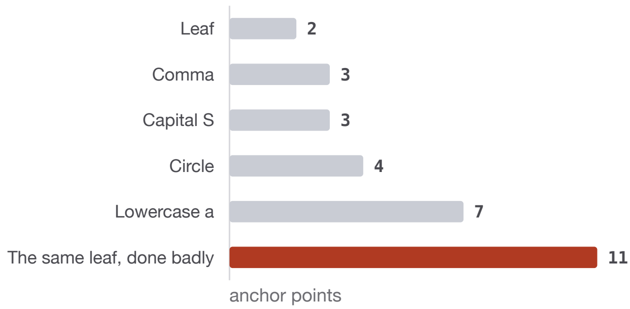
So the rule is not "draw carefully". It is **use fewer points, and put them at the extremes**.

An extreme is the topmost, bottommost, leftmost or rightmost point of a curve — the place where the tangent is exactly horizontal or exactly vertical. At an extreme, you know the handle direction with certainty, because there is only one correct answer.

Prove it to yourself in two minutes. Draw a circle with the ellipse tool. Switch to the node tool — **N** in Inkscape, the white arrow **A** in Illustrator, the Shape tool **F10** in CorelDRAW — and look at it. Four nodes: twelve, three, six and nine o'clock. Every handle is perfectly horizontal or perfectly vertical. That is not a coincidence; it is how the shape is defined, and it is how every glyph in every professional typeface is built.

An "S" needs three points. A teardrop needs two curve points and one corner. A leaf needs two. If your leaf has eleven, that is why it looks lumpy.

Anchor points these shapes actually need



Every anchor declares a tangent direction the curve must obey, so each one you add is another chance to kink the outline. Put them at the extremes only, then fix problems by changing handle length rather than by adding a point.

The habit that fixes it

1. Before you click anything, look at the shape and mark the extremes mentally. Those are your anchors. Nowhere else.
2. Click and drag at each one. Drag along the direction of travel, holding **Shift** to lock the handle to horizontal or vertical.
3. Make the handle roughly one third of the distance to the next anchor. This is a rule of thumb, not a law, but it gets you close and stops the overshooting bulge that comes from handles that are too long.
4. Only then go back and adjust. Adjust handle length before you consider adding a point. Adding a point is almost always the wrong fix.

Where the curve genuinely changes direction — a corner — you want a **cusp** node, where the two handles move independently. Hold **Alt** and drag a handle to break the pair in Illustrator and Inkscape; in CorelDRAW, select the node and press the cusp button on the property bar. Where the curve should flow through smoothly, you want a **smooth** node, where the handles stay in a straight line through the anchor. Inkscape has explicit commands for this on the node tool: **Shift+S** to make a node smooth, **Shift+Y** for symmetric, **Shift+C** for cusp.

The drill

Set a capital **S** in a heavy typeface at 400 points. Lock it on its own layer at 30% opacity. Trace it with three anchors. You will get it wrong the first four times. When the curve finally sits on the letter with three points, you have learned the thing this lesson exists to teach, and it will not leave you.

Then do a lowercase **a**, which has an inner counter and a spur, and needs about seven.

Auto-trace is not a shortcut past this

Image Trace, Trace Bitmap and PowerTRACE all place their nodes along the pixel boundary they found, at intervals the algorithm chose. They routinely give you 200 nodes on a shape that needs eight. The result looks acceptable on screen at 100% and reveals its wobble the moment it is enlarged, cut in vinyl or embroidered.

Simplify helps a little — **Ctrl+L** in Inkscape, Object > Path > Simplify in Illustrator, the node-reduction slider on CorelDRAW's Shape tool property bar — but it removes points by averaging, so it rounds off the corners you wanted to keep while leaving the misplaced points that caused the wobble. Treat a trace as a template to redraw over, not as artwork.

It transfers

This is cubic Bézier geometry, standardised decades ago. The same handles run the pen tool in Figma and Penpot, mask paths in After Effects, curves in Blender, motion paths in animation software, and the outline of every letter you are reading. Learn the handles once and you have learned every pen tool you will ever meet.

BEFORE YOU MOVE ON

A beginner traces a curved letterform and the outline comes out faceted, with flat stretches between small bumps. The path has 40 anchor points. What is the fix?

A. Convert all the nodes from cusp to smooth so the handles are forced collinear, keeping the 40 points in place.
B. Redraw with one point at each extreme of the curve — top, bottom, left, right — so a handful of anchors and their handles describe the whole shape, then correct by changing handle length rather than adding points.
C. Export at a larger size or raise the document resolution, since the faceting is a preview artefact rather than something in the geometry.
D. Run Simplify repeatedly until the point count falls, which restores a smooth curve automatically.

10 · 9 min

Most good logos are circles and rectangles cut into each other

THE ONE THING TO KEEP

Assemble marks from exact primitives and cut them with booleans, duplicate before every boolean because it is destructive, and remember a hole that fills solid is a fill-rule problem, not a stacking one.

Stop drawing logos freehand

Watch a beginner build a mark and they open a blank canvas and start clicking with the pen tool. Watch someone who does this for a living and they draw a circle, draw a rectangle, align them, and cut one out of the other. The pen tool comes out for the parts nothing else can make.

There is a practical reason. A circle drawn with the ellipse tool is mathematically a circle. A circle drawn by hand with the pen tool is four slightly wrong curves, and at 6 metres or in a 16-pixel favicon, "slightly wrong" is what people see without being able to say why. Primitives are exact for free.

The five boolean operations

Select two or more objects and combine them:

- **Union** (Weld in CorelDRAW) fuses them into one outline.
- **Difference** (Minus Front, Trim) removes the top shape from the one beneath.
- **Intersection** keeps only the overlap.
- **Exclusion** (Exclude, XOR) keeps everything except the overlap.
- **Division** (Divide, Break Apart) cuts the shapes into separate pieces along every intersection, which is the one people forget and the one that does the most work.

Where they live:

- **Inkscape**: the Path menu, and they are keyboard-driven, which makes them fast. Union **Ctrl++**, Difference **Ctrl+-**, Intersection **Ctrl+***, Exclusion **Ctrl+^**, Division **Ctrl+/**.
- **Illustrator**: the Pathfinder panel (Window > Pathfinder), and the Shape Builder tool, **Shift+M** — drag across regions to merge them, hold **Alt** and drag to delete a region. Shape Builder is the fastest way to work and worth the hour it takes to get used to.
- **CorelDRAW**: select two objects and Weld, Trim and Intersect appear on the property bar; the fuller set is in the Shaping docker. **Boundary** creates a new outline around a selection, which is useful for making a knockout behind a group.

Build something concrete

A wifi mark. Four concentric circles, all centred. Convert the outer three to rings by subtracting a smaller copy of each. Draw a rectangle covering the lower half and rotate it to a wedge; select all and use Division, then delete everything outside the wedge. Add a dot. That is the mark, and it is exact, because every element started life as a mathematically perfect circle.

A play button. Circle, plus a triangle made from a three-point polygon, then Difference. Ten seconds.

Alignment does the rest of the work. Learn one thing per program: Illustrator's **align to key object** — click one selected object again to make it the anchor, then align everything to it. Inkscape's Align & Distribute dialog has a "Relative to" dropdown; set it to "Last selected". CorelDRAW aligns with **C** for centre-horizontally and **E** for centre-vertically, and **P** centres on the page.

Booleans are destructive, and that costs people work

Union throws away the originals. Difference throws away the cutter. There is no going back after you save and close.

- Duplicate first. Keep an untouched copy on a locked layer named `source`. This costs nothing and saves an afternoon.
- Illustrator will make a **compound shape** instead of a flattened path if you **Alt-click** a Pathfinder button; the components stay editable inside it until you press Expand.
- Affinity Designer holds **Alt** while clicking a boolean to keep the operation live, and it is genuinely better than the alternatives here — you can move the cutter afterwards and watch the result update.
- Inkscape has no live boolean. Duplicate first, or you are relying on undo.

The bug everyone hits: the hole that will not appear

You make a ring by combining a big circle and a small one, and it previews as a solid disc.

The cause is the **fill rule**. A shape with more than one subpath needs a rule for deciding which regions count as inside. **Non-zero winding** counts a region as filled when the subpaths around it wind in the same direction, so two circles drawn the same way both count as inside and you get a disc. **Even-odd** counts crossings, so the inner region is outside again and you get a hole.

Two fixes, both one click:

- Reverse the direction of the inner path. Inkscape: Path > Reverse, **Shift+R**. Illustrator: the Reverse Path Direction button in the Attributes panel. CorelDRAW: with a compound path, use the Shape tool and reverse the subpath.
- Or set the object to even-odd fill. Inkscape's Fill & Stroke dialog has the two buttons next to the fill type; Illustrator's Attributes panel has "Even-Odd Fill Rule"; in SVG it is literally `fill-rule="evenodd"`.

This same rule is why a letter "O" imported from somewhere sometimes fills solid, and why a die-line with an interior window prints as a block.

Strokes are a promise, not a shape

A stroke is a rendering instruction: draw this path, this many units wide. It is not geometry until you make it geometry. That matters in three places, so learn the command now:

- Inkscape: Path > Stroke to Path, **Ctrl+Alt+C**
- Illustrator: Object > Path > Outline Stroke
- CorelDRAW: Object > Convert Outline to Object, **Ctrl+Shift+Q**

Do this before sending anything to a cutting plotter, a laser or an embroidery digitiser, and before applying a boolean to a stroked object. Also check "Scale strokes and effects" in your preferences — with it off, scaling a logo down leaves the stroke at its original weight, and your 2 mm outline becomes an inky mess at business-card size.

Everything in this lesson works identically in the free tools. Nothing here is behind a paywall.

BEFORE YOU MOVE ON

Someone makes a ring by drawing a large circle and a smaller circle inside it and combining them into a single path. It previews as a solid disc with no hole. What has happened?

A. Both subpaths wind in the same direction and the non-zero fill rule counts the inner region as inside; reversing the inner path's direction, or switching the object to even-odd fill, opens the hole.
B. The inner circle is behind the outer one in stacking order, so bringing it to the front will reveal the hole.
C. The inner circle still carries a fill colour, and setting its fill to none will make the hole appear.
D. A union was applied where a difference was needed, so the only fix is to undo and re-run the operation as a difference.

11 · 8 min

The pen tool is four gestures

THE ONE THING TO KEEP

The pen tool has exactly four gestures — click, click-drag, Alt-drag to break the handle pair, and click-to-close — and the one that goes wrong is dragging towards the bend instead of along the direction of travel.

Everything the tool does

People avoid the pen tool because they think it has a hundred behaviours. It has four, and they are the same in Illustrator, Inkscape, CorelDRAW, Affinity, Figma and Blender.

1. **Click.** Places a corner anchor with no handles. The segment arriving at it is straight.

2. **Click and drag.** Places a smooth anchor and pulls out two handles at once, mirror-image, along the direction you dragged. The segment arriving curves into that direction; the segment leaving continues it.
3. **Click and drag, then hold Alt and drag again** (before releasing, or on the handle afterwards). Breaks the mirror, so the outgoing handle points somewhere the incoming one does not. This is how you get a curve that arrives smoothly and leaves at an angle — a petal tip, the join of a bowl to a stem.
4. **Click the first anchor again.** Closes the path. Your cursor shows a small circle when you are over it.

That is the tool. Everything else is editing afterwards.

The gesture that trips everyone up

Drag *in the direction the curve is going*, not towards the shape. Beginners drag towards where they want the line to bend, which points the tangent the wrong way and produces a loop or an S where they wanted an arc.

A useful mental model: the handle is a speedometer and a compass. Direction says which way the pen is travelling as it leaves. Length says how long it keeps going that way before it starts turning.

Getting out of the path

Two problems everybody has in their first hour.

"I can't stop drawing." Press **Escape** (Illustrator, Affinity), **Enter** or double-click the last point (Inkscape), or **Space** then Escape (CorelDRAW). Some programs also end a path when you switch tools.

"I started a new shape and it joined onto the old one." You did not deselect. Press Escape first, every time, before starting a new path. Make it reflex.

Continuing and joining

- **Continue an open path**: hover the pen over an end anchor until the cursor shows a small slash or a plus, then click.
- **Join two open paths**: select both end anchors and press **Ctrl+J** in Illustrator or **Ctrl+J** in Inkscape (Path ▸ Join nodes). CorelDRAW: select both nodes with the Shape tool and press the *Join two nodes* button.
- **Cut a path**: Illustrator's Scissors (**C**), Inkscape's node tool *Break path at selected nodes*, CorelDRAW's *Break curve*.

The drill that works

Take a leaf, a comma and a heart, in that order.

- **Leaf**: two anchors. One at the tip, one at the base, both cusp. Two curves bulging opposite ways. If you use four, you will see the flat spots.
- **Comma**: three anchors — top, the outside of the curve, the tail. The tail is the Alt-drag case.
- **Heart**: four anchors — the two lobes at their outermost points, the dip at the top centre, the point at the bottom. The dip and the point are cusps; the lobes are smooth.

Do each one five times. On the fifth attempt you will place the anchors before you click, which is the whole skill.

Two modifiers worth learning immediately

- **Shift** while dragging a handle locks it to horizontal, vertical or 45°. At an extreme point — top, bottom, left, right of a curve — the correct handle is exactly horizontal or exactly vertical, so Shift is not a convenience here, it is correctness.
- **Space** while placing an anchor (Illustrator, Affinity) lets you reposition it before you commit. Saves the undo.

The tool nobody mentions: the pencil

Inkscape's pencil (**P**) and Illustrator's pencil draw a freehand path and fit curves to it, with a smoothing tolerance you can set. For an organic shape drawn on a graphics tablet — a brush mark, a signature, a hand-lettered flourish — it is faster than the pen and produces a better result, because the tangents follow an actual gesture.

It gives you more nodes than a hand-placed path, so simplify afterwards and check the count. For anything geometric it is the wrong tool; for anything gestural it is the right one. Most people never try it and draw wobbly organic shapes with the pen for years.

BEFORE YOU MOVE ON

A learner drawing a simple arc clicks the start point, then clicks and drags at the end point towards the middle of the intended curve. The result is an S-shape rather than an arc. What did the drag actually do?

A. It placed a cusp node instead of a smooth one, and cusp nodes always produce an inflection between segments.

B. It set the incoming tangent to point back towards the middle, so the curve reverses direction before arriving.

C. It lengthened the handle beyond the distance between anchors, which forces the curve to overshoot and loop back.

D. It applied the drag to the previous anchor rather than the new one, because the pen tool always edits the last committed point.

12 · 9 min

The kink you can see but cannot find

THE ONE THING TO KEEP

Vector editors enforce matching tangents but never matching curvature, so a join can be perfectly "smooth" and still show a visible crease — usually caused by mismatched handle lengths, a curve meeting a straight, or an anchor that is not quite at the extreme.

Three grades of smooth

Two curve segments meeting at an anchor can agree to different degrees, and the words for it come from geometry rather than from software menus.

- **G0** — the segments meet at the same point. That is all. A corner.
- **G1** — they meet *and* their tangent directions are the same. The handles are collinear: a straight line through the anchor. No corner, but the curvature can jump.
- **G2** — they meet, the tangents agree, *and* the curvature agrees. The curve bends by the same amount on either side.

Vector editors give you G0 (cusp) and G1 (smooth). **None of the common ones enforce G2.** Illustrator, Inkscape, CorelDRAW and Affinity will all happily let you build a G1 join where the curvature jumps from tight to loose, and your eye will catch it as a flat spot or a crease even though no corner exists.

How to see it

The trick that industrial designers use and graphic designers mostly do not: **add a thick stroke and look at the inside edge.** Where curvature jumps, the inner offset of the stroke develops a visible corner even though the centreline is smooth. Put a 6 mm stroke on your curve, zoom in, and the crease announces itself.

A second test: **mirror it.** Duplicate the shape, flip it horizontally, butt the two together. Any asymmetry or curvature mismatch at the join becomes obvious because now there are two of them.

The common causes, in order of frequency

Unequal handle lengths across a smooth node. The handles are collinear so the join is G1, but one is 5 mm and the other is 40 mm. The curve arrives bending hard and leaves bending gently. Fix by matching the lengths, or by making the node *symmetric* rather than merely *smooth* — Inkscape's **Shift+Y** makes handles symmetric, **Shift+S** makes them merely collinear.

A straight segment meeting a curve. A line has zero curvature. Any curve meeting it has non-zero curvature. The jump is mathematically unavoidable at the anchor, so what you control is how gentle it is: make the curve's handle leaving that anchor *long*, so it starts nearly straight. A short handle there produces the tell-tale "the arc starts too suddenly" look you see on cheap rounded rectangles.

A node that is not quite at the extreme. If the topmost point of a curve is 0.3 mm to the left of your anchor, the handle cannot be horizontal and still be correct, and every attempt to smooth it fails. Move the anchor to the extreme first, then straighten the handle.

Where it shows up in real work

- **Rounded rectangles.** A plain rounded rectangle is four quarter-circles joined to four straight lines: eight G1 joins with a curvature jump at each. At large sizes on a sign, you can see all eight. This is precisely why Apple's superellipse corners and CSS's "squircle" exist, and why a serious logo with rounded corners is drawn with manually tuned handles rather than a corner-radius field.
- **Letterforms.** The join between a stem and a bowl is where a badly digitised typeface betrays itself. Set a large **o**, **b** and **S** and look at the inside of the curve.
- **Vehicle graphics and signage.** A swoosh 3 m long makes a 1% curvature mismatch a 30 mm visible flat.

The practical fix list

1. Put anchors only at extremes and at genuine corners.
2. Make smooth nodes symmetric unless you have a reason not to.
3. Where a curve meets a straight, lengthen the handle on the curve side.
4. Stroke it thick and look at the inside edge.
5. If it still creases, remove the anchor entirely and let one longer segment do the work. Fewer anchors means fewer joins means fewer places for curvature to jump.

What the free tools can and cannot do

Inkscape's node tool has explicit *make symmetric* and *make smooth* commands and a **Ctrl+L** simplify. It has no curvature comb and no G2 tools. Illustrator has neither either. Genuine curvature analysis lives in CAD — FreeCAD and Blender both show curvature, and Blender's curve editor will display a curvature comb on a Bézier.

For logo work you do not need CAD. You need the thick-stroke test and the discipline to use three anchors where you wanted six.

BEFORE YOU MOVE ON

A designer draws a long swoosh for a 3 m shopfront. Every node is set to smooth and every handle pair is perfectly collinear, yet a visible flat spot appears at one node when the sign is installed. What is the most likely cause?

A. The node is a cusp that the software is displaying as smooth because the handles happen to line up.
B. The file was scaled up after drawing, and scaling amplifies the rounding error in the stored coordinates.
C. The handles either side of that node are very different lengths, so curvature jumps across a tangent-only join.
D. The stroke alignment is set to inside on one segment and to centre on the other, producing a visible step at the join.

13 · 9 min

Booleans, and the mess they leave behind

THE ONE THING TO KEEP

A boolean splits paths at every intersection and restitches the survivors, so it adds nodes, hands the result a fill you did not choose, and destroys the inputs — clean up immediately, set colour afterwards, and keep a locked copy of the sources.

What a boolean actually computes

Union, difference, intersection and exclusion are not shape tools. They are set operations on filled regions, and the program computes them the same way every time:

1. Find every point where the input paths intersect.
2. Split all paths at those points into segments.
3. Decide, for each segment, whether it lies on the boundary of the result.
4. Stitch the surviving segments into new closed paths.

Three consequences fall straight out of that, and they are the whole lesson.

What a boolean computes, every time

Find crossings	→	Split there	→	Keep or drop	→	Stitch up
Every point where the inputs intersect		Each path is cut into segments		Is this segment on the new boundary?		Survivors joined into closed paths

Two things follow from step two and you will meet both. Every old crossing survives as a new node with handles the solver chose, and the inputs are gone, so duplicate onto a locked source layer before you start and set the colour afterwards.

One: every intersection becomes a node

Weld two circles and the result has the original anchors *plus* two new ones where the outlines crossed. Do six booleans in a row and the node count compounds. A mark that should be forty nodes arrives at three hundred, and it prints the same but cuts, stitches and enlarges badly.

Clean up after every boolean, not at the end:

- Delete nodes that sit on a straight run or in the middle of a smooth curve. Select them and press **Ctrl+Delete** in Illustrator (delete without breaking the path) or **Ctrl+Delete** in Inkscape (delete preserving shape).
- Look specifically at the new nodes at the old intersections. They almost always have handles the algorithm chose, not handles you would have chosen.
- Do not reach for Simplify. Simplify averages across the whole path and rounds off the corners you meant to keep. It is a last resort on traced artwork, not a cleanup for your own drawing.

Two: appearance is inherited, and the rule differs

Which object's fill does the result take?

- **Illustrator Pathfinder** — the *bottom* object's appearance for Unite, the *top* object's for Intersect in some versions. This has changed across releases; test rather than trust.
- **Inkscape** — the *bottom* object's style for Union and Difference.
- **CorelDRAW Weld** — the *target* object's, where the target is whichever you selected last (or the bottom, if you used a marquee).

The practical habit: set the colour after the boolean, never before. You will stop losing five minutes a day to it.

Three: it is destructive, and the escapes are not equal

- **Affinity Designer**: hold **Alt** while clicking a boolean. You get a live compound whose children stay editable and movable, updating the result. This is genuinely the best implementation of the four.
- **Illustrator**: **Alt+click** a Pathfinder button makes a compound shape. Components remain editable until you press *Expand*. The Shape Builder

tool (**Shift+M**) is different and better for fast work: drag across regions to merge, **Alt+drag** to delete a region, and you see the result as you go.

- **CorelDRAW**: the Shaping docker has *Leave original: source / target* checkboxes. Tick them and the inputs survive beside the result.
- **Inkscape**: no live booleans in the Path menu. The *Boolean operation* Live Path Effect (1.x) gets close, and otherwise you duplicate first (**Ctrl+D**) and stash the copy on a locked layer.

> *Make a layer called `source`, put an untouched copy of every input on it, and lock it. Thirty seconds now against an afternoon later.*

Division is the underused one

Division (Illustrator's *Divide*, Inkscape's Cut Path / Division, CorelDRAW's *Simplify* and *Front Minus Back* family) cuts every input into separate pieces along every intersection and keeps them all.

This is how you build a mark whose parts must line up exactly: draw the whole silhouette, draw the cutting lines across it, divide, then colour the pieces. The edges are shared by construction, so there is no hairline gap between them and no overlap. Trying to achieve the same result by drawing each piece separately is how you get the thin white line that shows on press.

The failure to watch for: self-intersection

A path that crosses itself, or two identical paths stacked exactly, makes the boolean solver's job ambiguous. Symptoms: the operation does nothing, or produces a shape with a stray sliver, or produces zero-area artefacts you cannot see but a cutting plotter finds.

Before a boolean on anything traced or imported: run Union on the object *with itself* to clean overlaps, then check the node count. Inkscape's Edit ▸ *Select Same* and the XML editor let you spot duplicate paths; Illustrator's Document Info panel gives object counts.

BEFORE YOU MOVE ON

A mark built from six successive Weld and Trim operations is sent to a vinyl plotter. The cut has small stray nicks that do not appear anywhere in the on-screen artwork. What is the most likely cause?

A. The blade pressure was too high, so the plotter cut through the backing sheet at every change of direction.
B. The file was exported as SVG, which stores coordinates at reduced precision and rounds some nodes onto each other.
C. The strokes were not converted to outlines, so the plotter cut the centreline of each stroke as an extra path.
D. Zero-area slivers left by the boolean solver where two inputs overlapped almost exactly.

14 · 8 min

Offsets, and the spike that appears at a sharp corner

THE ONE THING TO KEEP

An offset runs parallel to the original, so a mitred corner at angle θ lands at `offset ÷ sin(θ/2)` from the point — 5 mm becomes 29 mm at a 20° tip — which is why join style and miter limit exist and why round joins are the safe default.

Offset is not the same as a thicker stroke

Two operations look similar and behave completely differently.

- **Outline stroke / stroke to path** takes a 2 mm stroke and turns it into a closed shape 2 mm wide, straddling the original path. The centreline is where your path was.

- **Offset path / contour** takes a path and produces a *new* path a fixed distance away from it, on one side. The original stays where it was.

You want the first when preparing artwork for a cutter. You want the second for a sticker border, a keyline around a logo, a bleed contour, a CNC toolpath, or an embroidery underlay.

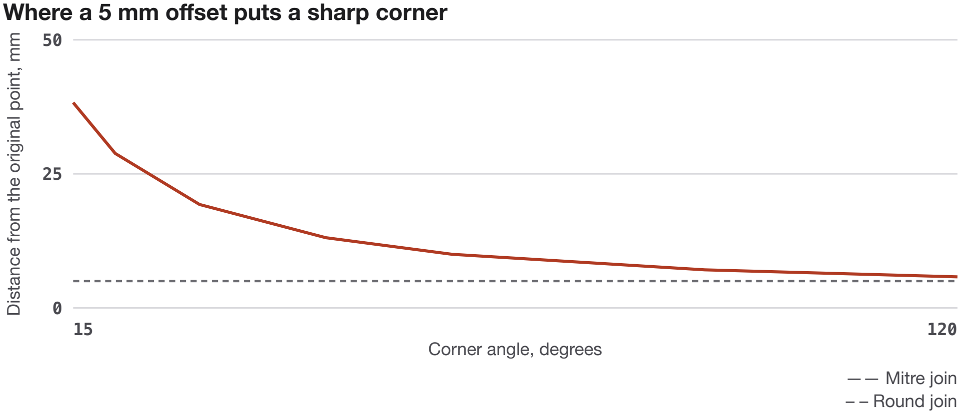
Where they live: Illustrator has Object ▸ Path ▸ *Offset Path* and Object ▸ Path ▸ *Outline Stroke*. Inkscape has Path ▸ *Outset* / *Inset* (fixed 2 px steps), Path ▸ *Dynamic Offset* (**Ctrl+J**, draggable and live), Path ▸ *Linked Offset*, and the *Offset* Live Path Effect for a numeric value. CorelDRAW has the interactive **Contour** tool, which does the same thing with a step count.

Why a sharp corner grows a spike

Offset a rectangle outwards by 5 mm and every corner is still a right angle, 5 mm further out. Now offset a shape with a 20° point — the tip of a star, the point of an arrow, an italic serif.

The two offset edges run parallel to their originals. Extended outwards, they do not meet 5 mm from the tip. They meet much further out. At angle θ, the mitred tip sits at a distance of `offset ÷ sin(θ/2)` from the original point. At 20°, `sin(10°)` is 0.174, so a 5 mm offset puts the new tip **28.8 mm** away. At 10° it is 57 mm.

Where a 5 mm offset puts a sharp corner



A mitred tip lands at the offset divided by the sine of half the angle, so it runs away as the corner sharpens. A round join stays at the offset distance whatever the angle, which is why it is the safe default on anything you cannot inspect.

That is the spike. It is not a bug; it is what a parallel offset means.

Every implementation therefore offers a **join** setting with three options:

- **Miter** — extend to the true intersection. Correct, and produces the spike.
- **Round** — arc across at the offset radius. Always safe, slightly softens every corner.
- **Bevel** — cut straight across. Flat corners.

And a **miter limit**: the ratio of spike length to offset distance at which the program gives up and bevels instead. SVG and PDF default to 4, which switches over at roughly 29°.

Set joins to round on any offset you cannot inspect, and set them deliberately on the ones you can.

Inner offsets eat themselves

Offset inwards far enough and thin parts of the shape vanish, then reappear inside out. A 2 mm-wide bar offset inwards by 1.5 mm leaves a 0 mm bar; at 2 mm it self-intersects and most solvers produce a small backwards loop rather than nothing.

This matters for three real jobs:

- **Embroidery underlay**, which is an inset of the shape. A thin serif with too much inset produces a loop the machine stitches as a knot.
- **Screen-print underbase choke**, typically 0.3–0.5 mm inset.
- **Laser kerf compensation**, where an interior cut-out must be offset inwards by half the kerf.

Check thin features explicitly after every inward offset. Measure the narrowest part of your shape before you choose the distance.

Corner rounding is a different tool again

"Round the corners of this rectangle by 4 mm" is not an offset. Illustrator's *Live Corners* (drag the small circle widgets with the Direct Selection tool), Inkscape's rounded-rectangle handles and the *Corners* Live Path Effect, and CorelDRAW's *Fillet/Scallop/Chamfer* docker all do this properly, and all of them refuse when the requested radius is larger than the shorter adjacent edge.

Do not fake a rounded corner by offsetting out and back in. It works, and it costs you nodes and precision each way.

One habit

After any offset, zoom to 400% on the sharpest corner in the shape and look. The spike is either there or it is not, and it takes three seconds.

And measure the result rather than trusting the dialog. Offset implementations differ: Inkscape's Inset and Outset move by a preference value in pixels, not by the number you had in mind, and CorelDRAW's Contour tool offsets by *step* × *offset*, so a 2-step contour at 1 mm reaches 2 mm. Draw a 50 mm square, offset it by 5 mm, and measure the result once in whatever program you use. You will know for the rest of your career which number that field means.

BEFORE YOU MOVE ON

A 5 mm keyline is added around a star-shaped logo using Offset Path with mitre joins. The five outer points now extend nearly 30 mm past where the designer expected. What determines that distance?

A. The offset value is applied along the path rather than perpendicular to it, so it accumulates around each point.
B. The miter limit was set above 4, which multiplies the offset by the limit value at every corner.
C. The two offset edges stay parallel to their originals, so at a sharp angle θ they only meet at `offset ÷ sin(θ/2)` from the tip.
D. The star was drawn as a self-intersecting path, so the offset was computed on the winding boundary rather than the visible outline.

15 · 8 min

Transforms are arithmetic, so type the numbers

THE ONE THING TO KEEP

Every transform is a matrix multiply about an origin you can move, so type the numbers rather than dragging — rotate once by the total instead of repeatedly, and know whether you want the group transformed or each object transformed about its own centre.

Dragging is for looking; typing is for building

Every move, scale, rotate, skew and flip is the same operation underneath: each point `(x, y)` is multiplied by a 2×3 matrix.

```
x' = a·x + c·y + e
y' = b·x + d·y + f
```

- Translate: `e` and `f` are the offsets.
- Scale: `a` and `d` are the factors.
- Rotate by θ: `a = cos θ`, `b = sin θ`, `c = −sin θ`, `d = cos θ`.
- Flip horizontally: `a = −1`.

You never write this out. But knowing it explains three things people find mysterious.

Rotation happens about the origin. Every implementation rotates about (0,0) and then translates the result back. The point you rotate *about* is a choice the software makes — normally the bounding-box centre. Move the origin and the same rotation lands somewhere completely different.

Rotations compound rounding. Rotate 15° twelve times and you do not get exactly 180°; you get a shape whose coordinates have been through twelve floating-point round-trips. On screen this is invisible; on a laser cutting a slot to 0.05 mm it is not. Rotate once, by the total.

Non-uniform scale changes stroke weight and letterform. Scale to 200% × 100% and a 1 mm circular stroke becomes an ellipse; type becomes a mechanically stretched parody of itself. Hold Shift.

Set the origin before you rotate

This is the single most useful transform skill and most people never learn it.

- **Illustrator**: press **R** for the Rotate tool, then **Alt+click** where you want the pivot. A dialog opens with the angle; type it. Press **Copy** instead of OK to duplicate as you rotate.
- **Inkscape**: click an object twice to get rotation handles, then drag the small cross to move the pivot. Or use Object ▸ Transform (**Shift+Ctrl+M**) ▸ Rotate for numeric entry, and Edit ▸ *Paste in Place* for exact duplicates.
- **CorelDRAW**: click twice for rotation handles, drag the centre marker, then use the Transformation docker for numbers.
- **Affinity**: the origin handle can be dragged and the Transform panel takes typed values with arithmetic (`120/3` works in the field).

Step and repeat, which pays for itself

Twelve marks around a circle, forty labels on a sheet, a row of perforations.

Illustrator: rotate about a pivot with **Copy**, then **Ctrl+D** to repeat the last transform. Eleven presses gives you a twelve-spoke wheel, exact.

Inkscape: Edit ▸ *Clone* ▸ **Create Tiled Clones** does rows, columns, rotation, symmetry groups and randomisation, and clones update when you edit the original. It is more capable than anything in the paid tools and almost nobody uses it.

CorelDRAW: Edit ▸ Step and Repeat (**Ctrl+Shift+D**) with offsets and counts — this is the tool that makes it good at label sheets.

Transform each versus transform together

Select twelve circles and scale to 50%. The *group* shrinks: the circles get smaller and move closer together.

Select twelve circles and use **Transform Each** (Illustrator: Object ▸ Transform ▸ Transform Each, **Alt+Shift+Ctrl+D**). Each circle shrinks about its own centre and none of them move.

Inkscape's Transform dialog has an *Apply to each object separately* checkbox that does the same. This is the difference between "make the dots smaller" and "make the pattern smaller", and picking the wrong one is why a carefully spaced layout collapses.

Align to a key object

Aligning to a selection's bounding box is rarely what you want; aligning everything to one specific object usually is.

- **Illustrator**: select all, then click once more on the object you want to anchor. Its outline goes heavier. Now align.
- **Inkscape**: Align & Distribute (**Shift+Ctrl+A**) ▸ *Relative to: Last selected* or *First selected*.
- **CorelDRAW**: the last-selected object is the target by default when you select with Shift-click.

And *distribute spacing* is not the same as *distribute centres*. For objects of different widths, equal centre spacing looks uneven; equal gaps looks right. Both buttons are in every align panel and people reach for the wrong one constantly.

One last habit: before any transform on a stroked object, decide whether the stroke should come with it. Illustrator's *Scale Strokes & Effects*, Inkscape's four toggle buttons on the selection tool bar, and CorelDRAW's *Scale outline with object* all govern this, and none of them is announced when you drag a handle.

BEFORE YOU MOVE ON

A designer needs twelve spokes evenly around a hub. They rotate a copy by 30°, then rotate that copy by 30°, and so on eleven times, each time dragging by eye with snapping on. The last spoke does not quite meet the first. What is the structural fix?

A. Increase the snapping tolerance so that each successive rotation lands exactly on the intended angle every single time.
B. Set the pivot on the hub centre and use rotate-with-copy plus repeat-last-transform, each applied to the original.
C. Group the eleven copies and apply a single 360° rotation to close the ring exactly.
D. Convert each spoke to a compound path before rotating, so coordinates are stored at full precision.

16 · 8 min

Grids and snapping that help instead of fighting

THE ONE THING TO KEEP

Snapping fights you because it is set to every target at once — keep node-to-node and grid, kill bounding-box snapping because a bounding box includes stroke width — and set the grid from the job's real module rather than from the software default.

Snapping is a preference, not a feature

Out of the box, every vector program snaps to too many things. You reach for a node and it jumps to a guide, a bounding box corner, a page edge and an object centre in quick succession, and you conclude snapping is broken. It is not broken; it is set to snap to everything at once.

The fix is the same everywhere: **turn most of it off and keep two or three targets**.

- **Inkscape**: the snap toolbar (right edge, or the bar at the top in 1.x). Press **%** to toggle all snapping. Enable *snap nodes to nodes*, *snap to paths*, and *snap to grid*. Disable bounding-box snapping entirely — it is the one that fights you, because a bounding box includes stroke width.
- **Illustrator**: View ▸ Smart Guides (**Ctrl+U**), Snap to Point (**Alt+Ctrl+"**), Snap to Grid. Smart Guides alone is usually enough; combined with Snap to Grid it becomes unpredictable.
- **CorelDRAW**: Alt+Z toggles snapping; the Snap To dropdown on the standard toolbar lists objects, grid, guidelines, page and baseline grid separately.
- **Affinity**: the magnet icon's dropdown has a *Snapping Manager* with a preset list; the "Advanced" preset is the usable one.

The bounding-box point deserves repeating. A 2 mm stroke extends 1 mm beyond the path on each side. Snap two stroked rectangles together by their bounding boxes and their *geometry* is 2 mm apart. This is why things that look aligned on screen print misaligned.

Set a grid that means something

A grid at "10 px" means nothing on a 90 × 54 mm card. Set the grid from the job.

- **Print layout**: set the grid spacing to your baseline leading, or to a module you chose — 5 mm with a 1 mm subdivision is a decent default for stationery.
- **Icons**: set the grid to 1 px at your target size and design on it. A 24 px icon drawn on a 24 px grid has crisp edges at 24 px; drawn on a 100 px canvas and scaled down, it does not.
- **Signage**: 10 mm major, 1 mm minor. You will be working at 1:1 or 1:10 and round numbers matter for the fabricator.

Inkscape: Document Properties ▸ Grids ▸ New ▸ set *Spacing X/Y* and *Major grid line every*. It supports an **axonometric grid** for isometric work — set angle X and angle Z to 30° and you can draw isometric illustrations that are actually correct rather than approximately correct. Illustrator: Preferences ▸ Guides & Grid, plus View ▸ Perspective Grid for a different job.

Guides are for decisions, grids are for drawing

A guide records a decision: this is the margin, this is where the fold falls, this is the safe area. Make them, then **lock them** (Inkscape: Edit ▸ Preferences ▸ Behaviour, or right-click a guide ▸ Lock; Illustrator: **Alt+Ctrl+;**). An unlocked guide will be dragged by accident and you will not notice.

Inkscape lets you type exact guide positions by double-clicking a guide — a dialog with X, Y and angle. Illustrator: drag a guide, then use the Transform panel to set its position numerically. CorelDRAW has a Guidelines docker with numeric entry and angle guides.

For a die-line, a fold or a safe area, put the guides in and then **also** put a locked layer with the actual rectangles on it, named `guides-do-not-print`. Guides do not export; a locked layer does, which means the printer can see your intent and you can delete the layer before final output.

Pixel grid, and when it matters

If the artwork's destination is a screen at a known size — an app icon, a favicon, a UI element — enable *Align New Objects to Pixel Grid* (Illustrator) or *snap to pixel* and make every edge land on a whole pixel. A 1 px line sitting on a half-pixel boundary renders as two grey half-lines.

If the destination is print, **turn it off**. Snapping a 90 mm card's artwork to a 96-per-inch pixel grid introduces up to 0.13 mm of error on every edge for no benefit at all. This setting is on by default in some Illustrator web presets and it quietly ruins print files.

The thirty-second setup

New document, before drawing: set units, set page size, set one grid that means something, place your margin and bleed guides, lock them, turn off bounding-box snapping. It saves more time than any keyboard shortcut you will learn.

BEFORE YOU MOVE ON

Two rectangles with 2 mm strokes are dragged together until they snap flush, and they look perfectly butted on screen. On the cut sheet there is a 2 mm gap between them. What happened?

A. Snapping was set to bounding boxes, which include the full stroke width, so the paths stopped 2 mm apart.
B. The stroke was set to align outside on one rectangle and inside on the other, which doubles the apparent offset.
C. Stroke-to-path was applied before export, which expands every shape outwards by the stroke width.
D. The grid spacing was larger than the stroke width, so both rectangles snapped to the nearest grid line and left a gap.

---

17 · 9 min

Rebuilding a mark that only exists as a photograph

THE ONE THING TO KEEP

Reconstructing a logo from a photograph is measurement, not tracing: correct the perspective, lock a dimmed template, identify the typeface and the underlying primitives, redraw with shape tools, then verify with an overlay, a flip and a 16-pixel test.

The job you will actually be given

A client has a logo. It exists on a printed visiting card from 2009, a shop board, and a WhatsApp forward at 480 pixels wide. There is no file. Nobody knows who made it. They want it on a hoarding next week.

This is one of the most common paid jobs in the trade, and there is a method.

Step one: get the best source you can

Rank what they can give you:

1. **The original artwork file**, if anyone still has it. Ask the previous printer — small shops keep CDR files for years and will often hand them over.
2. **A printed piece you can scan.** A visiting card at 1200 dpi on a flatbed is enormously better than any photograph. Ask them to courier one.
3. **A flat, square-on photograph** in even light. Not at an angle, not in sunlight with a shadow across it.
4. **The 480 px WhatsApp image**, which is where most jobs start.

Even at (4) you can work, because you are not going to trace it. You are going to *measure* it.

Step two: correct the perspective

A photograph of a shop board is a trapezoid. Fix it before you draw, in GIMP or Photopea, both free:

- GIMP: Tools ▸ Transform Tools ▸ Perspective. Drag the four corners onto the four corners of the sign.
- Photopea: Edit ▸ Free Transform, then **Ctrl+drag** a corner handle.

Then Levels, so the ink is black and the board is white, and rotate so the baseline is horizontal. Five minutes here saves an hour of drawing something slightly skewed.

Step three: lock it as a template

Place the image on its own layer. Set the opacity to about 30%. **Lock the layer.** In Inkscape, Layer ▸ Add Layer, then the lock icon in the Layers dialog; in Illustrator, the Layers panel's lock column and *Template* option in the layer's options, which also dims it automatically.

Locking is not optional. An unlocked template gets selected and nudged, and you will redraw everything against a reference that has moved.

Step four: identify what it is made of, before drawing anything

Look at the mark and answer three questions.

- **Is the type a known typeface?** Upload the image to a font identifier, or look at distinguishing letters — the **g**, the **a**, the **R** leg, the terminals. If you can identify it, set it as live type and you are 80% done. If you cannot, ask whether it was ever meant to be that face, or whether it was hand-drawn.
- **Which parts are primitives?** Circles, squares, arcs, regular polygons. Draw these with the shape tools, not the pen. The 2009 designer used the same tools you have.
- **What is the construction?** Is the swoosh a circle with a wedge removed? Is the monogram two letters overlapped? Find the underlying geometry and build it that way.

Step five: draw, then verify against the original

Draw over the template with primitives, booleans and the pen. Then check, in this order:

1. **Overlay test.** Put your version at 50% opacity directly over the original at 100%. Every edge should coincide. Where it does not, fix yours — or decide deliberately to improve it and tell the client.
2. **Flip test.** Mirror your version horizontally and look at it. Asymmetries you stopped noticing become obvious.
3. **Small-size test.** 16 px, black on white.
4. **Node count.** Under fifty for a typical mark. If you are at two hundred, you traced when you should have drawn.

What to tell the client

Be straight about two things.

It will not be identical. A reconstruction from a photograph is your best reading of somebody else's geometry. Curves will differ by a fraction of a millimetre. Say so before you start, and show the overlay so they can see how close it is.

You are redrawing, not inventing. The rights in the original mark belong to whoever owned them; redrawing it for the same owner is ordinary practice. Redrawing somebody *else's* mark for a new client is not, and it is a trademark problem rather than a technical one. If the brief smells like that, ask who owns it.

Charge for it properly

This is skilled work with a clear deliverable. Two to four hours for a typical mark, plus the delivery pack. Clients routinely expect it free because "you just have to copy it". It is the opposite of copying — it is reconstruction from evidence, and it is worth saying so once, politely, in the quote.

BEFORE YOU MOVE ON

A designer rebuilds a logo from a 480-pixel WhatsApp image by running it through Image Trace at a high fidelity setting and cleaning up a few nodes. The result looks right on screen. Why is this the wrong method even though the output is vector?

A. Image Trace cannot handle images below 600 pixels, so the output is an approximation rather than a true trace.

B. The trace turns compression blur and the pixel staircase into real geometry, so circles are not circular.

C. Traced output is always in RGB and must be converted to CMYK before printing, which shifts every colour in the mark.

D. Image Trace embeds the source bitmap inside the exported SVG, so the delivered file still contains raster data at 480 pixels.

CASE STUDY · MODULE 2

The star-shaped label and the twenty-six millimetre spike

A label printer in Coimbatore, and a die-maker who telephoned before he bent any steel.

A filter-coffee brand ordered a die-cut star label, 70 mm across the points, for the lid of a 500 g tin. Eight points, each one about 22 degrees at the tip, because the eight points stand for the eight blends the company sells. The brand manager chose the sharp points deliberately and said so twice in the approval email.

The designer built the artwork in Illustrator. Around the star she wanted a white keyline 5 mm wide, so she selected the star, applied Offset Path at 5 mm, and used the resulting outline as the die-line. It previewed correctly on her screen. She exported the PDF, put the cut path on a spot swatch named Dieline, and sent it.

The die-maker telephoned the next morning, before cutting anything.

The die-line he had received was not a star 80 mm across. It was a star 122 mm across, with each point drawn out into a long thin needle. Nothing was corrupt. The offset had done exactly what an offset means: the two edges leaving the tip run parallel to their originals, and parallel lines from a 22 degree corner do not meet 5 mm out. They meet at the offset divided by the sine of half the angle, which at 22 degrees is 5 divided by 0.191, or 26.2 mm from the original point.

So the label would not fit the tin, would not fit two-up on the sheet the press runs, and would have needed a die twice the size of the one quoted.

The die-maker had seen it before and could have simply rounded the corners himself. He rang instead, which is why this case is worth reading, and put the decision back where it belonged.

The designer now had two hours before the press slot on Tuesday, and two options.

Round the joins. Five minutes. Illustrator's Offset Path has a Joins setting; changing it from Mitre to Round produces a border that arcs across each tip at a 5 mm radius. The result is safe, the die fits the quoted size, and every one of the eight points the brand manager specifically approved becomes a 5 mm blunt curve. She would have to send that change for approval, and the brand manager is in Chennai until Thursday.

Build the border as geometry. Forty minutes. Draw a second star, constructed the same way as the first but larger, with the same 22 degree points, so the border sits 5 mm from the artwork along the flats and narrows gradually toward each tip. The points stay sharp. The border is no longer a uniform 5 mm, which is a real difference: at the tip it tapers to nothing, and if anybody measures it, it is not what the specification says.

Both are honest. One changes the shape the client chose; the other changes the specification the client was given. Steel rule dies are cut by hand and a wrong one costs money and a week, so guessing was not among the options.

One more thing surfaced in the phone call, and it is the part worth carrying away. The designer's own preview had shown blunt points, not needles. Her renderer was applying the PDF default miter limit of 4, which gives up and bevels at around 29 degrees, so at 22 degrees it had quietly bevelled the spikes on screen. The die-maker's software was not applying the limit and drew the true mitre. Two programs, one file, both following a specification, and a 26 mm difference between them.

STOP HERE

Before turning the page: what would you have done, and what would it have cost if you were wrong? Write it down. The point of the next section is to compare.

WHAT HAPPENED

She drew the border as geometry, and she measured the result rather than trusting the dialog.

The second star took thirty-five minutes, built with the same construction as the first so the eight points stayed at 22 degrees. She then measured the overall width in the exported PDF: 80.0 mm across the points, which was the quoted die size. She sent the die-maker a marked-up PDF showing the border width at the flats and at the tips, with a one-line note saying the border tapers at the points by design.

She also wrote back to Chennai the same afternoon, before the press slot, with two renders side by side: the rounded version and the tapered version. The brand manager picked the tapered one in an hour. She had chosen not to make the change quietly and then explain it later.

Since then she has one habit she did not have before: after any offset, zoom to 400 per cent on the sharpest corner in the shape and look at it, then measure one overall dimension against the agreed size. Both take a few seconds.

Why did the same file look correct on the designer's screen and wrong in the die-maker's software?

The miter limit. It is the ratio of spike length to offset distance at which a renderer gives up on a true mitre and bevels instead, and the SVG and PDF default of 4 cuts in at roughly 29 degrees. At a 22 degree point one program applied the limit and drew a blunt corner, and the other computed the true intersection and drew the spike. Neither was broken, and the file did not say which behaviour it wanted, which is why joins and miter limits have to be set deliberately rather than left at a default.

Work out where a 5 mm offset puts the tip of a 12 degree point, and say what that implies for the habit of inspecting corners.

Five divided by the sine of 6 degrees, which is 5 divided by 0.1045, or about 47.8 mm from the original point. The relationship runs away as the angle closes, so the sharpest corner in a shape is always the one to check and everything else will be milder. Checking the sharpest corner at 400 per cent after every offset is therefore a complete test rather than a sample.

The designer could have rounded the joins quietly and told the client afterwards. Why is that a worse answer even though the result looks fine?

The eight sharp points were the thing the client chose and explained, so changing them is a design decision rather than a production fix, and it is not hers to make alone. Making it quietly also trains the client to believe the approval process means nothing. Sending two renders and getting an answer in an hour cost less than the conversation that follows a carton arriving with blunt points.

MODULE 2 · TEST

Check what stuck

6 questions, one right answer each. This one is for you, not for a record: answers and the reason behind each are at the back. If two questions from the same module go wrong, re-read that module before the exam rather than after.

1 Why does adding more anchor points around a curve usually make it lumpier rather than smoother?

A. Each anchor forces the application to recompute the whole path
B. Each anchor declares a tangent direction the curve has to obey
C. Each anchor resets both handle lengths to a default ratio
D. Each anchor adds six numbers to the path, so the renderer starts approximating

2 A join is marked smooth in the editor, the handles are in a straight line through the anchor, and there is still a visible crease. What is happening?

A. The anchor is a cusp node that the editor has mislabelled
B. The stroke alignment is set to inside on one of the two segments
C. The two handles are collinear but of very different lengths
D. The path was closed at that point, so the fill rule changes there

3 Offsetting a shape outwards by 5 mm puts the tip of a 20 degree point about 29 mm from the original point. Why?

A. The offset is applied to the shape's bounding box, which grows fastest at a corner
B. The miter limit multiplied the offset by four at that angle
C. Corner rounding added a radius equal to the offset distance
D. The parallel edges only meet at the offset divided by sin of half the angle

4 Six booleans in a row have tripled the node count of a mark. Where did the extra nodes come from?

A. Union adds a node at the centre of every shape it merges
B. Every intersection between the inputs becomes a node in the result
C. The simplify pass runs automatically after a boolean and inserts smoothing points
D. The fill rule changed to even-odd, which stores both subpaths

5 Why rotate once by 180 degrees rather than twelve times by 15?

A. Each rotation is a matrix multiply, so coordinates round twelve times
B. Applications cap the total rotation they will record on an object
C. Repeated rotation converts the object into a compound path
D. The pivot returns to the page origin after each rotation, moving the shape

6 Two rectangles with 2 mm strokes were snapped together by their bounding boxes and print 2 mm apart. Why?

A. The strokes were set to scale with the object, changing the geometry
B. Snapping rounded both positions to the nearest grid division
C. The pixel grid moved each edge by half a pixel in opposite directions
D. A bounding box includes the stroke, which extends 1 mm each side

MODULE 3

Type, which is also vector

Letters are paths in somebody else's file, delivered under a licence, shaped by an engine you did not write. What is inside a font, what your licence actually permits, how to kern a wordmark by hand, how to set a paragraph that reads, how type behaves on a curve, why Devanagari and other complex scripts break in ways Latin never does, and what a variable font does to a press.

> **BY THE END OF THIS MODULE YOU CAN**
>
> Set, space and convert type in a vector file without breaking it — choose a typeface you are licensed to use for the job in hand, correct spacing at the pair and the paragraph level, handle a complex script correctly, and decide per job whether to outline, embed or subset, stating the consequence of each to the printer.

18 · 9 min

Why the printer asked you to outline the fonts

THE ONE THING TO KEEP

Outline text so a missing font cannot reflow your layout, but keep an editable master and embed rather than outline for body text, because thousands of glyph paths choke a RIP.

The message, and what is behind it

"Curve karke bhejo." "Please convert text to curves." "Outline the fonts." Every print shop sends this, and most designers do it without knowing why, which means they also do it in the places where it is a mistake.

Here is the mechanism. Text in a design file is not a picture of letters. It is a string of characters plus a reference to a font: *set this in Poppins SemiBold*. The shapes live in the font file on your computer. Open that design on the shop's machine, which does not have Poppins, and the application substitutes something else. The letters change width. The line breaks move. A headline that fitted on two lines becomes three, a logo gets wider than its box, and nobody notices until 2,000 copies are printed.

Converting text to curves replaces the reference with actual geometry. The letters become paths. There is no font to be missing.

- CorelDRAW: **Ctrl+Q**, Convert to Curves
- Illustrator: **Shift+Ctrl+O**, Type > Create Outlines
- Inkscape: **Shift+Ctrl+C**, Path > Object to Path
- Affinity Designer: Layer > Convert to Curves

The cost, which nobody mentions

Once outlined, the text is not text. No spellcheck, no search, no editing, no restyling. A client asking to change a phone number means redrawing it.

So the working rule is: **keep an editable master, and export an outlined copy.** Name them so a tired person at 11pm cannot confuse them:

```
poster–diwali–v3–EDITABLE.cdr
poster–diwali–v3–OUTLINED.cdr
poster–diwali–v3–PRESS.pdf
```

When outlining is the wrong answer

Outlining a 64-page book is a bad idea, and it is a mistake people make trying to be careful. Each glyph becomes a path of maybe 30 to 90 nodes. A page of body text goes from a few hundred font references to tens of thousands of curves. The file balloons, the RIP crawls, and jobs time out on older equipment.

For body text, **embed the fonts** instead. A PDF export embeds a subset of each font — only the glyphs used — and the press sees exactly your typeface with none of the node explosion. Every PDF/X preset does this by default. There is a permission bit inside font files controlling whether embedding is allowed; nearly all commercial and open fonts permit it, but a small number do not, and your export dialog will complain if you hit one.

Outline for logos, headlines, short display text, and anything going to a cutting machine. Embed for everything else.

Outline it, or embed it

Outline it

- Logos, wordmarks and headlines
- Anything going to a cutter or a laser
- Short display text nobody will edit
- Cost: no spellcheck, no search, no resetting
- Keep an editable master beside it

Embed it

- Body text, and any long document
- Only the glyphs you used travel
- The RIP sees your typeface, not curves
- Cost: refused if the font's fsType forbids it
- Every PDF/X preset does it already

Both remove the missing-font risk. Outlining replaces the reference with geometry; embedding carries a subset of the font itself. The deciding number is how many glyphs there are, because each one becomes thirty to ninety nodes when it is outlined.

Fonts you can legally use, free

Body text in a paid job with a pirated font is a real risk, and "I downloaded it from a free fonts site" is not a defence. Many aggregator sites host cracked cuts of commercial faces.

Safe sources: **Google Fonts** (almost all under the SIL Open Font Licence — free for commercial use, embeddable, and you can sell what you make), **Fontshare** from the Indian Type Foundry, **Ek Type** for open Devanagari and Gujarati, **The League of Moveable Type**, and Adobe's own open-source releases like Source Serif. Read the licence file that ships with the font once; it takes two minutes.

For Indic and other complex scripts, test the actual words before committing. Correct rendering of conjuncts depends on OpenType shaping tables and on the application's text engine, and support varies across Inkscape, Illustrator and CorelDRAW versions. Type your client's real name in the real font at the real size, then look at it. Do not test with Latin placeholder text.

Kerning, tracking, and where they show

Tracking is uniform space applied across a range of characters. **Kerning** is the space between one specific pair. They are different jobs.

Bad kerning is invisible in body text and glaring in a headline or a logo, because the flaw scales with the type. The pairs that break most often are the ones where a diagonal or an open shape meets something else: **AV**, **To**, **Ya**, **LT**, **P.** followed by a full stop, and any capital next to a quotation mark.

Most fonts ship a kerning table the designer wrote — that is **metrics** kerning. Some applications also offer **optical** kerning, which ignores the table and spaces by looking at the shapes. For a well-made text face, metrics is better. For a cheap or free display face with a thin kerning table, optical is usually better. Illustrator and InDesign expose the choice in the Character panel.

Adjusting a single pair by hand:

- Illustrator and InDesign: click between the two letters, then **Alt+Left/Right**

- Inkscape: place the text cursor between them, **Alt+Left/Right**
- CorelDRAW: Shape tool (**F10**) on the text, select the character node, and drag — or use the interactive spacing arrows at the corners of the text block

Never use a "faux bold" or "faux italic" button. It thickens or slants the outline mechanically, which fattens curves unevenly and slants the counters wrongly. Use the real weight the family ships.

Type on a path

Illustrator has a Type on a Path tool. Inkscape: select text and a path, then Text > Put on Path. CorelDRAW: hover the Text tool over a curve until the cursor changes, or Text > Fit Text to Path.

The honest problem: on a tight circle, letters splay apart at the top and jam together at the bottom of an inner curve, because the spacing is measured along the baseline while the letter bodies sit outside or inside it. The fix is a larger radius, more tracking, and baseline shift to move the text off the path — not scaling the letters, which changes their weight. For a circular badge, set the bottom text on a second, separate circle drawn in the opposite direction so it reads left to right rather than upside down.

Before you send

Open the outlined copy on a machine that does not have your fonts. If it looks identical, you are done. That test takes thirty seconds and catches the failure this whole lesson is about.

BEFORE YOU MOVE ON

A designer outlines every piece of text in a 64-page book, including the body copy, to be safe. The press reports the file is extremely slow to process and one job times out. Why?

A. Outlining strips the font's hinting, so the RIP has to recompute hinting for every glyph at output resolution.
B. Outlined text can no longer be compressed inside the PDF, so file size grew and transfer, not processing, is the bottleneck.
C. Outlining forces the text into a transparency group that must be flattened, and the flattening is what slows the RIP.
D. Every glyph became a path of dozens of nodes, so each page went from a few hundred font references to tens of thousands of individual curves to render. Body text should have embedded fonts; outlining is for logos, headlines and files going to a cutter.

19 · 9 min

What is actually inside a font file

THE ONE THING TO KEEP

A font is a set of tables — outlines, a character map, widths, kerning, substitutions and a permission bit — and \`pdffonts\` plus \`fonttools\` will tell you in seconds what a font menu never will.

A font is a database with drawings in it

A `.ttf` or `.otf` file is a container of tables. Five of them decide everything you experience.

- **`glyf` or `CFF `** — the outlines. TrueType (`glyf`) stores **quadratic** Béziers: one handle per segment. PostScript/OpenType-CFF stores **cubic** Béziers: two handles, the same curves your drawing program uses.

- `cmap` — the map from character codes to glyphs. `U+0041` → glyph 36. Without this, the font has shapes but nobody can type them.
- `hmtx` — advance widths. How far the pen moves after each glyph.
- `kern` or `GPOS` — the adjustments between specific pairs.
- `GSUB` — substitutions: ligatures, small caps, alternates, and all the conjunct machinery that complex scripts depend on.

Plus `OS/2`, which carries the licensing flags, and `name`, which carries the family and style names the print shop will see in a font list.

The tables inside a font file

glyf or CFF	The outlines. TrueType quadratic, CFF cubic
cmap	Character code to glyph. Without it nobody can type it
hmtx	Advance widths: how far the pen moves each time
kern or GPOS	Adjustment between one specific pair and the next
GSUB	Ligatures, small caps, and every conjunct
OS/2	fsType, the bit deciding whether it may embed at all
name	The family name the print shop's font list will show

A font is a database with drawings in one of its tables. Two command lines read all of this: ttx lists the tables and prints the name and OS/2 entries, and pdffonts tells you which of them made it into a PDF you have already exported.

Cubic versus quadratic, and why it matters once

Convert a TrueType glyph to outlines and you get quadratic curves expressed as cubics — more nodes than the designer drew, with handles at positions nobody chose. Convert a CFF glyph and the nodes come across as they were drawn.

For body text this is invisible. For a wordmark you are about to edit by hand, it is the difference between eight nodes on an **o** and fourteen. If you have the choice, take the OTF.

Hinting, and why outlining changes the shape slightly

A TrueType font carries **hints** — small programs that nudge outlines onto the pixel grid at small sizes so stems stay the same width and baselines stay flat. Hinting is why body text is legible at 11 px on a screen.

Hints are a *rendering* instruction. Convert text to outlines and they are discarded, because there is no longer any text to hint. On paper this does not matter; a press has no pixel grid. On screen it does, which is why an outlined SVG logo can look subtly muddier at 16 px than live text would.

Formats you will meet

- **TTF** — TrueType. Quadratic. Universal.
- **OTF** — OpenType, usually with CFF cubic outlines. Preferred for print.
- **TTC** — a collection, several fonts in one file. Some applications import only the first.
- **WOFF / WOFF2** — compressed wrappers for the web. Not installable on most systems as-is; convert with `fonttools` if you own the licence to do so.
- **Type 1 (.pfb)** — the old PostScript format. **Adobe ended support in January 2023**, and current Illustrator and InDesign no longer render them. If you inherit a 2005 archive, this is why the text is missing.
- **Variable fonts** — one file with continuous axes, covered later in this module.

Read a font from the command line, free

`fonttools` is a Python package and the fastest way to answer questions the font menu will not.

```
pip install fonttools
ttx -l Poppins-SemiBold.ttf          # list the tables
ttx -t name -o - Poppins-SemiBold.ttf | head -40   # family,
style, licence URL
ttx -t OS/2 -o - Poppins-SemiBold.ttf | grep fsType
```

`fsType` is the embedding permission bit, and it is worth understanding because your PDF export obeys it:

- `0` — installable, no restriction
- `2` — restricted: embedding not permitted
- `4` — preview and print embedding allowed
- `8` — editable embedding allowed

A font with `fsType 2` will refuse to embed and your export dialog will complain. Nearly everything commercial and everything under the SIL Open Font Licence is `0` or `4`.

And to see what is inside a PDF you have already made:

```
pdffonts job.pdf
```

Every font, whether it is embedded, and whether it is subset. A line with an empty "emb" column is the failure this whole area exists to prevent.

The practical consequence

When a shop says "we do not have that font", the honest answers are: send a PDF with it embedded, send outlines, or send them the font file — and the third is a licensing act, not a technical one. Which of those you may do is the next lesson.

One more thing worth knowing about embedding: what goes into a PDF is normally a **subset**, containing only the glyphs you actually used, renamed with a six-letter prefix like `ABCDEF+Mukta-Regular`. That prefix in a `pdffonts` listing is how you tell a subset from a full embed at a glance. Subsetting keeps files small and is why a 40-page document in a 400-glyph face adds a few tens of kilobytes rather than a megabyte. It also means nobody can extract a usable font from your PDF, which is part of why foundries permit it.

BEFORE YOU MOVE ON

A designer converts a headline set in a TrueType font to outlines and notices the letterforms look very slightly different at small sizes on screen, though the printed result is fine. What changed?

A. The quadratic curves were converted to cubic, which introduces a small approximation error in every segment.
B. Hinting instructions were discarded, so the outlines are no longer snapped to the screen's pixel grid.
C. The `cmap` table was dropped, so the renderer fell back to a substituted glyph for some characters.
D. Outlining applies the font's default tracking as real spacing, which shifts each glyph by a fraction of an em unit.

20 · 9 min

The licence, which is the part that bites

THE ONE THING TO KEEP

Desktop, web, app and broadcast are licensed separately; "free for personal use" excludes client work; outlining is not a licensing workaround; and sending a printer the font file is distributing software, which is a different permission from using it.

Four different permissions, sold separately

A typeface is software. What you bought is a licence, and it is normally split into uses that have nothing to do with each other.

- **Desktop** — installing it on a machine and setting type with it. Usually sold per seat or per workstation. This is the one you need for print work.
- **Webfont** — serving it from a website, usually priced by monthly pageviews.

- **App / embedded** — putting it inside software, a game, a kiosk or a device.
- **Broadcast / e-publication** — video, ads, ebooks.

Buying desktop does not give you web, and vice versa. A designer who licenses a face for their own machine, delivers the source file to the client, and the client installs it on six machines, has put the client in breach. Nobody intended anything dishonest and the invoice is still real.

"Free for personal use" means what it says

The largest category of trouble. Free-font aggregator sites host enormous numbers of faces marked *free for personal use*, which excludes anything a client pays you for. Many of them also host cracked cuts of commercial families under slightly altered names.

Two tests before you use a free font on paid work:

1. **Find the actual licence file.** A real open font ships an `OFL.txt` or a `LICENSE` alongside it. No licence file, no job.
2. **Check the `name` table's licence URL.** `ttx -t name -o - font.ttf` prints it. A commercial foundry's URL on a file you downloaded free is the signature of a cracked font.

What you can safely use, for anything

- **Google Fonts** — nearly all under the **SIL Open Font Licence**. Free for commercial use, embeddable, and you may sell the work you make. You may not sell the font itself, and if you modify it the modified version must also be OFL and must not use a reserved font name.
- **Fontshare**, from the Indian Type Foundry — free for personal and commercial use, and the quality is high.
- **Ek Type** — open Devanagari, Gujarati and Latin families.
- **The League of Moveable Type**, **Velvetyne**, **Collletttivo** — open foundries with real design behind them.

- **Adobe's open-source releases** — Source Serif, Source Sans, Source Code.
- **Noto**, from Google — covers nearly every script in Unicode. Not always the most characterful choice, and always a correct one.

The OFL's one practical trap: if you modify an OFL font — changing outlines for a wordmark, say — you must rename it. Reserved Font Names exist precisely so a modified file cannot masquerade as the original.

Outlining is not a licensing workaround

A common belief: converting text to outlines means the licence no longer applies, because there is no font in the file. It does not follow. The outlines are still the foundry's drawings, and most desktop licences explicitly permit output as outlines in a final artwork file while forbidding redistribution of the font itself. Read the one paragraph; foundries are generally reasonable about exactly this case.

Where outlining *does* change things is in what you hand the printer. Outlines mean you are sending artwork. Sending the `.ttf` means you are distributing software, and that usually needs the licence to cover it or a separate "output provider" clause. Most licences have one. Check before you attach the file.

When a client supplies a font

They send you a `.otf` with the brief. Ask two questions in one email:

> *Do you hold a desktop licence for this typeface, and does it cover me as your supplier for the duration of this job?*

If the answer is no or unclear, the safe move is to set the job in a licensed alternative and show them both. You are not being difficult; you are keeping their exposure off your invoice.

The practical record

Keep a `fonts.txt` in every project folder:

```
Body      Mukta Regular / SemiBold   SIL OFL 1.1   fonts.-
google.com/specimen/Mukta
Display   Cabinet Grotesk Bold       Fontshare     fontshare.-
com/fonts/cabinet-grotesk
```

Two minutes per job. It answers the question a year later when somebody needs to reset a page, and it is the document you point at if anybody ever asks.

BEFORE YOU MOVE ON

A designer licenses a commercial typeface for their own workstation, sets a client's 64-page catalogue in it, and sends the printer the InDesign package — which includes the font files, as packaging always does. What is the actual problem here?

A. The catalogue's text should have been outlined, because a desktop licence never permits output in any embedded form.

B. Packaging embeds a subset rather than the whole font, and subsetting a commercial font without permission modifies it.

C. Nothing is wrong, because the printer is acting as an agent of the licensee and inherits the licence automatically.

D. Handing over the font files distributes the software, which a single-seat desktop licence usually does not cover.

21 · 9 min

Kerning a wordmark by hand

THE ONE THING TO KEEP

Kern by equalising the white *area* between letters, judged three letters at a time and confirmed by blurring — and kern a reversed wordmark separately, because light on dark closes the counters and reads tighter.

Why a wordmark always needs it

A typeface's kerning table was written for running text at reading size, where a small imperfection between two letters is invisible. A wordmark is one word at 40 mm, 400 mm or 4 m, and every imperfection scales with it.

So: set the word, then fix it. Every logotype you admire has been through this, usually for hours.

The method that works: three letters at a time

Spacing is not a property of a pair. It is a property of the rhythm across the word. Judge it in threes.

1. Set the word in caps of the actual typeface at a large size — 200 pt on screen is fine.
2. Cover everything but the first three letters, with a rectangle or your hand.
3. Ask: is the gap on the left equal to the gap on the right? Not the *distance* — the **area** of white.
4. Slide the middle letter until they match. Move the frame one letter to the right. Repeat to the end.

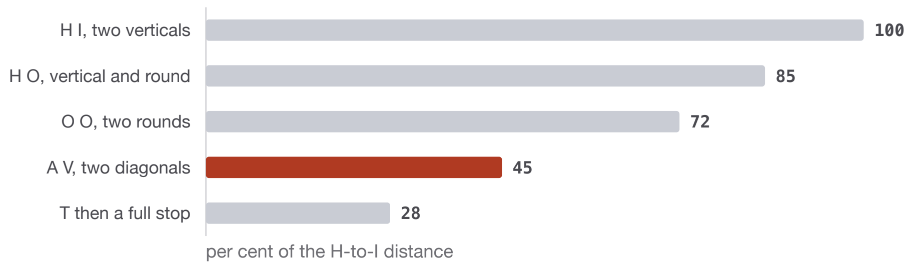
Shuffling the whole word and looking at it "as a word" does not work, because your eye reads it instead of measuring it. Three letters at a time forces measurement.

Why it is area and not distance

`A` and `V` present diagonals, so the white between them is a narrow wedge. `H` and `I` present two verticals, so the white is a full-height rectangle. Set them at the same *distance* and the AV gap looks enormous.

The ranking, loosest to tightest, for the same measured distance:

The same optical gap, in measured distance



Spacing is equal white area, not equal distance. Two verticals already have the least white between them and need the most room; two diagonals have a wedge of white and need far less. Set them all at the same distance and the AV looks like a hole.

- Two straight verticals — `HI`, `IN`, `ME` — need the most distance, because the white is already minimal.
- A vertical against a round — `HO`, `DE` — slightly less.
- Two rounds — `OO`, `CO` — less again.
- A diagonal against anything — `AV`, `LT`, `Ya`, `To`, `VA` — least, often dramatically so.
- Anything against a full stop, a comma, or a quotation mark — least of all.

The three tests

- **Blur it.** Squint, or apply a 3 px Gaussian blur to a screenshot in GIMP or Photopea. Even spacing becomes an even grey; a loose pair becomes a light hole and a tight pair a dark blob. This is the single most reliable test and it takes ten seconds.
- **Upside down.** Rotate the word 180°. You stop reading it and start seeing it.

- **Reversed.** Set it white on black. Light-on-dark letterforms appear heavier and the counters close up, so a wordmark kerned only on white will look tight when reversed. If both versions will be used, kern once for each and deliver both.

Doing it in each program

Before outlining, while it is still live text:

- **Illustrator / InDesign**: cursor between two letters, **Alt+Left/Right**. The step is set by Preferences ▸ Type ▸ *Size/Leading* and *Kerning* — default 20/1000 em, which is coarse. Set it to 5 for fine work. The Character panel shows the current value in thousandths of an em.
- **Inkscape**: text cursor between the letters, **Alt+Left/Right** moves by a small step; **Alt+Shift+Left/Right** by a larger one. The values go into the SVG as `dx` offsets.
- **CorelDRAW**: Shape tool (**F10**) on the text, select the node at the lower-left of a character, then arrow-key it — or drag the horizontal spacing arrow at the bottom right of the text block for overall tracking.

After outlining, you just move the glyph. Some people do all their kerning this way. It works and it is unrecoverable, so do it on a copy.

Metric, optical, and when each is right

- **Metric kerning** uses the font's own table. For a well-made text face, this is the designer's own judgement and is better than an algorithm.
- **Optical kerning** ignores the table and spaces by analysing the shapes. For a cheap or free display face with a thin table — and many are — it is a genuine improvement, and it is the sane default when you have mixed two faces in one line.

Illustrator and InDesign expose both in the Character panel. Inkscape and CorelDRAW use the font's metrics and leave optical to you.

Record what you did

If the wordmark stays as live text in a master file, write the kern values down in the project notes — `A→V −40`, `V→A −35`, `T→o −25`. Somebody will need to reset it at a different size, and the numbers are in thousandths of an em, so they scale with the type rather than with the millimetres.

BEFORE YOU MOVE ON

A designer kerns a white-on-black wordmark for a shopfront until the spacing looks even. The same artwork is later used in black on white in the brochure, where the spacing now looks slightly loose. Why?

A. Converting from a reversed to a positive version inverts the kerning values stored as `dx` offsets in the file.
B. Light shapes on a dark ground appear larger, so the letters looked heavier and the gaps tighter when kerned.
C. Ink spread on the shopfront vinyl widened every single stroke, and the brochure's litho printing does not spread in the same way.
D. The brochure uses a different optical size of the same family, which carries a different kerning table entirely.

22 · 9 min

Setting a paragraph somebody can read

THE ONE THING TO KEEP

Size, leading and measure carry most of readability — 45 to 75 characters a line, leading at 120 to 145% — and justified text needs a long measure plus controlled H&J values or it develops rivers.

Three numbers decide almost everything

Size, leading, measure. Get these right and mediocre type looks competent. Get them wrong and no amount of typeface choice rescues it.

- **Size.** Body text in a book or brochure is 9–11 pt. In a wall-mounted notice it is whatever is legible at the distance. On a phone it is 16 px, because that is the browser default for a reason.
- **Leading** — line spacing, measured baseline to baseline. Start at **120–145%** of the size. 10 pt type takes 13 or 14 pt leading. The looser end for a long measure or a large x-height face, the tighter end for short lines.
- **Measure** — the line length. **45 to 75 characters**, counting spaces. Below 45, the eye jumps back too often and the rag gets ugly. Above 75, the return sweep loses its place and readers skip lines.

Measure and leading interact: a long measure needs more leading, because the return sweep is longer and the eye needs a clearer target.

To count your measure without counting: set a line of lowercase alphabet twice — `abcdefghijklmnopqrstuvwxyzabcdefghijklmnopqrstuvwxyz`. Roughly two and a half alphabets is the right column width.

Ragged right or justified

Ragged right — one word space everywhere, uneven right edge. Almost always the better choice, and always the better choice at a narrow measure.

Justified — word spaces stretched to fill each line. Requires hyphenation on, and requires a measure long enough that stretching a few spaces is not visible. Justified text at a 45-character measure produces **rivers**: vertical channels of white running down the paragraph, which you see instantly once you have been told to look.

If you justify, set the H&J values rather than accepting the defaults. In InDesign, Paragraph panel ▸ Justification:

- Word spacing: minimum 80%, desired 100%, maximum 133%
- Letter spacing: minimum 0%, desired 0%, maximum 4% — never more; stretched letterspacing in body text is worse than an uneven rag
- Glyph scaling: 98/100/102% at most

InDesign's **Adobe Paragraph Composer** looks at the whole paragraph rather than one line at a time and produces markedly better justification than the single-line composer. It is on by default in recent versions; check, because it is the largest single improvement available for free. Scribus has no equivalent; LibreOffice has none; LaTeX has had one since 1982 and is still the best justification available anywhere.

Fixing a rag

A good rag is a soft, irregular wave. Bad rags have shapes:

- **A ladder** — three or more consecutive lines ending in a hyphen. Limit consecutive hyphens to 2.
- **A wedge** — lines getting steadily shorter or longer, which reads as a deliberate shape and is not.
- **A hole** — one very short line beside long ones.

Fix with a discretionary hyphen (**Ctrl+Shift+-**), a non-breaking space (**Ctrl+Alt+Space**) to keep a short word with its neighbour, or by editing the copy. Never fix a rag by tracking one paragraph; the change in colour is more visible than the rag was.

Widows and orphans

- An **orphan** is a lone first line of a paragraph stranded at the bottom of a column.
- A **widow** is a lone last line carried to the top of the next column — or, in the stricter sense, a single word alone on the last line of a paragraph.

Set *Keep Options* ▸ Keep with next: 2 lines, and start/end lines minimum 2. In Scribus this is in the paragraph style's *Distances and Alignment*. Fix a single-word last line with a non-breaking space between the last two words.

Optical margin alignment

Punctuation and the round or diagonal parts of letters make a left or right edge look ragged even when it is mathematically flush. **Optical margin alignment** hangs quotes, hyphens and the overhang of `o`, `T` and `W` slightly outside the column so the edge *looks* straight.

InDesign: Story panel ▸ Optical Margin Alignment, set to the type size. It is one checkbox, it is free, and it is the difference people notice without being able to name.

The free path

Scribus does styles, H&J, keep options and hyphenation properly, and is genuinely capable of a well-set book. LibreOffice Writer with real paragraph styles will set a readable novel interior. LaTeX sets the best justified text of anything on this list and is the hardest to control visually. All three are free; the thing you cannot get for nothing is Adobe's paragraph composer and optical margins.

BEFORE YOU MOVE ON

A 10/12 pt justified column is set at a 40-character measure with default justification settings. Vertical channels of white appear running down the paragraph. What is the mechanism, and what is the smallest effective change?

A. Leading is too tight for the size, so lines crowd and the white between words appears to connect vertically.
B. A short measure forces a few word spaces to stretch a long way, and oversized spaces on successive lines align vertically.
C. Hyphenation is switched off, so long words are pushed whole to the next line and leave large gaps behind them on every line.
D. The typeface has unusually wide default word spacing, and reducing the word space in the character panel will remove the channels.

23 · 8 min

Type on a curve, and why it splays

THE ONE THING TO KEEP

Type on a path is positioned by arc length along the baseline while the glyph is rigid, so letters splay on the outside of a curve and crowd on the inside — fixed by a larger radius, more tracking, and a second reversed circle for the bottom of a badge.

What the software is actually doing

Type on a path is positioned by **arc length along the baseline**. The engine walks the path, placing each glyph's origin at the distance the previous glyph's advance width demands, and rotates the glyph to the path's tangent at that point.

Every problem with curved type follows from one fact: **the glyph has height, and the baseline does not.**

On a circle of radius r, the baseline has circumference 2πr. The top of a capital, at cap height h above the baseline, sits on a circle of circumference 2π(r+h). Those two circles have different lengths, but the letter is rigid. So:

- **Text on the outside of a curve** splays: the tops of the letters are pulled apart, the bottoms crowd together.
- **Text on the inside** does the reverse: the tops jam, the bottoms fan out.

The tighter the radius relative to the cap height, the worse it is. This is not a bug and no setting fixes it.

The four things that do help

1. **Increase the radius.** The ratio that matters is cap height to radius. At r = 20× cap height the splay is barely visible; at r = 3× it is unusable.
2. **Add tracking.** Splay is most obvious when letters nearly touch at the bottom. Open the whole line by 30–80 thousandths of an em and the crowding disappears; the top-heavy look remains but stops mattering.
3. **Choose the face.** A typeface with short capitals relative to its width — a wide, low grotesque — splays less than a tall condensed one, because h is smaller relative to the advance width.
4. **Outline and adjust individually.** For a badge that must be right, convert to curves and rotate each letter about its own optical centre. Twenty minutes, and it is the reason good circular badges look good.

Baseline shift, which is the control people miss

Type sits *on* the path by default. **Baseline shift** moves it up or down relative to the path without changing the path.

- Illustrator: Character panel, the `A↕` field, or **Alt+Shift+Up/Down**.
- Inkscape: Text and Font dialog, or select the characters and use the vertical-shift field on the text toolbar.
- CorelDRAW: the *Distance from path* field on the property bar when text is fitted to a path.

Use it to lift text off a circle so the letters' descenders clear it, or to push text inside a ring.

The bottom of a circular badge

The classic problem: text on the bottom half of a circle reads upside down.

The fix is not to flip the text. It is to **draw a second circle, in the opposite direction**, and set the bottom text on that one.

1. Draw circle A, clockwise. Put the top text on it.
2. Duplicate it, reverse its direction (Inkscape: **Shift+R**; Illustrator: Attributes ▸ Reverse Path Direction), and make it slightly smaller so the two lines of text sit at the same visual distance from the centre — because the bottom text will hang below its baseline while the top text sits above.
3. Put the bottom text on circle B and set it to align centred at the 6 o'clock position.

Hide both circles by setting their stroke and fill to none. They stay in the file and the text stays live.

Moving text along the path

Every implementation gives you a bracket or handle you drag to move the start point, and dragging it across the path flips the text to the other side. That flip is the source of most "my text suddenly went inside the circle" confusion. In Illustrator the handle is the small perpendicular bar at the start of the text; double-clicking it opens Type on a Path Options, which has **Align to path: Ascender / Descender / Centre / Baseline** and an *Effect* dropdown with Rainbow, Skew, 3D Ribbon, Stair Step and Gravity. Rainbow is the default and the only one you usually want.

What to do before sending it anywhere

Convert to curves, and check the shape of every letter afterwards. Some text engines apply the rotation at render time in a way that does not survive export cleanly, and CorelDRAW in particular has historically produced slightly different spacing on curved text between the on-screen view and the PDF. Outline it, look at it, then send it.

BEFORE YOU MOVE ON

A designer sets a shop name around the top of a 30 mm-radius circular badge in a tall condensed face at 8 mm cap height. The letters fan apart badly at the top and nearly touch at the bottom. Which change addresses the mechanism most directly?

A. Switch to a wider face with a lower cap height and add tracking, so the ratio of cap height to radius falls.
B. Apply baseline shift to lift the text further from the path, which restores even spacing between the letters.
C. Set the type on a path drawn as a polygon with many sides rather than a true circle, so each letter sits on a straight segment.
D. Change the Type on a Path effect from Rainbow to Skew, which keeps the letters vertical and removes the fan.

---

24 · 10 min

Devanagari, Arabic, and the scripts that break

THE ONE THING TO KEEP

Complex scripts need a shaping engine that forms conjuncts and reorders matras, so the same text can be correct on screen and wrong in the PDF — test with a real string of hard cases, have a native reader check it before outlining, and never use legacy non-Unicode fonts.

Latin is the easy case

In Latin, a string of characters maps almost one-to-one onto a string of glyphs, left to right. Almost every text engine gets it right.

Most of the world's scripts do not work that way, and the failures are specific and recognisable.

What shaping means

Between "the user typed these characters" and "draw these glyphs here" sits a **shaping engine**, which consults the font's `GSUB` and `GPOS` tables and applies rules. For Devanagari the rules include:

- **Conjuncts.** क + ् + ष becomes क्ष — one glyph, not three. There are hundreds of these and the font must contain them.
- **Reordering.** The vowel sign ि is typed *after* its consonant and drawn *before* it. कि is stored as क then ि and rendered with the matra on the left.
- **Matra positioning.** Vowel marks attach above, below, left or right, and their exact position depends on the consonant's width and on whether a conjunct formed.
- **Reph.** र followed by a virama moves to the top right of the following cluster as a small hook.

Arabic adds contextual forms — each letter has isolated, initial, medial and final shapes — plus right-to-left order and vertical stacking in some styles. Tamil, Kannada, Telugu, Bengali, Khmer and Thai each have their own rules.

HarfBuzz is the open-source shaping engine that does this correctly, and it is what LibreOffice, Firefox, Chrome, Android, Scribus and Inkscape 1.x use. Applications that use an older engine, or their own, get it wrong in visible ways.

The failures you will see

- **Boxes** (`□`) or the missing-glyph rectangle: the font has no glyph for that character. Change the font.
- **Conjuncts split into separate letters with a visible virama**: the shaping ran but the font lacks the conjunct, so it fell back to the explicit half-form. Change the font.

- **Matras in the wrong place, or detached**: the engine did not reorder. This is an application problem, not a font problem.
- **Correct on screen, wrong in the PDF**: the export path used a different engine from the editing view. This is the one that costs money, because it is invisible until the proof arrives.

The test you must actually run

Never test a complex script with placeholder text. Type the client's real name, the real address, the real headline, in the real font, at the real size, and look at it.

Build a test string that exercises the hard cases. For Devanagari:

```
क्ष त्र ज्ञ श्री हिन्दी कार्यालय द्वितीय पुस्तक
```

Those cover a three-consonant conjunct, a reph, a left-side matra and a below-base form. If all of them render correctly, the combination of font and application is sound. If any of them is wrong, you have found it in thirty seconds rather than after printing.

Outlining a complex script

This is where jobs are lost. Converting to curves freezes whatever the shaping engine produced. If the engine was wrong, you have now made the error permanent and undetectable — the file no longer contains characters anyone can proofread, only shapes.

So the order is:

1. Render it.
2. **Have a native reader check it on screen**, at size, before you outline anything.
3. Outline.
4. Check the outlined version against the live version, side by side, letter by letter.

Step 2 is not optional and it is not something you can do yourself unless you read the script. A wrong matra is not a typo a non-reader can see.

The only safe order for a complex script

Set it live	→	**A reader checks**	→	**Outline**	→	**Compare**
Real words, real font, real size		Somebody who reads the script, on screen		Only now, and the shaping is frozen		Outlined against live, letter by letter

Outlining freezes whatever the shaping engine produced. After step three the file holds shapes rather than characters, so a wrong matra can no longer be proofread by anybody, in any application, ever.

Font sources that work

- **Noto** — Google's family, covering essentially every script in Unicode, OFL, well-shaped.
- **Ek Type** — Mukta, Baloo and others, designed for Devanagari and Gujarati first rather than Latin-first.
- **Indian Type Foundry / Fontshare** — high-quality Indic and Latin.
- **Google Fonts' Indic section** — Hind, Rozha, Yantramanav, Tiro Devanagari.

Avoid legacy non-Unicode fonts entirely — Kruti Dev, Chanakya, Shree-Lipi and similar map Devanagari shapes onto Latin codepoints. Text set in them is not Devanagari at all; it is Latin letters wearing a costume. It cannot be searched, spellchecked, copied into anything, or converted without a mapping tool. A great deal of Indian government and legal material is still typed this way, and converting it is real paid work — tools like the free `Unicode Converter` scripts and several web converters handle the common mappings.

In each program

Inkscape 1.x and Scribus use HarfBuzz and are dependable. LibreOffice is dependable. Illustrator needs the **World-Ready Composer** enabled — Paragraph panel menu ▸ *Middle Eastern & South Asian Composer*; without it, Indic text will not shape. CorelDRAW's support varies markedly by version and is the single most common source of broken conjuncts in Indian print work. Test your own version with the string above before you promise anybody anything.

BEFORE YOU MOVE ON

A Hindi wedding card renders correctly in the designer's editing window, but the exported PDF shows the ि matra after its consonant rather than before it. The font contains every required glyph. Where is the fault?

A. The font lacks the reordering rules in its `GSUB` table, which only load when the font is embedded rather than installed.
B. The PDF export path used a different text engine from the editing view, and that one did not apply the reordering rules.
C. The text was outlined on export, and outlining always freezes glyphs in their stored character order.
D. The PDF was exported in a version that predates Unicode support for Devanagari, so the characters fell back to code order.

25 · 8 min

Figures, dashes and the characters nobody checks

THE ONE THING TO KEEP

Quotes, the three dashes, tabular versus proportional figures, the rupee sign's patchy coverage and silent font substitution are the details that separate set type from typed text — and each has a specific check you can run in seconds.

The small marks that make work look amateur

Type is mostly letters, and the things that give away an untrained setting are mostly not letters.

Quotes

`'` and `"` are typewriter marks. Real typography uses `'` `'` and `"` `"` — and in some languages `„ "` or `« »`.

Turn on smart quotes and then check the exceptions, because smart-quote logic guesses from context and guesses wrong on an elided year (`'09` needs a right single quote, not a left one) and on a measurement (`5' 6"` needs the straight primes `′` and `″`, which are different characters again).

Where: InDesign ▸ Preferences ▸ Type ▸ Use Typographer's Quotes. Inkscape has none; type them directly or paste. On Linux and with a compose key set, `compose " <` gives `"`.

Dashes

Three different characters for three different jobs:

- **Hyphen** `-` joins words and breaks them at line ends. `well-made`.
- **En dash** `–` (the width of an `n`) spans a range and joins equal things. `1998–2004`, `Delhi–Mumbai`. **Alt+Shift+-** in InDesign, `U+2013`.
- **Em dash** `—` (the width of an `m`) breaks a sentence. `She paused — then agreed`. **Alt+Shift+Ctrl+-**, `U+2014`.

Using a hyphen for a date range is the single most common typographic error in print, and it is one keystroke to fix.

Figures

Most OpenType text faces carry more than one set of numerals:

- **Lining figures** — all the height of a capital. Correct in tables, in all-caps settings, and on a form.
- **Oldstyle (text) figures** — with ascenders and descenders, like lower-case. Correct in running prose, where lining figures shout.
- **Tabular** — every figure the same width, so columns line up. Essential in a price list, a timetable, any set of numbers stacked vertically.
- **Proportional** — each figure its natural width. Better in a sentence.

A price list set in proportional figures will not align, no matter how many tabs you add, because the `1` is narrower than the `8`. Switch to tabular figures and the problem disappears. InDesign: Character panel menu ▸ OpenType ▸ Tabular Lining. In CSS it is `font-variant-numeric: tabular-nums`. Inkscape exposes OpenType features in the Text and Font dialog's *Features* tab in 1.x.

Currency and the rupee sign

`₹` is `U+20B9`, adopted in 2010. Fonts released before then do not have it, and fonts released after do not all have it. When it is missing you get a box, or worse, a silent fallback to a different typeface for that one character — so the price is set in Arial while the rest of the line is set in your display face.

Check before you commit to a face. Set `₹1,29,999` and look at it. Note also the Indian digit grouping: two, two, three from the right, not three-three-three. `12,34,567` not `1,234,567`. No software does this automatically; you type it.

The missing-glyph fallback, which hides itself

When a font lacks a character, applications do one of two things and only one of them is honest.

- **Show `.notdef`** — the empty or crossed box. Obvious, easy to fix.
- **Silently substitute another font** for that character. The text looks fine at a glance and one character is in the wrong typeface.

InDesign highlights substituted glyphs in pink if Preferences ▸ Composition ▸ *Highlight Substituted Glyphs* is on. Turn it on. In other applications, the check is `pdffonts` on the exported PDF: if a font you did not choose is listed, something substituted.

Ligatures, and when to switch them off

`fi`, `fl`, `ffi` are drawn as single glyphs so the `f` hood does not collide with the dot of the `i`. Leave them on for body text.

Switch them off when you are tracking a word heavily — a ligature does not separate, so a tracked-out `FLAT` keeps its `fl` welded together — and check them in a script or display face, where discretionary ligatures can be applied automatically and produce something you did not intend in a client's name.

The five-second check

Before any file goes out: search the text for a straight apostrophe, search for a hyphen with spaces around it, and set one line of figures to see whether they align. Three searches, and they catch most of what an experienced eye would notice immediately.

BEFORE YOU MOVE ON

A price list is set with tab stops, but the numbers in the column still do not line up — some rows sit slightly left, others slightly right. The tabs are correctly set to right-align. What is actually wrong?

A. The tab character is being typeset at the font's default space width rather than advancing to the stop.
B. The paragraph is justified, so word spacing is being stretched differently on each line.
C. The font is using proportional figures, so each numeral has a different advance width and the digits do not stack.
D. The figures are oldstyle, so some digits carry descenders and appear to sit at slightly different heights in the column.

26 · 8 min

Variable fonts, and what a press does with one

THE ONE THING TO KEEP

A variable font is instanced — frozen at your axis values and embedded as an ordinary static font — before it reaches a press, so it is safe for print if you verify the export, and its `wdth` axis is the only legitimate way to condense a headline.

One file, a continuum of weights

A variable font stores one set of outlines plus **deltas** — instructions for how each point moves as an axis changes. Ask for weight 437 and the renderer interpolates between the designer's masters and draws a shape that exists in no static file.

The registered axes have four-letter tags:

- `wght` — weight, usually 100–900
- `wdth` — width, as a percentage
- `opsz` — optical size, which changes stroke contrast and spacing for small versus large setting
- `ital` and `slnt` — italic and slant

Plus any custom axis the designer invented: grade, serif shape, terminal angle.

`opsz` is the one worth caring about in print and the one nobody uses. A face designed at 9 pt wants sturdier strokes, wider spacing and lower contrast than the same face at 60 pt. Before variable fonts this needed separate cuts — Display, Text, Caption — which most families never shipped. With an `opsz` axis you get the correct optical size at every point size, automatically. Inter, Source Serif 4, Literata and several Noto families carry it.

What happens at the RIP

This is the part that decides whether you use one.

A PDF cannot contain a live variable font with axes a RIP will resolve. When you export, the application **instances** the font — freezes it at the values you set — and embeds that instance as an ordinary static font, usually CFF. Which means:

- The press sees a normal font. Nothing exotic reaches the RIP.
- The instance is a *new font*, generated by your application. Its name may be something like `Inter-VF_wght437`, which a printer's preflight may flag as unknown.
- Two applications may instance the same font slightly differently, so a headline exported from InDesign and one exported from Illustrator can differ by a hair.

Practically: variable fonts are safe for print, provided you check the exported PDF. `pdffonts job.pdf` will show you exactly what got embedded and whether it is subset.

Application support, honestly

- **InDesign and Illustrator** — support variable fonts with a slider in the Character panel. Reliable.
- **Affinity** — support added and improved across versions; check yours.
- **Inkscape** — variable font support landed in 1.x with a *Variations* tab in the Text and Font dialog. It works and it is less polished than the paid tools.
- **Scribus** — no variable-axis control at the time of writing. Install a static instance instead.
- **CorelDRAW** — support has been present since 2018 but varies; test.

If your tool cannot drive the axes, you are not stuck. Generate a static instance yourself with `fonttools`:

```
pip install fonttools
fonttools varLib.instancer Inter.ttf wght=550 wdth=100 -o
Inter-550.ttf
```

That produces an ordinary static font you can install anywhere. This is also the honest fix when a printer's workflow refuses the variable file.

When not to use one

- **When the shop will open your source file.** They will not have the axis settings, and a substituted static weight will not match.
- **When you cannot test the export.** If you cannot run `pdffonts` or open the PDF somewhere clean, use a static cut and remove the variable.
- **For a logo.** A wordmark should be outlined anyway, and a mark whose weight depends on an interpolation is a mark that can change when somebody updates a font file.

When they earn their place

- **A long document with many weights.** One file rather than nine, no risk of the wrong cut being installed, and consistent metrics.
- **Anything with an `opsz` axis**, where the typographic gain is real and visible.
- **Tight copy-fitting.** A `wdth` axis lets you condense a headline by 3% honestly, using outlines the designer drew, rather than by horizontally scaling the type — which is the thing you must never do and which is visibly different.

That last point is the strongest argument. Horizontal scaling to 92% thins the vertical strokes by 8% and leaves the horizontals untouched, so the contrast of the face changes and the round letters go oval. A width axis narrows the letter the way the designer intended it to narrow, redrawing the counters and keeping the stroke weights. It is the same instinct and only one of them is correct.

One caution on naming. Some applications write the instance name into the PDF with the axis values appended, and some write the family name unchanged. If a printer's preflight reports a font it does not recognise, that is usually all this is — but confirm it rather than assuming, because an unrecognised font name and a genuinely missing font look identical in a preflight report.

BEFORE YOU MOVE ON

A designer sets a headline with a variable font's width axis at 92% rather than horizontally scaling static type to 92%. Both look narrower. Why is only one of them acceptable?

A. Horizontal scaling is applied at render time and cannot be embedded in a PDF, so the press would receive the unscaled type.

B. A width axis keeps the file as live text while horizontal scaling forces the type to be outlined before export.

C. Scaling thins vertical strokes while leaving horizontals alone; a width axis uses outlines drawn for that width.

D. Variable instancing happens at 1000 units per em, so the width axis produces a mathematically exact result where horizontal scaling introduces rounding.

CASE STUDY · MODULE 3

The consent form, the Friday deadline and the reader who was away

A district hospital in Solapur, six thousand Marathi surgical consent forms, and a plate-making slot on Saturday morning.

Nandini works freelance and sets forms for three hospitals and a co-operative bank. She does not read Marathi.

The district hospital asked her to reset its surgical consent form, two pages, Marathi on one side and English on the other. The administrative officer sent the text as a Word file. The Marathi in it had originally been typed in Kruti Dev by a clerk in 2011 and converted to Unicode by somebody at some point since, using a tool nobody remembered the name of.

She set it in Mukta, which is an Ek Type family designed for Devanagari first, and built the layout in Scribus. On her own machine it rendered correctly. The shop that prints for the hospital runs CorelDRAW, on a version whose conjunct handling is known in the trade to be uneven, so she supplied a PDF rather than a working file.

The shop sent back a laser proof on Friday afternoon. Two things on it were wrong. One three-consonant conjunct had come apart into separate letters with a visible virama between them. One vowel sign was sitting on the wrong side of its consonant. Neither is a typo in the ordinary sense; a non-reader looking at the page would see nothing unusual at all, and Nandini could only see them because the shop's proofreader had circled them in pencil.

The shop now wanted curves. Convert the text to outlines, send it back tonight, and they would make plates at seven on Saturday morning, print over the weekend and deliver on Monday. The Health Department inspection was on Tuesday and the administrative officer had already told the district office the forms would be in place.

The hospital's matron is the person authorised to sign off the Marathi text. She reads it, she uses the form every day, and she was at a family function in Pandharpur until Monday morning. The ward clerk, who also reads Marathi, offered to look at it, but the officer was uneasy about a clerk signing for a consent document.

So the decision was this.

Outline tonight, print six thousand, and accept that if anything in the Marathi is wrong, it is wrong on a document a patient signs before surgery, and it is wrong permanently, because outlined text contains no characters for anybody to proofread. Six thousand forms is a reprint cost, an inspection finding, and a document in active clinical use that says something nobody intended.

Or hold. Plates move to Tuesday, delivery to Wednesday, and the inspection happens with the old forms on the desk. The officer has given an assurance he would then have to withdraw.

Both costs are real, and the second one is the kind that gets a freelancer replaced.

The thing worth noticing is that the shop's request and the actual risk are not the same thing. The shop is afraid of one failure: that they open a file, the font is missing, and the layout reflows. That fear is entirely reasonable and it has a second answer.

STOP HERE

Before turning the page: what would you have done, and what would it have cost if you were wrong? Write it down. The point of the next section is to compare.

WHAT HAPPENED

Nandini did not outline anything on Friday night.

She sent the shop a PDF with the fonts embedded rather than outlined, and she checked it with pdffonts before it left, so every row said yes under emb and every entry carried the six-letter subset prefix. That answered the shop's real worry: no font can go missing from a file that carries its own glyphs.

Then she rendered the two pages to PNG at 300 ppi and sent them to the matron's phone and to the ward clerk, with a short note asking them to read every line rather than to look at the layout. She also set a test string of hard cases beside the form, in the same font, and rendered that too.

The ward clerk came back in under ten minutes with one more error nobody had circled: a wrong vowel sign in the word for surgery, which traced back to the 2011 Kruti Dev conversion rather than to the font or the application. The two errors on the shop's proof turned out to be the same class of fault.

She corrected all three, re-rendered, and the matron confirmed by phone on Saturday morning after looking at the images. Plates were made on Saturday afternoon, printing ran on Sunday, and the forms reached the hospital on Monday evening, a day before the inspection.

She kept the editable Scribus file and the embedded-font PDF. Neither the shop nor the hospital has asked her for curves since.

Why does outlining make this particular failure permanent in a way it would not for an English form?

Outlining replaces characters with geometry, so the file stops containing text anyone can read, search, spellcheck or proofread. For an English form a wrong letter is still visible to anybody who looks. For Devanagari the failure is a shaping failure, and a reordered vowel sign or a broken conjunct is invisible to a non-reader, so removing the characters removes the only remaining way to catch it. The error becomes a drawing that looks deliberate.

The shop asked for curves. What answers the shop's real concern without freezing the text?

Embedding. A PDF export carries a subset of each font containing only the glyphs used, so the shop's machine does not need the font installed and nothing can reflow. Every PDF/X preset does this by default. The check is pdffonts on the exported file: every row should show yes under emb, and a six-letter prefix means a subset, which is normal. That covers the missing-font risk, which is the only risk the shop was actually raising.

An error traced back to a 2011 Kruti Dev document. Why was that predictable, and what does it imply for any Marathi or Hindi text a

client sends?

Kruti Dev and its relatives are not Devanagari encodings at all; they map Devanagari shapes onto Latin codepoints, so the stored text is Latin letters wearing a costume. Converting it to Unicode is a mapping exercise, and the mappings are imperfect, particularly around conjuncts and vowel signs. Any text with that history has to be read by somebody who reads the script before it is committed, and treating the conversion as reliable is the mistake.

MODULE 3 · TEST

Check what stuck

6 questions, one right answer each. This one is for you, not for a record: answers and the reason behind each are at the back. If two questions from the same module go wrong, re-read that module before the exam rather than after.

1 Why is outlining the text of a 64-page book a mistake, even though it does remove the missing-font risk?

A. Outlined text cannot be colour managed, so the black becomes rich black
B. PDF/X forbids outlined text, so the file will fail conformance
C. Each glyph becomes 30 to 90 nodes, and the RIP slows or times out
D. Outlining discards the kerning, so every line reflows on export

2 Your export dialog refuses to embed one particular font. Which part of the font file is it obeying?

A. cmap, which has no entry for the characters you used
B. OS/2, whose fsType bit forbids embedding
C. GSUB, which lacks the substitutions the text needs
D. hmtx, whose advance widths do not match the layout

3 Converting text to outlines is often described as a way round a font licence. Why is it not?

A. The outlines are still the foundry's drawings, and the licence covers use
B. Outlining leaves the font name in the PDF, so it remains detectable
C. Outlining counts as redistribution of the software, which licences forbid outright
D. Outlined text can be reverse-engineered back into an installable font

4 Why do A and V need less measured distance between them than H and I, to look equally spaced?

A. Diagonals carry more ink on the page, so the eye compensates for the extra weight
B. Kerning tables always set diagonals tighter, so the metrics do it
C. The white between two diagonals is a wedge, so the area is already large
D. V has a narrower advance width, which absorbs the difference

5 Capitals set on the outside of a tight circle splay apart at the top. What causes it?

A. The engine scales each glyph horizontally to fit the arc it occupies
B. The baseline and the cap line lie on circles of different lengths
C. Tracking is applied per glyph rather than per pair on a path
D. The path direction reverses at the top of the circle

6 Marathi text renders correctly on your screen and wrongly in the exported PDF. Where is the fault most likely to be?

A. The font lacks the conjunct glyphs, so it fell back to half-forms
B. The text was pasted as Unicode rather than in a legacy encoding
C. The exported subset dropped glyphs that were missing from the character map
D. The export path used a different shaping engine from the editing view

MODULE 4

Colour that survives the press

Why a screen and a sheet of paper disagree, and what to do about it. Device-independent colour and Delta E, ICC profiles and the difference between assigning and converting, gamut and rendering intents, halftone screening and dot gain, spot inks and the economics of a plate, designing in one and two colours, trapping, and how to proof without a budget.

> BY THE END OF THIS MODULE YOU CAN
>
> Specify a colour so the sheet that comes off the press matches what was agreed — choose between process, spot and duotone, read a gamut warning and pick a rendering intent, allow for dot gain and trapping, and state which of screen, proof and press you are trusting and why.

27 · 8 min

Two different physical processes

THE ONE THING TO KEEP

Screens add emitted light and paper subtracts reflected light, so white is the paper and black is limited by it — and because two different spectra can match under one illuminant and differ under another, print is judged under a standard D50 light rather than under whatever bulb is overhead.

Emitted light and reflected light

Your screen makes colour by **emitting**. Three phosphors or LEDs per pixel put out red, green and blue light and your eye adds them. All three at full is white. All three off is black — or rather, whatever black your screen manages, which on an LCD is dark grey.

Paper makes colour by **subtracting**. White paper reflects nearly all the light that falls on it. Cyan ink absorbs red and reflects the rest. Magenta absorbs green. Yellow absorbs blue. Lay cyan over magenta and you have absorbed red and green, leaving blue. Lay all three and you have absorbed almost everything, leaving a muddy dark brown — which is why black ink exists as a fourth.

Two different physical processes

A screen adds light

- Red, green and blue emitted per pixel
- All three at full strength is white
- Black is whatever the panel manages
- Ships at 6500K, which is bluish
- Two panels disagree before ink is involved

Paper subtracts light

- Cyan, magenta and yellow each absorb a band
- All three together is a muddy brown
- White is the paper. There is no white ink
- Judged under D50, at 5000K
- Black stops at about 5% reflectance

Nothing on the right is a degraded version of the left. They are opposite operations on light, which is why some screen colours have no printed equivalent and why the white in your logo is an absence of ink rather than an ink.

Two consequences follow immediately.

White is not a colour you can print. It is the paper. There is no white ink in a four-colour process job; the white in your logo is an absence of ink. On coloured stock, on a dark T-shirt, on transparent film, this becomes a real problem that needs a fifth station and a real white ink.

Black is limited by the paper. The darkest black on offset litho is around 5% reflectance. On a screen with a decent contrast ratio it is far lower. A photograph with deep shadows will always look flatter on paper than on the screen you approved it on, and that is physics, not a mistake.

Why two screens disagree

Even before ink enters the argument, two screens showing the same file disagree, for three separate reasons.

- **Different primaries.** A wide-gamut monitor's red is a different physical red from a laptop's. The same RGB triplet is a different colour on each.
- **Different white points.** Screens ship at 6500K by default, which is bluish. Print is judged under 5000K, which is warmer. The same neutral grey looks cool on one and warm under the other.
- **Different transfer curves.** Gamma, brightness, and whatever the manufacturer's "vivid" preset is doing.

Nothing in that list is fixable by being careful. It is fixable by profiling, which is two lessons away.

Metamerism, which catches everyone once

Two colours can match under one light source and differ under another. This is **metamerism**, and it happens because the eye reduces a whole spectrum to three numbers; two different spectra can produce the same three numbers under one illuminant and different ones under another.

The practical version: a brand colour printed as a spot ink and the same colour built from CMYK can look identical in the shop's fluorescent light and visibly different in daylight, because their spectra are different even though their appearance matched under one light.

This is why print is judged under a **standard illuminant**, normally **D50** (5000K), in a viewing booth or a light box with D50 tubes. If you are approving a proof under a shop's tube light, you are approving it under an unknown illuminant, and so is the client, and you may be approving different things.

A cheap and honest fix: buy a D50 bulb for one desk lamp. Around the price of a decent lunch, and it makes every colour conversation reproducible.

What to say to a client

Two sentences, before the job starts, save an argument later:

> *Screen colour and printed colour are made by different physical processes, so some colours on screen have no printed equivalent. I will pick from a printed chart and show you a proof on the actual stock.*

Then do it. Choosing from a printed swatch removes the screen from the decision entirely, which is the only fully reliable method available at any budget. It also changes what the client is approving: a sheet of paper in their hand rather than an image on a device neither of you has measured.

Ask for the chart on the stock the job will actually run on. The same CMYK build on gloss art card, matt art and uncoated bond gives three visibly different colours, because coated paper holds ink on the surface and uncoated absorbs it. A shop that runs all three will usually have all three charts.

The honest limitation

None of this is settled by precision. Two presses running the same file on the same paper on the same day produce slightly different sheets; ink density drifts through a run; paper varies by batch. Print tolerances are real and the trade works in **Delta E**, which is the next lesson. Anyone who promises an exact match without naming a tolerance is either being imprecise or has not run enough jobs.

BEFORE YOU MOVE ON

A brand colour is printed as a mixed spot ink on a carton and built from CMYK on a leaflet. Under the shop's lighting they match. In daylight the carton looks noticeably warmer. Nothing was printed wrong. What is the mechanism?

A. The carton stock is uncoated and the leaflet coated, so dot gain differs and shifts the CMYK build.
B. The two inks have different spectra that match under one illuminant but not under another.
C. Spot inks are formulated for D50 viewing and process inks for D65, so they are calibrated to different white points.
D. The leaflet was printed with a different rendering intent, which remapped the colour when it was converted to CMYK.

28 · 8 min

Lab, Delta E, and what "close enough" means

THE ONE THING TO KEEP

Lab describes colour by measurement rather than by device, Delta E measures the distance between two colours in it, and a specification that names a ΔE figure and a formula is something a printer can act on where "match it exactly" is not.

A number that means the same thing everywhere

`R 200 G 30 B 40` is an instruction to a device, not a description of a colour. It means "drive your red channel hard", and what comes out depends on the device.

To talk about colour independently of any device, you need a space defined by measurement rather than by hardware. **CIELAB** is the one everyone uses:

- **L*** — lightness, 0 (black) to 100 (white)
- **a*** — green to red
- **b*** — blue to yellow

Lab is defined from measurements of how human vision actually responds, and it is *perceptually roughly uniform*: a given distance in Lab corresponds to roughly the same perceived difference anywhere in the space. That uniformity is what makes the next idea possible.

Delta E

ΔE is the distance between two colours in Lab. It is the number the whole print industry argues in.

Rough interpretation, for the commonest formula (ΔE*ab, or CIE76):

- **Under 1** — not distinguishable by a human eye, even side by side.
- **1 to 2** — visible to a trained eye in a side-by-side comparison.
- **2 to 3.5** — visible to anyone in a side-by-side, usually invisible when seen separately.
- **3.5 to 5** — obviously different side by side.
- **Over 5** — a different colour.

Later formulas — ΔE94, ΔE2000 — correct known weaknesses of CIE76, particularly that it overstates differences in saturated blues. **ΔE2000 is the current standard** and gives smaller numbers than CIE76 for the same pair, so a supplier quoting ΔE must say which formula. Ask.

What tolerance to actually ask for

The international standard for offset printing, **ISO 12647-2**, sets tolerances for the solid primaries. In broad terms it permits a deviation of around ΔE 5 for the process colours on a production run, with tighter figures for the OK sheet against the proof.

For practical conversations:

- **A logo colour that matters** — ask for ΔE ≤ 2 against the supplied reference, and expect to pay attention to it on press.
- **General four-colour work** — ΔE ≤ 5 is normal and nobody will notice.
- **Across two different suppliers** — do not promise better than ΔE 3 without spot inks.

Saying "ΔE 2 against Pantone 2925 C measured under D50" is a specification. Saying "it must match exactly" is not, and the shop cannot act on it.

Measuring it, free and not free

- **A spectrophotometer** — an X-Rite i1 or similar, several hundred pounds new and much less used. It measures a printed patch and gives you Lab. Shops have one; ask them to measure your logo on the sheet.
- **Free calculation**, once you have two Lab values: any ΔE calculator, or a few lines of Python. `colormath` and `colour-science` both implement ΔE2000.
- **Free and rough**: photograph a printed sample and your reference side by side under the same light, open in GIMP, and use the colour picker. This tells you *whether* there is a difference, not *how much*, because your camera and your screen are both unprofiled. It is still better than arguing from memory.

Where you will actually use this

You will not compute ΔE often. You will use the vocabulary constantly:

- To ask a supplier what tolerance they work to, and to hear a real answer.
- To decide whether a reprint is justified, rather than arguing about whether the blue "looks right".
- To specify a brand colour in a usage sheet as Lab as well as CMYK, RGB and hex — because Lab is the only one of the four that means the same thing in two different factories.

That last is the concrete takeaway. Put a Lab value in your brand sheet. Almost nobody does, and it is the value a printer in another city can actually hit.

One thing to be careful about

A ΔE figure quoted without a measurement condition is incomplete. Two labs measuring the same printed patch can report different Lab values depending on the instrument's illuminant, its filter, the backing behind the sheet, and whether the paper contains optical brightening agents that fluoresce under UV. The industry standard for this is **M1**, which specifies a defined UV content in the measuring light, and it exists precisely because brightened papers gave inconsistent readings under the older M0 condition.

You do not need to own an instrument to use this. You need to know that "ΔE 1.8" from one supplier and "ΔE 1.8" from another are only comparable if both were measured the same way — which is a fair question to ask, and a supplier who can answer it is telling you something useful about how they work.

BEFORE YOU MOVE ON

A brand sheet specifies a logo colour as CMYK 100/60/0/5 and hex #0055A4. A second printer in another city produces a noticeably different blue and insists they printed the file correctly. Why does adding a Lab value to the sheet help where the other two do not?

A. Lab has a wider gamut than CMYK or sRGB, so it can describe saturated blues that the other notations clip.
B. Lab values are stored at higher precision, so rounding errors between the two printers' workflows are eliminated.
C. CMYK and hex are instructions to unspecified devices; Lab is a measured target a printer can verify.
D. Lab is the native space of every RIP, so specifying it removes one colour conversion from the workflow.

Profiles, and the dialog that decides everything

THE ONE THING TO KEEP

Assign changes what the existing numbers mean; convert changes the numbers to preserve the appearance — and a CMYK conversion is only as correct as the press profile you converted to, which is a question for your printer rather than a default.

What a profile is

An **ICC profile** is a lookup table plus some maths that translates between a device's own numbers and a device-independent space — Lab or CIEXYZ, called the **Profile Connection Space**.

An RGB profile says: *on this device, R 200 G 30 B 40 produces this Lab value.* A CMYK profile says: *on this press, on this paper, C 0 M 90 Y 85 K 0 produces this Lab value.*

Every colour conversion is therefore two steps: source device numbers → PCS → destination device numbers. Both profiles must be right for the answer to be right.

Assign versus convert, which is the whole lesson

Two menu items, adjacent in most applications, doing opposite things.

Assign profile changes the *interpretation* and leaves the numbers alone. The file still says R 200 G 30 B 40; you have changed what that means. The colour on screen changes. The pixels do not.

Convert to profile changes the *numbers* to preserve the appearance. The file becomes R 193 G 52 B 47, or whatever the destination needs to produce the same Lab value.

Get this the wrong way round and you either destroy colours or misdescribe them:

Two menu items, opposite jobs

	The numbers	The appearance
Assign	**Untouched** the file is not edited	**Changes** now read differently
Convert	**Rewritten** new values written	**Preserved** same measured colour

Assign when you know what the file is and its tag is wrong or missing. Convert when you are moving the file to a different device. Assigning sRGB to an Adobe RGB photograph flattens it; converting the same file twice loses whatever fell outside the gamut the first time.

- **Assigning** sRGB to a file that is actually Adobe RGB makes everything look flat and desaturated — because the saturated numbers are now being read as if they came from a narrower device.
- **Converting** a file twice — to CMYK and back to RGB and to CMYK again — loses a little each time, because each conversion clips whatever fell outside the destination gamut and cannot recover it.

The rule: **assign when you know what the file is and the tag is wrong or missing. Convert when you want to move the file to a different device.**

Which profiles to use

For source RGB:

- **sRGB IEC61966-2.1** — phones, web, almost everything. If a file has no tag, sRGB is the correct guess.
- **Adobe RGB (1998)** — wider in greens and cyans. Worth using only if your camera, screen and workflow all support it end to end.
- **Display P3** — Apple devices and modern phones. Increasingly what a client's photograph actually is.

For destination CMYK — and this matters more than most people realise, because a "CMYK" file converted with the wrong press profile is simply wrong:

- **FOGRA39 / ISO Coated v2** — European coated sheetfed, total ink around 330%. The most widely used default.
- **FOGRA51 / PSO Coated v3** — the updated version for modern papers with optical brighteners.
- **FOGRA47 / PSO Uncoated** and **FOGRA52** — uncoated stock, lower ink limits, much larger dot gain.
- **GRACoL 2013 / CRPC6** — the North American equivalent.
- **Japan Color 2001 Coated** — widely used in Asia, and what many Indian shops' RIPs are set to by default.

Ask your printer which one they are set to. It takes one line in an email and it is the difference between a proof that matches and one that does not. If they cannot tell you, ISO Coated v2 is a defensible default for coated work.

Free downloads: the **European Color Initiative** publishes the ECI profiles at no cost, and **Adobe** ships several in its applications. Ghostscript and Krita ship a set too.

Setting them up

- **Scribus** — File ▸ Preferences ▸ Colour Management. Set the RGB working space, the printer profile, tick *Simulate printer on screen* for soft proofing. Scribus's colour handling is the strongest of the free tools and the reason it is the free press chain's export stage.
- **GIMP** — Edit ▸ Preferences ▸ Colour Management, and Image ▸ Colour Management ▸ Assign / Convert as separate menu items, correctly named.
- **Krita** — profiles set per document at creation; the soft-proof toggle is **Ctrl+Y**.
- **Inkscape** — works internally in RGB; colour management is in Document Properties ▸ Colour and is thinner than the others. For press colour, hand off to Scribus.
- **Illustrator / InDesign** — Edit ▸ Colour Settings, and Edit ▸ Assign Profile / Convert to Profile as separate items.

The commonest real failure

A photograph arrives from a client with no embedded profile. The application assumes sRGB. The photograph was actually shot in Adobe RGB. Nobody notices until the printed skin tones are slightly grey.

The fix is to ask, and the check is to look: an untagged file whose saturated colours look oddly flat is usually a wide-gamut file being read as sRGB. Assign Adobe RGB and see whether it comes alive. If it does, that was it.

BEFORE YOU MOVE ON

A client's photograph arrives with no embedded ICC profile. Opened in the designer's application it looks flat and desaturated, particularly in the greens. The photographer confirms it was shot and exported in Adobe RGB. What is the correct action?

A. Convert the file to Adobe RGB, which will restore the saturation the numbers originally carried.
B. Assign Adobe RGB, leaving the numbers alone so they are read against the correct device description.
C. Increase saturation in an adjustment layer until the greens look correct, then convert to the press CMYK profile.
D. Re-export the file from the original camera raw with sRGB selected, since the application cannot read Adobe RGB reliably.

30 · 8 min

What happens to a colour the press cannot print

THE ONE THING TO KEEP

Perceptual compresses the whole gamut smoothly and suits photographs; relative colorimetric with black point compensation keeps in-gamut colours exact and clips the rest, which is what artwork and brand colours need.

Gamut is a solid, not a triangle

The familiar horseshoe diagram with triangles drawn on it is a slice. A real gamut is a three-dimensional solid in Lab, and its shape differs by lightness: a press can produce a vivid mid-tone orange it cannot produce as a light tint or a dark shade.

Where CMYK is smaller than sRGB, reliably:

- **Saturated blues and violets.** RGB blue (0,0,255) has no CMYK equivalent at all. It comes back as a dull purple-navy, and this is the single most common colour disappointment in print.
- **Saturated greens**, particularly bright emerald and spring green.
- **Vivid orange and warm red** at high saturation.
- **Anything fluorescent.** No process ink is fluorescent.

Where CMYK is *larger* than sRGB, which surprises people: some deep cyans and some rich yellows on coated stock exceed what an sRGB screen can show.

The four rendering intents

When a colour will not fit, the conversion must decide what to do with it. That decision is the **rendering intent**, chosen in the convert dialog.

- **Perceptual.** Compresses the whole gamut smoothly so relationships between colours are preserved. Out-of-gamut colours come in; in-gamut colours also shift slightly to make room. Correct for photographs, where a smooth gradation matters more than any one exact value.
- **Relative colorimetric.** Leaves in-gamut colours exactly where they are and clips out-of-gamut ones to the nearest reproducible colour. White is mapped to paper white. Correct for artwork, logos and anything with specified brand colours — which is most vector work.
- **Absolute colorimetric.** Like relative, but does *not* remap white — it simulates the source's paper white, so a proof on bright white stock shows the yellowish cast of the newsprint you are simulating. Correct only for proofing one press on another.
- **Saturation.** Maximises vividness, ignores accuracy. Intended for business charts. Not for anything that matters.

Relative colorimetric with black point compensation is the right default for vector artwork. **Perceptual** is the right default for photographs. Most people use whatever is preset and never learn there was a choice.

Black point compensation

Without it, a source black that is darker than the destination's darkest black gets clipped, and every shadow detail near it collapses into one flat tone. Black point compensation scales the whole tonal range so the source black maps onto the destination black, preserving separation in the shadows at the cost of a small overall lightening.

Leave it **on** for relative colorimetric. It is on by default in Adobe applications and in Scribus, and it is one of the few defaults that is genuinely correct.

Seeing the problem before you print

Soft proofing shows you the destination's gamut on your screen. Free everywhere:

- **Scribus**: Preferences ▸ Colour Management ▸ *Simulate printer on screen*, plus the *Out of gamut* warning checkbox.

- **GIMP**: View ▸ Colour Management ▸ *Soft-proofing*, set the profile, and enable *Out of gamut warning* with a chosen marker colour.
- **Krita**: **Ctrl+Y**.
- **Photopea**: View ▸ Proof colours.
- **Illustrator / Photoshop**: View ▸ Proof Setup ▸ Custom, then **Ctrl+Y**; Photoshop adds View ▸ Gamut Warning.

A soft proof on an uncalibrated screen is a rough guide, not a match. It will still tell you truthfully *which* colours are unreachable, because that is a property of the profiles rather than of your monitor.

What to do when a colour will not fit

In order of preference:

1. **Choose a different colour**, from a printed CMYK chart, before the design is signed off. Cheapest fix by a wide margin.
2. **Use a spot ink** for that one colour. Costs a plate; reaches colours process cannot.
3. **Accept the shift** and tell the client what it will look like, showing a printed sample rather than a screen.
4. **Change the substrate.** A brighter, whiter, coated stock has a visibly larger gamut than uncoated. Sometimes the paper is the answer.

What not to do: push the CMYK values higher in the hope of more saturation. Beyond the press's ink limit you get a wet, unstable sheet and no extra colour, because the ink is already absorbing everything it can absorb.

The one that catches designers out

RGB blue is the classic, and it is worth seeing rather than reading about. Fill a rectangle with `#0000FF`, soft-proof it against a coated press profile, and look at what comes back: a flat, slightly purple navy, several ΔE away from anything anyone would call blue. Now do the same with `#00A0E9` — a cyan-leaning blue — and it survives almost untouched, because it sits near the cyan primary the press actually has.

That is the practical lesson of gamut in one experiment. Brand colours chosen near a process primary reproduce well everywhere. Brand colours chosen in the corners of the RGB cube reproduce nowhere, and will need a spot ink for the rest of the company's life.

BEFORE YOU MOVE ON

A poster combines a photograph with a flat brand orange specified in CMYK. Converted with perceptual intent, the brand orange comes out slightly different from the same orange on the client's existing stationery. What happened, and what should have been done?

A. Perceptual intent compressed the whole gamut, shifting in-gamut colours too; relative colorimetric would have left the orange untouched.
B. The photograph's profile overrode the artwork's, so both elements were converted using the image's embedded space.
C. Perceptual intent clips out-of-gamut colours, and the brand orange was outside the press gamut, so it was moved to the nearest printable value.
D. Black point compensation was enabled, which rescales the tonal range and lightens saturated colours along with the shadows.

31 · 9 min

Why your 20% tint printed at 30%

THE ONE THING TO KEEP

Tone is made of dots, and a dot grows when it hits paper — around 15% on coated and 25% on uncoated at the midtone — which the press profile already corrects for, so the failures come from converting with the wrong profile or from relying on very light tints and small reversed type.

Ink is one density; tone is an illusion

A press lays down ink at full strength or not at all. There is no such thing as 30% cyan ink. A 30% tint is a field of tiny cyan dots covering 30% of the area, and your eye averages them.

That conversion — continuous tone into a pattern of dots — is **screening**, and the RIP does it.

AM screening, the traditional kind

Amplitude modulated screening puts dots on a regular grid at a fixed spacing and varies their *size*. The spacing is the **line screen**, measured in lines per inch:

- Newsprint: **85 lpi**
- Uncoated stock, general commercial: **133 lpi**
- Coated stock, quality commercial: **150–175 lpi**
- Fine art and packaging: **200 lpi and above**
- Screen printing on textile: **35–65 lpi**, because the mesh cannot hold finer
- Large format outdoor: as low as **15–25 lpi**, seen from distance

Each of the four inks is screened at a **different angle** so the dots do not sit on top of each other. The conventional set is cyan 15°, magenta 75°, yellow 0°, black 45° — black at 45° because a diagonal pattern is least visible to the eye, and yellow at 0° because it is the palest and its pattern shows least.

Get the angles wrong and you get **moiré**: a coarse interference pattern from two fine patterns overlapping. Get them right and you get a **rosette**, the tiny flower shape you can see with a loupe on any printed photograph.

Moiré also appears when your *artwork* contains a fine regular pattern — a scanned halftone, a fabric weave, a pinstripe — that interferes with the press screen. The fix is to blur the artwork very slightly or change its scale, and it must be done before the RIP sees it.

FM screening

Frequency modulated or stochastic screening uses dots of one small size scattered pseudo-randomly, varying their *number*. No grid, so no angles and no moiré at all, and much finer detail. It demands better press control and it dries differently. Increasingly common on good digital and offset work; ask whether your shop offers it if the job has a scanned pattern in it.

Dot gain, and the number you need

A dot printed on paper comes out bigger than the dot on the plate. Two separate mechanisms:

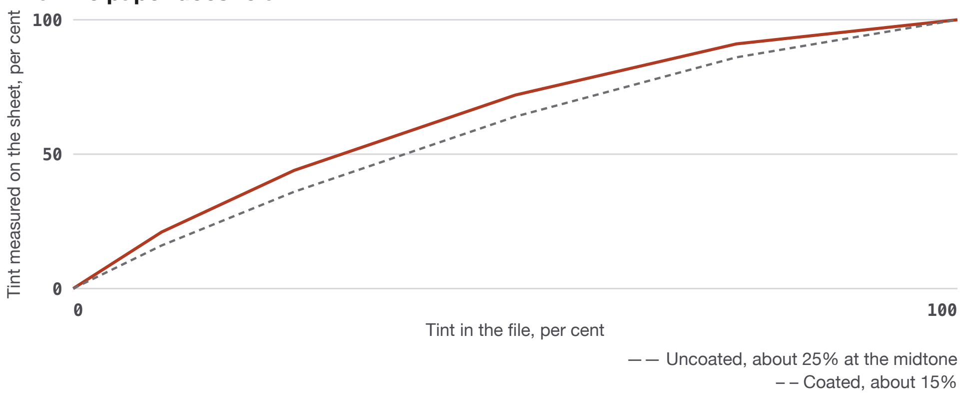
- **Mechanical gain.** Ink physically spreads as it is pressed into the paper. Worse on absorbent, uncoated stock.
- **Optical gain.** Light enters the paper beside a dot, scatters, and emerges under it. The dot *appears* larger than it is even when it is not.

Together they are substantial. Typical midtone (50%) gain:

- Coated sheetfed offset: **13–17%** — so a 50% tint reads as about 64%
- Uncoated: **20–25%** — a 50% tint reads as about 72%
- Newsprint: **25–30%**
- Screen print on textile: **can exceed 30%**

The press profile already accounts for this. Convert with FOGRA51 for coated or FOGRA52 for uncoated and the numbers going to the plate are adjusted so the printed result is right. Dot gain becomes a problem when you *do not* use a profile, or when you use the wrong one — an uncoated job converted with a coated profile prints visibly dark and muddy.

What the paper does to a tint



The press profile already corrects for this, so the trouble comes from converting with the wrong one.
An uncoated job converted with a coated profile prints dark and muddy across every midtone in it.

What to do about it in your artwork

Three practical consequences you control directly.

Light tints close up. A 5% tint on uncoated can print at 12% or vanish entirely depending on the press's minimum dot. If a design depends on a very pale wash, print a test or use a heavier tint of a lighter colour.

Small reversed type fills in. White text on a dark ground loses width as ink spreads into the counters. Below about 8 pt in a light weight on uncoated stock, an `e` closes up. Use a heavier weight, a larger size, or accept it.

Fine screens in artwork clash. Avoid pinstripes below about 0.3 mm pitch in anything going to litho.

The check that costs nothing

Ask the printer for a printed **tint chart** on the stock you are using — a grid of 5% steps in each ink and in common builds. Many shops have one and will hand it over. It is the single most useful object in the trade: it tells you exactly what 20% cyan looks like on that paper, on that press, with no screen involved in the decision.

BEFORE YOU MOVE ON

A brochure designed with a delicate 5% grey background wash is proofed on a coated digital printer and looks right. The production run is on uncoated stock and the wash prints as a distinctly visible grey. Nothing was changed in the file. Why?

A. The uncoated run used FM screening, which distributes dots randomly and increases apparent coverage at low percentages.

B. Uncoated stock is less bright, so the same ink coverage reads darker against a duller paper white.

C. The digital proof rendered the 5% as a continuous tone while the offset press screened it into dots, which always look darker.

D. Dot gain on uncoated stock runs around 25% at the midtone, so a 5% tint prints as considerably more.

32 · 9 min

A spot colour is a plate, and a plate is money

THE ONE THING TO KEEP

A spot colour is a physically mixed ink on its own plate, which buys consistency, out-of-gamut reach and economy on low colour counts at the cost of a plate — and a spot must be defined as a spot swatch and then verified in the separations, because a spot that exports as process is a silent failure.

What a spot ink is

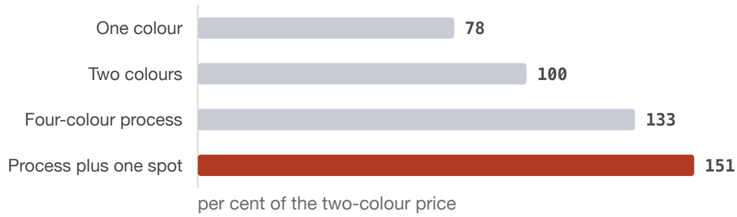
A **spot colour** is an ink mixed to a recipe before it goes on the press, and printed from its own unit. Not simulated from four process inks — a physically different ink.

That buys three things process cannot:

- **Consistency.** The same mixed ink on ten different jobs at ten different suppliers gives the same colour. A CMYK build depends on the press, the paper and the day.
- **Reach.** Metallics, fluorescents, and saturated colours outside the CMYK gamut. There is no CMYK build for silver.
- **Economy on short colour counts.** A two-colour job — black and one spot — needs two plates and two units. Printing the same job in process needs four of each, and costs more.

And it costs: **each spot is an additional plate, an additional unit and an additional wash-up.** On a job already running four-colour process, adding a fifth spot is a real line on the quotation. Ask before you specify it.

What a plate adds on a 1,000-piece run



A short run is dominated by setup rather than ink: each plate carries plate cost, make-ready and a wash-up. On a 100,000-piece run the same four bars are almost level, which is why the honest advice is short run, few colours; long run, colours are cheap.

The Pantone situation, stated plainly

Pantone's colour libraries were removed from Adobe's applications in **2022**. Using the current libraries inside Illustrator, InDesign or Photoshop requires a separate paid Pantone Connect subscription. The physical guide books were already expensive and are recommended to be replaced periodically because the printed inks fade.

What this means practically:

- **You do not need Pantone to work.** Most small and mid-size shops work from their own printed CMYK charts and their own mixed-ink books.
- **You do need a shared reference.** Whatever you and the printer both hold — their chart, a physical swatch, a printed sample of the previous job — is the reference.
- **A Pantone number is still the lingua franca**, and an ink mixer can mix to one from the formula regardless of what software you used. If a client's brand sheet says PANTONE 2925 C, that is actionable; you do not need the library installed to write it in a note to the printer.

Free alternatives for getting a defined spot into a file: define your own named spot swatch (below), and specify the reference in words on the docket.

Defining a spot so it stays a spot

The mechanic matters, because a "spot" that exports as process is worse than useless.

- **Scribus**: Edit ▸ Colours & Fills ▸ New, name it, and tick **Spot Colour**. Scribus will keep it as a separate separation through PDF export. This is the strongest free path.
- **Illustrator**: Swatches panel ▸ New Swatch ▸ Colour Type: **Spot Colour**. The swatch gets a small dot in its corner.
- **CorelDRAW**: choose a colour from a spot palette, or Palette Editor to create one; it exports as a named separation.
- **Inkscape**: no true spot colour support. You can name a colour, but it exports as process. For spot work, lay out in Scribus.

Then **verify** in the exported PDF. Acrobat Pro's Output Preview ▸ Separations lists every plate. Free equivalent:

```
gs -q -o /dev/null -sDEVICE=inkcov job.pdf     # per-page CMYK
coverage
gs -o sep-%d.tif -sDEVICE=tiffsep job.pdf      # one file per
separation, spots included
```

`tiffsep` writes a file per plate and names the spot ones. If you see a file you did not expect, you have an accidental extra plate — and that is a cost the client will find on the invoice.

Tints of a spot

You can screen a spot ink like any other: 100%, 60%, 30% of the same ink gives you three tones from one plate at no extra cost. This is how good two-colour work gets its range, and it is the subject of the next lesson.

Tints of a spot are subject to the same dot gain as anything else, and a spot at 20% on uncoated behaves exactly like 20% cyan.

When to convert a spot to process

Convert when: the job is already full process, the colour is inside the CMYK gamut, and the client has not specified a tolerance you cannot meet otherwise.

Keep the spot when: the colour is out of gamut, it is metallic or fluorescent, the job is one or two colours anyway, or the brand genuinely requires cross-supplier consistency.

Say which you have done, in writing, on the docket. "Logo blue supplied as a spot, please convert to process" is a clear instruction. Leaving an unexplained fifth plate in the file and hoping is not.

BEFORE YOU MOVE ON

A designer defines the brand colour in Inkscape, names the swatch "Brand Blue Spot", and exports a PDF for a two-colour job. The printer quotes for four plates. What has happened?

A. The printer's RIP does not read swatch names, so every named colour is quoted as a separate plate regardless of its type.

B. Inkscape has no spot support, so the named swatch exported as a CMYK build across four separations.

C. The PDF was exported as PDF/X-4, which converts all spot colours to process to keep transparency live.

D. The swatch was defined in RGB rather than CMYK, and RGB colours are always separated into four plates by the RIP.

33 · 8 min

Getting a whole design out of two inks

THE ONE THING TO KEEP

One plate gives five usable values including the reversed paper, and two inks give tints of each plus overprints — which is enough range for real work, provided the colour count is decided before the design is drawn rather than after.

The constraint that improves the work

A great deal of real print is one or two colours: letterheads, forms, invoice books, rubber stamps, screen-printed bags and shirts, newspaper advertising, cheap flyers, packaging for small producers. It is cheaper, it turns round faster, and the shops that do it are everywhere.

It is also where design gets interesting, because the constraint forces decisions.

The range available from one ink

One plate does not mean one tone. It means one hue at any percentage.

- **100%** — the solid.
- **60–70%** — reads as a distinctly lighter version, still confident.
- **30–40%** — a background tone, still clearly the colour.
- **10–15%** — a tint panel. Below this, dot gain and the press's minimum dot make it unreliable.
- **Reversed** — the paper showing through a solid. This is your second "colour", free.

Five values from one plate. Most one-colour work uses two of them and looks impoverished for no reason.

Two inks make more than two colours

With two plates you get: each ink's tint range, plus **overprinting** — one ink printed on top of the other, which mixes.

A warm red and a deep blue overprinted give a dark plum. Tinted against each other — 100% blue over 40% red, 40% blue over 100% red — give two more. A two-colour job can honestly show eight or nine distinguishable colours.

Two cautions, both real:

- **The overprint colour is not predictable from the screen.** Transparent ink over transparent ink is not the same as a screen's multiply blend. Ask the shop to run a small overprint chart, or use one they already have.
- **Ink sequence matters.** Blue over red and red over blue are not the same colour, because the first ink down is partly covered rather than fully transparent. The press decides sequence; ask.

Duotone for a photograph

A photograph in one ink is a monotone: grey levels mapped onto one colour. Flat, and it loses shadow depth because one ink cannot reach a deep black.

A **duotone** maps the image onto two inks with different tonal curves — typically black carrying the shadows and a colour carrying the midtones. The result has visibly more depth than either alone, and it is a genuine technique rather than a fallback.

Free path, in GIMP: convert to greyscale, then Colours ▸ Map ▸ Gradient Map with a two-colour gradient, or use Colours ▸ Curves on separate channels after converting to RGB. Photopea has a Duotone mode under Image ▸ Mode. Krita has per-channel curves. None of these is as controllable as Photoshop's Duotone dialog with its per-ink curves, and all of them produce a usable result.

For a screen-printed job, remember that the two "inks" here become two screens, and the halftone angles must differ or you get moiré.

Designing for it from the start

Decide the colour count **before** you draw. Retrofitting a four-colour design into two inks is a redesign, not a conversion.

Practical rules:

- Build everything from flat fills and tints of the two inks, named as swatches, from the first minute.
- Set up your document with exactly two spot swatches and delete the rest. If you cannot pick a colour, you cannot use it.
- Work in greyscale first for the tonal structure, then assign the two hues. The structure is what makes it work.
- Keep a printed tint chart of your two inks beside you.

The economics, so you can advise a client

A shop's quotation for a short run is dominated by setup, not by ink. Each additional plate adds plate cost, make-ready time, and a wash-up between jobs. On a 1,000-piece run, going from two colours to four can add a third to the price; on a 100,000-piece run the difference is small because setup is amortised.

So the honest advice to a small client is: **short run, few colours; long run, colours are cheap.** Being able to say that, with the reason, is worth more to them than any amount of styling.

Digital presses change the arithmetic, and it is worth knowing which kind of machine a job is going on. A toner or inkjet digital press has no plates at all, so four colours cost the same as one and a run of fifty is viable. What it cannot do is spot inks, metallics, or the particular density of a solid litho colour. So the rule inverts: on digital, use all four colours freely and forget about plate counts; on litho, count the plates before you draw.

Ask which press before you design. "Is this digital or offset?" is one line, and it changes the design brief rather than just the quotation.

BEFORE YOU MOVE ON

A two-colour job is designed with a red and a blue spot, and the designer specifies a plum passage as red overprinting blue. The printed plum is lighter and pinker than the on-screen multiply preview. Why is this expected rather than an error?

A. Overprint preview in design software always previews the result at 100% opacity, regardless of the tint values that were actually set.
B. The inks were printed in the wrong sequence, so the blue went down first when the design assumed red first.
C. Ink over ink is only partly transparent and depends on sequence; a screen's multiply assumes full transparency.
D. Spot inks cannot overprint at all, so the press knocked out the blue beneath the red and only the red printed.

34 · 8 min

Trapping, and the white line around everything

THE ONE THING TO KEEP

Registration drifts by a tenth of a millimetre or so, exposing white slivers where two knocked-out colours meet — so the lighter colour is spread into the darker by around 0.1 mm, usually by the printer's RIP rather than by you, while 100% black text overprints and white must never overprint.

The problem: paper moves

A four-colour sheet passes four printing units. Between them the paper stretches slightly, absorbs a little moisture, and is gripped and released. The four impressions therefore do not land exactly on top of each other. A drift of **0.1 to 0.2 mm** is normal on offset litho; flexo on film is worse; screen printing on stretchy fabric is worse still.

Where two different colours meet edge to edge, that drift shows as a sliver of unprinted paper between them. A thin white line around every letter and every shape.

Knockout, which causes it

By default, an object **knocks out** what is beneath: the press does not lay the background ink where the object sits, so the inks do not mix and the colour is what you specified. Knockout means the two inks meet along an exact shared edge — and an exact shared edge is precisely what misregistration exposes.

Trapping, which fixes it

Trapping deliberately overlaps the two colours by a small amount, so that when registration drifts, the drift reveals an overlap rather than a gap. An overlap is a slightly darker line, which the eye forgives. A white gap, it does not.

Two directions, and the choice is not arbitrary:

- **Spread** — the *lighter* colour is expanded outwards into the darker one.
- **Choke** — the darker background is contracted inwards, which amounts to the same thing from the other side.

The lighter colour always moves. The reason is that the eye reads the edge of a shape from its darker side; enlarging the light colour slightly leaves the perceived boundary where it was, while enlarging the dark one visibly fattens the shape.

Typical trap width: **0.08 to 0.15 mm** on sheetfed offset, more on web, considerably more on screen print and flexo. Ask.

Who does it, and why it is usually not you

On almost all commercial offset work, **the printer's prepress or the RIP applies trapping automatically**, using in-RIP trapping settings tuned to their press. If you also trap by hand, you get double traps and visibly fat edges.

So the default answer is: **do not trap, and say so.** Put "no manual trapping applied" on the docket and let them do their job.

You trap by hand when:

- **Screen printing**, especially on garments, where registration is coarse and the shop expects you to supply trapped separations.
- **Flexo packaging** on film, where the trap values are large and job-specific.
- **A two-colour job at a small shop** with no in-RIP trapping.
- **Vinyl and signage** where two cut colours butt together.

Doing it by hand

The manual method is an offset plus overprint:

1. Duplicate the lighter object.
2. Offset the duplicate outwards by the trap width (Object ▸ Path ▸ Offset Path, or a stroke of twice the trap width set to the lighter colour).
3. Set that duplicate to **overprint fill** and **overprint stroke**, so it prints on top of the darker colour rather than knocking it out.
4. Place it beneath the original.

Illustrator has Window ▸ Separations Preview plus Effect ▸ Pathfinder ▸ Trap for simple cases. Scribus has per-object overprint control in the properties palette. CorelDRAW has *Overprint fill* and *Overprint outline* in Object Properties and an in-RIP trapping section in its PDF export dialog.

The two cases you must know without being told

Black text overprints. Small black type set to knock out will show coloured fringes at the slightest drift. Set 100% K text to overprint and no trap is needed at all, because the black simply prints over whatever is there. Most workflows do this automatically for 100% K; check rather than assume.

White never overprints. White is the absence of ink. An object at 0/0/0/0 set to overprint prints nothing, and whatever is behind it shows through. Your white logo vanishes from the middle of the photograph, and the file looked perfect on screen the whole time. Turn on overprint preview before every export and look.

Where trapping is somebody's whole job

On a garment screen-printing floor, trapping is the difference between a saleable shirt and a reject, because the fabric moves under the squeegee and registration on a manual carousel is measured in whole millimetres rather than tenths. Separations there routinely carry traps of 0.5 mm or more, plus a choked underbase, and the shop will tell you the values they want if you ask for them by name.

That is worth remembering as a general principle. Trap width is a property of the process, not of the artwork — so the only correct way to choose one is to ask the person running the machine. A number you found on the internet belongs to somebody else's press.

BEFORE YOU MOVE ON

A packaging designer spreads the darker of two adjacent colours outwards by 0.12 mm to create a trap. Registration is now forgiving, but the artwork looks subtly wrong compared with the approved proof. What is the error in the method?

A. The trap width should be measured in points rather than millimetres, so 0.12 mm is far too narrow to have any effect.
B. The trap was applied as a spread when a choke was needed, and the two are computed from opposite reference edges.
C. Expanding the dark colour moves the perceived edge, because the eye reads a boundary from its darker side.
D. The duplicate was not set to overprint, so it knocked out the lighter colour and widened the gap instead of closing it.

35 · 8 min

Proofing when you cannot afford a proofing system

THE ONE THING TO KEEP

A soft proof shows which colours are unreachable, a calibrated digital proof predicts the sheet, and the first good sheet off the press is the only thing that shows the real ink on the real paper — so approve under a known light, against a physical reference, and keep the signed artefact.

Four things called a "proof", and they are not equal

- **A soft proof** — your screen, simulating the destination profile. Free. Tells you which colours are unreachable; tells you almost nothing about how the sheet will look unless your screen is profiled.
- **A digital proof** — an inkjet proofer calibrated to simulate a specific press profile, printed on proofing stock. This is what "contract proof" usually means. It carries a colour bar and often a signature line. Costs money; catches nearly everything.
- **A press proof or wet proof** — a short run on the actual press, actual ink, actual paper. Expensive, usually reserved for packaging and long runs. The only thing that shows you the real substrate and the real ink film.
- **A machine proof or "first-off"** — the first good sheet of the actual run, approved at the press. Free if you can be there; the strongest thing available to a small job.

If your budget allows one, take the machine proof and go to the press.

Calibrating a screen without buying anything

A profiled screen is the cheapest large improvement available, and there are three tiers.

1. **Borrow a colorimeter.** An X-Rite i1Display or Datacolor Spyder takes ten minutes. Many print shops, colleges and photography clubs have one. **DisplayCAL** with ArgyllCMS is free, open source, and better than most bundled software.
2. **Rent or share one.** A group of freelancers buying one between six is entirely normal.
3. **If you genuinely cannot**: set your monitor to its sRGB preset, set brightness to about 120 cd/m² (roughly 40–50% on most panels, which will feel too dark at first), turn off any dynamic contrast or "vivid" mode, and work in a room with consistent light. This is not calibration and it removes the worst of the variation.

Do not use a phone or a laptop with an automatic brightness that changes through the day for any colour decision.

The phone-photo trap

A client photographs the printed sheet and sends it saying the colour is wrong. That photograph has been through: the room's illuminant, the camera's automatic white balance, its tone curve, HDR processing, JPEG compression, and their screen. It is evidence of almost nothing.

The reply that resolves this, politely:

> *Photographs of print are not reliable for colour. Can you hold the sheet next to the printed reference we agreed and tell me whether they differ to your eye, under daylight or a 5000K lamp?*

That reframes the conversation onto the only comparison that means anything: two physical objects under one light.

Getting a free printed reference

Shops will generally give you, if you ask:

- **Their CMYK tint chart** on the stocks they run. The single most useful thing in this lesson.

- **A paper dummy** — the actual stock, folded and trimmed to size, unprinted. Free, and it settles arguments about weight and feel before anything is printed.
- **Samples of previous jobs** with colours near yours.
- **A "gang run" slot** — some shops let a small job ride along on a larger sheet at low cost, which gets you a real printed sample for a fraction of a normal run.

Approving a proof properly

Whatever the proof is, approve it under a known light, mark it, and keep it.

1. View under **D50**, or failing that north daylight, never under a warm tube.
2. Check against the *reference*, not against your memory of the screen.
3. Mark corrections on the physical proof, in a colour that is not in the job.
4. Sign and date it, photograph it, and keep both the photograph and the exact PDF you supplied.

That last step is not bureaucracy. When a run comes back wrong, the question is always whether the file was wrong or the press was, and the only way to answer it is the artefact you both agreed on. Handling that conversation is a later lesson; having the evidence is this one.

What you can honestly promise

You can promise: the file is built correctly, the colours are specified against a named reference, and a supplied proof matched. You cannot promise an exact colour across two suppliers without spot inks, or an exact match between a screen and a sheet. Saying this at the start costs nothing and removes the single most common source of dispute in print work.

BEFORE YOU MOVE ON

A client sends a phone photograph of a printed leaflet claiming the green is wrong. What is the most useful next step, and why?

A. Compare the photograph's pixel values against the specified CMYK build to quantify the difference.
B. Re-export the PDF with a different rendering intent and send it to the printer for a corrected run.
C. Ask them to hold the sheet beside the agreed printed reference under daylight and say whether the two differ.
D. Ask the client to recalibrate their phone's display before photographing it again in better lighting conditions.

CASE STUDY · MODULE 4

The grape carton, the fifth plate and two printers in two cities

A table-grape growers' co-operative near Sangli, printing export cartons in Pune and domestic cartons locally.

The co-operative packs table grapes for two markets. Domestic cartons are printed at a five-unit press in Sangli, forty thousand a season. Export cartons need a different board and a longer run, and go to a larger shop in Pune, about ninety thousand.

Their brand green was chosen in 2016 on somebody's laptop. It is a vivid emerald, a little towards blue, and it is on every carton, every crate label and the co-operative's board outside the collection centre.

The problem surfaced when the Pune shop's prepress operator soft-proofed the artwork against the press profile and rang to say the green would not hold. The operator did the useful thing and sent a printed sample rather than a screenshot: the nearest CMYK build on the export board, beside the current Sangli carton. Under the office tube light they looked similar. Under the D50 lamp the Pune operator had put on the desk, they did not. The export sample was flatter and about four Delta E away, which is a difference anybody sees when the two are side by side.

That is the worse case for a fruit exporter, because retail buyers do see them side by side. The co-operative's cartons stand on a pallet next to each other at the mandi and, in season, on the same shelf in two different cities.

The managing committee had three options and money on all three.

Specify the green as a spot ink. A mixed ink is a recipe, so the same green comes off both presses, and it reaches a colour four process inks cannot. It costs a fifth plate, a fifth unit and a wash-up on every run at both shops. On the domestic carton, which is a four-colour job, that is a real per-thousand addition, and during the season the Sangli shop's five-unit press is the one everybody wants, so a five-colour job is also a scheduling risk.

Convert to the nearest CMYK and accept it. No extra plate anywhere. The green goes flat, and, more importantly, the two shops will not match each other, because a CMYK build is a simulation that depends on the press, the board and the day. The co-operative would be buying a colour that is different in two cities and calling it their brand colour.

Change the brand green to one sitting near the cyan primary, which reproduces almost untouched everywhere. Cheapest to print, forever. It means a colour change on the board, on the crate labels, on nine thousand printed cartons already in the shed, and a conversation with a committee of growers who chose the green in 2016 and like it.

The secretary's instinct was that spot inks are for large companies. The Pune operator's instinct was that the export carton is the one seen beside a competitor's on a shelf in Rotterdam and the domestic carton is not.

What nobody had was a number anybody could act on. The brand sheet, such as it was, held a hex value and a CMYK build typed by a designer in 2016, and neither means the same thing in two factories.

> **STOP HERE**
>
> Before turning the page: what would you have done, and what would it have cost if you were wrong? Write it down. The point of the next section is to compare.

WHAT HAPPENED

They split the decision, which is usually the wrong answer and here was the right one.

First they got a measurement. The Pune shop measured the current Sangli carton with their spectrophotometer and gave the co-operative a Lab value for the green as it was actually printing. That single number became the reference for everything afterwards, because it is the only colour notation that means the same thing at two suppliers.

Then they chose per carton. The export carton, longer run and seen beside competitors, takes the green as a spot ink; the plate cost spread over ninety thousand pieces was a small figure per carton. The domestic carton runs four-colour process with the closest CMYK build, specified as within Delta E 4 of the measured Lab value, measured under D50.

They wrote a one-page sheet carrying the Lab value, the spot reference, the CMYK build and the tolerance, and gave the same sheet to both printers. It cost an afternoon.

The brand green was not changed. But the committee also agreed that if the co-operative ever rebrands, the shortlist will be checked against a printed chart before anybody falls in love with anything, which is the part of this that generalises.

Why does a spot ink solve the two-printer problem where the best possible CMYK build does not?

A spot is a physically mixed ink made to a recipe, so the same recipe on two presses gives the same ink. A CMYK build is a simulation assembled from four inks by a particular press, on a particular board, on a particular day, under a profile that may differ between the two shops. The build can be made very close at one supplier and will drift at the other, and neither shop is doing anything wrong when it does.

The brand sheet carried a hex value and a CMYK build. What one value would have prevented most of this, and why?

A Lab value. Hex describes what one screen does, and a CMYK build describes what one press and one paper do, so neither is portable. Lab is defined by measurement of human vision rather than by any device, so a printer in another city can measure the sheet and say how far off it is in Delta E. That turns match it exactly, which nobody can act on, into a specification with a number and a tolerance.

Under what circumstances would changing the brand green have been the right answer?

If the mark were new, or if the printed stock in circulation were small, or if the co-operative expected many suppliers and short runs where a fifth plate could never be amortised. A colour chosen near a process primary reproduces well nearly everywhere and costs nothing extra for the rest of the organisation's life, and a colour chosen in the corner of the RGB cube commits it to spot inks permanently. The decision is cheap at the start of a brand and expensive nine years in.

MODULE 4 · TEST

Check what stuck

6 questions, one right answer each. This one is for you, not for a record: answers and the reason behind each are at the back. If two questions from the same module go wrong, re-read that module before the exam rather than after.

1 A brand colour printed as a spot ink and the same colour built from CMYK match under the shop's tube light and differ in daylight. What is this called, and what does it imply?

A. Dot gain, which changes with the illuminant
B. A gamut clip, which reappears under a wider illuminant
C. Metamerism: two spectra that match under one light only
D. Optical brightening in the paper, which only fluoresces indoors

2 A photograph is tagged sRGB, was actually shot in Adobe RGB, and its saturated colours look oddly flat. What do you do?

A. Convert it to Adobe RGB, which restores the missing saturation
B. Assign Adobe RGB, which changes what the existing numbers mean
C. Convert it to CMYK and back to RGB, which rebuilds the wider gamut
D. Raise the saturation until it matches the original on screen

3 Which rendering intent for a logo with specified brand colours, and why that one?

A. Perceptual, because it compresses the whole gamut smoothly
B. Saturation, because a brand colour should be as vivid as possible
C. Absolute colorimetric, because it simulates the source's own paper white
D. Relative colorimetric, because in-gamut colours are left exactly alone

4 An uncoated job was converted with a coated press profile and printed dark and muddy. Why?

A. The coated profile assumes less dot gain than the press actually produces
B. Uncoated stock has the wider gamut, so the colours clipped on conversion
C. The uncoated press runs a finer line screen than the coated profile expects
D. Uncoated stock takes less total ink, so the blacks came out weak

5 Where two knocked-out colours meet, which one is spread into the other, and why that one?

A. The darker one, because spreading it hides the overlap
B. Both equally, so the overlap is centred on the shared edge
C. Whichever object is on top, because the RIP processes that one last
D. The lighter one, because the eye reads an edge from its dark side

6 A white logo has vanished from the middle of a photograph on press, and the PDF looked perfect on screen. What happened?

A. The object was set to overprint, and white is an absence of ink
B. The logo knocked out, so the photograph printed straight through it
C. The white became 0/0/0/0 during flattening and was discarded
D. The RIP trapped the white into the photograph and choked it away

MODULE 5

Designing a mark

The whole job of a logo, from the questions you ask before drawing to the folder you hand over. Mark types and what each costs to draw and to defend, sketching and narrowing, real construction versus the golden-ratio story, lockups and clear space, hostile backgrounds, the law around naming and ownership stated honestly, presenting work, and the delivery pack that stops the client's next supplier redrawing it in Word.

> BY THE END OF THIS MODULE YOU CAN
>
> Take a logo from brief to delivered asset pack — interrogate the brief against the surfaces it will appear on, choose and draw a mark that holds in one colour at 16 pixels and on a hostile background, define its lockups and clear space, and hand over files plus a usage sheet a stranger can work from.

36 · 10 min

If it does not work in black at 16 pixels, it is not a logo

THE ONE THING TO KEEP

Design the mark in flat black at small size first, because embroidery, stamps, vinyl and one-colour print cannot reproduce anything else, and the colour version can be added afterwards.

The two sizes that decide everything

A logo will be seen at 16 pixels — a browser tab, an app icon in a crowded folder, a WhatsApp profile picture on a 5-inch screen at arm's length. And it will be seen at 40 feet on the side of a building. Almost every design decision that goes wrong goes wrong at the small end.

So test there first. Before you show anyone anything:

1. Export at 16, 32 and 64 pixels.
2. Put them on your phone and hold it at arm's length.
3. If you cannot tell what it is, the mark is too complicated, no matter how good it looks at 800 pixels.

Then do the second test, which is harder to argue with.

The black-and-white test

Set the whole mark to 100% black on white. Then invert it: white on black.

If it still reads in both, you have a logo. If it collapses, you have an illustration that happens to have a company name near it.

This is not an aesthetic preference. It is a list of the places your mark actually has to go:

- Embroidered on a shirt pocket in one thread colour
- A rubber stamp on a delivery challan
- Engraved on a pen given away at a trade fair
- A single-colour screen print on a jute bag
- A newspaper classified advert
- A thermal-printed invoice
- Etched on glass, cast in a mould, cut in vinyl

None of these can render a gradient, a shadow or a transparency. All of them are ordinary requests from ordinary clients.

Where a gradient can go, and where it cannot

Will reproduce a gradient	Will not, at any price
Litho on coated stock	Embroidery in one thread colour
A digital toner press	A rubber stamp on a delivery challan
Large-format inkjet	Vinyl cut for a shop shutter
A screen, at any size	Engraving on a trade-fair pen
	A thermal-printed invoice
	A newspaper classified advert

Every surface on the right is an ordinary request from an ordinary client, and none of them can carry a gradient, a shadow or a transparency. That is why the mark is built in flat black first and the colour version added afterwards.

The inversion matters separately because ink spreads. A light mark on a dark ground closes up: thin counters fill in, hairline gaps disappear, and the mark that looked crisp goes muddy. If your counters vanish when inverted, open them up now.

Why a gradient mesh is a liability

Gradient meshes, transparency, blend modes and drop shadows are lovely on a screen and are the four things that cause the most trouble downstream.

- They cannot be embroidered, cut, stamped or one-colour printed at all.
- They force **transparency flattening** on export to older formats. Flattening slices the artwork into regions and rasterises the ones containing transparency at a fixed resolution. The result is the classic complaint: a

logo comes back from the printer with a faintly visible grey or white box behind it, or a hairline seam across the middle of a soft shadow. The box is the boundary of the flattened region.

- They are heavy to process, and small shops' RIPs are old.

None of this means never use colour or depth. It means: **build the mark as flat black first.** Get it working with nothing but silhouette. Then add colour, and keep the flat version as a deliverable in its own right. If the black version is boring, colour will not rescue it — it will just be a boring logo in nice colours.

A useful crude limit: if your mark needs more than about 150 nodes, it is an illustration. Real marks are usually under 50.

Optical corrections, which nobody measures and everybody sees

Geometry and perception disagree, and perception wins.

- A circle set beside a square of the same height looks smaller. Overshoot the circle by 1–2%. Every typeface does this — the "O" is taller than the "H" and you have never noticed.
- A horizontal bar placed at the exact vertical centre of a letterform looks too low. Raise it slightly.
- A triangle pointing right, centred in a circle by its bounding box, looks off to the left, because its visual mass sits left of its geometric centre. Nudge it right. This is why the play button on every media player is not mathematically centred.

You do this by eye, at 100%, then again at 16 pixels.

Grids are the reverse of this: the golden-ratio circle diagrams people post over famous logos are decoration applied afterwards. No one has ever chosen a mark because a set of circles fitted it.

What you actually hand over

The deliverable is not one file. It is a small set, and delivering it properly is a large part of what separates a paid job from a favour:

- Master vector working file (`.cdr`, `.ai`, `.svg`, `.afdesign`)
- Vector PDF with fonts outlined, for print and for anyone who has nothing
- SVG for the web
- PNGs with transparency at 1024, 512, 256, 64, 32 and 16 px
- **Single-colour black** version and **white knockout** version, separately
- A one-page usage sheet: minimum size, clear space around the mark expressed in units of the mark itself, the colour values in CMYK, RGB and hex, and two or three things not to do

All of it exports from Inkscape. There is no part of this list you need a subscription for.

The AI question, briefly

Logo generators produce either a raster image or an auto-traced vector with hundreds of untidy nodes, which fails everything in this lesson at the small end. There is also a real duplication risk: a generator trained to produce logos will produce marks close to ones it has seen, and a mark that resembles an existing registered trademark is a legal problem, not a style problem.

If you use generated output at all, treat it as a sketch and redraw the final mark yourself with primitives and booleans. Ownership of generated output is genuinely unsettled and differs by country — Making Things With AI sets out the actual state of the disagreement. For a mark, route around it: trademark protects a logo through use and registration regardless of how it was drawn, and that is the protection a client is really buying.

Today

Take your current logo. Set it to 100% black. Export at 16 px. Look at it on your phone. Most people find something to fix in the first thirty seconds.

BEFORE YOU MOVE ON

A logo with a soft gradient and a drop shadow is approved. The client then orders embroidered shirts, a rubber stamp and a one-colour newspaper advert. What should have happened at design time?

A. The logo should have been supplied at higher resolution so the gradient survives at the small sizes those three uses require.
B. A solid one-colour version of the mark should have been designed and tested first, because embroidery, stamps and one-colour print can only reproduce flat shapes; the gradient version then becomes one option rather than the only artwork.
C. The gradient should have been converted to a fine halftone pattern, which any of the three processes can reproduce.
D. Spot colours should have been chosen instead of process colours, so the gradient reproduces consistently across the three suppliers.

37 · 8 min

The questions to ask before you draw anything

THE ONE THING TO KEEP

The surfaces list — every specific place the mark will appear — decides more of a logo's design than any adjective the client uses, so collect it first, convert vague directions into choices between real examples, and send a one-page written brief back for confirmation.

Most logo work goes wrong before any drawing

A client says "we need a logo, make it modern, make it pop". If you start drawing from that, you will produce three concepts, they will reject all three, and neither of you will be able to say why. The failure is not in the drawing.

There is a list of questions, and asking them takes twenty minutes.

The surfaces list, which is the most useful question

> *Where will this appear, specifically? Name every place you can think of.*

Write down the answers. The list decides most of the design.

- A shop board 3 m wide → the mark must hold at distance and probably be made in vinyl or acrylic letters, so no fine detail, no gradients.
- Embroidered shirt pockets → one to three thread colours, nothing below 1.2 mm.
- A rubber stamp → pure black, no tints, nothing thin.
- An app icon → square, works at 16 px, no wordmark.
- A vehicle → deforms over curves, gets dirty, seen at speed.
- Invoices from a thermal printer → black only, low resolution.
- A sponsor board with thirty other logos → must survive being 40 mm wide next to a competitor's.

Clients who say "I don't know, everywhere" have just told you to design for all of these, which means flat, simple and one-colour-capable. That is a design decision derived from a question, not from taste.

The rest of the list

About the organisation

- What do you actually do, in one sentence a stranger would understand?
- Who is the customer? Where do they see you first?
- What is the full legal name, and what do people actually call you? These differ more often than you would think, and the mark should usually carry what people say.
- Is the name fixed? If they are still arguing about it, stop until they have decided.

About the current situation

- Is there an existing mark? Who owns the artwork, and is the source file available?
- Which parts of it must survive — a colour, a symbol, a name? Sometimes there is an equity you must not throw away.
- Is this a refresh or a replacement? These are different jobs at different prices.

About constraints

- Is the colour count constrained by how you print? (A shop printing its own bags in one colour has a real constraint.)
- Any language other than English? A bilingual lockup is a different design problem and must be planned, not retrofitted.
- Is there a sector convention you want or want to avoid — green for organic, a caduceus for medical, a tick for sportswear?

About the process

- Who signs this off? Get the name. Work approved by a manager and then shown to the owner's family is a project with two clients.
- How many revision rounds are we agreeing on?
- When do you need it, and what is the first thing it will be printed on?

The competitor look

Do this before sketching, not after. Collect ten competitors' marks, print them at 30 mm on one sheet in black and white, and look.

You will learn two things immediately. First, the conventions of the sector — which you can follow deliberately or break deliberately, and either is a decision. Second, what is already taken: if six of the ten use a swoosh, the seventh swoosh is invisible.

Keep the sheet. It is also the most persuasive thing you can put in front of a client who wants a swoosh.

Turning "make it pop" into something workable

When you get an unusable direction, do not argue with it. Convert it into a choice.

> *"When you say modern, do you mean something like this, or like this?" — and show two real examples.*

Adjectives are unreliable because they mean different things to different people. Examples are not. Ask the client to bring three marks they admire from *any* sector and three they dislike, and to say one sentence about each. Ten minutes of that is worth an hour of adjectives, and it gives you something to point at later when a decision is questioned.

Write it down and send it back

One page: what they do, where it appears, constraints, who approves, how many rounds, deadline. Send it and ask them to confirm.

This is the single cheapest piece of professional practice in the trade. It catches the misunderstandings while they cost nothing, and when somebody later says "I never said one colour", the page says otherwise.

BEFORE YOU MOVE ON

A client asks for a "modern, premium" logo for a bakery, and mentions in passing that they print their own paper bags on a single-colour flexo machine. Which of these follows most directly from that passing remark?

A. The mark must be designed and tested as a solid one-colour form, because the packaging cannot reproduce anything else.
B. A premium look requires foil or a spot colour, so the client should be advised to change printers before the design work starts.
C. The logo should be supplied as a high-resolution PNG with transparency, since flexo plates are made from bitmap images.
D. Colour is unconstrained because the bags are only one item among many surfaces the mark will appear on.

38 · 8 min

Five kinds of mark, and what each one costs

THE ONE THING TO KEEP

Wordmarks read immediately and fail small, monograms work small and are hard to make distinctive, pictorial marks cross languages and cost iterations, abstract marks are ownable but carry no meaning until somebody pays to build it — and a combination mark is the honest default for most small clients.

The taxonomy, and why it is useful

Not a style guide — a set of trade-offs. Each type costs a different amount to draw, to protect and to reproduce.

Wordmark. The name, set and refined as type. Google, Coca-Cola, Tata. Cheapest to draw well, because the letterforms exist and the work is spacing and customisation. Reads immediately at any size because it *is* the name. Needs no explanation. Weak at very small sizes — a wordmark at 16 px is illegible — so it usually needs a separate icon eventually.

Lettermark or monogram. Initials, drawn as a mark. HP, IBM, CNN. Compact, works small, and useful when the full name is long. The problem is crowding: three-letter combinations are heavily used and a new one is hard to make distinctive. Expect to check a trademark register.

Pictorial mark. A recognisable thing, simplified. Apple, Shell, the WWF panda. Crosses languages, works small, memorable. Expensive to draw well, because a simplified representational form needs many iterations. And if the thing is generic for the sector — a tooth for a dentist, a house for an estate agent — it is weak both as design and as a trademark.

Abstract mark. A geometric form with no representational meaning. Pepsi's circle, Chase's octagon, Airbnb's Bélo. Distinctive, ownable, works at any size. It carries no meaning on day one, so it requires the organisation to build meaning into it over years. This is why an abstract mark is a poor recommendation for a small business with no marketing budget — it is a promise to spend money you do not have.

Combination mark and emblem. Mark plus name, either side by side (combination) or fused (emblem, like a crest or a badge). The combination is the practical default: use the lockup normally, the icon alone when small. The emblem is the trap — Starbucks and Harley-Davidson are emblems and both are illegible at 16 px, which is why Starbucks kept simplifying theirs.

Five kinds of mark, and what each one costs

Wordmark	Cheapest to draw well. Illegible at 16 pixels
Lettermark	Works small. Three-letter space is crowded
Pictorial	Crosses languages. Many iterations to draw
Abstract	Ownable, and means nothing until somebody pays
Combination	Lockup plus separable icon. The honest default

This is a set of trade-offs rather than a style menu. The question that usually settles it: if you could only ever use one version, in black, 12 mm wide, on a form, what would it be?

Which to recommend

The honest default for a small business is a **combination mark**: a simple icon plus a well-set wordmark, delivered as a lockup and as separable parts. It gives them something that works on a sign, on a shirt pocket and as a profile picture, which is the whole list for most clients.

Recommend a pure wordmark when: the name is short, distinctive and memorable, and the client has no need for an app icon. Recommend an abstract mark only when the client has the budget and the patience to build recognition.

What each costs to defend

This matters more than designers usually admit.

- A **descriptive or generic** pictorial mark — a loaf for a bakery — is weak as a trademark because you cannot stop other bakers using loaves.

- A **distinctive abstract** form is strong, because nobody else has a reason to use that exact shape.
- A **wordmark** is protected mostly through the name itself rather than the drawing, which means its strength depends on how distinctive the *name* is. "Quick Print" protects nothing; an invented word protects a great deal.

Say this to the client at the recommendation stage. It is the difference between a designer and someone who supplies pictures.

Reproduction cost

- Wordmark: cheap in vinyl, awkward in embroidery below 5 mm cap height, fine in print.
- Monogram: cheap everywhere, ideal for embroidery and stamps.
- Pictorial with fine detail: expensive in embroidery, unweedable in vinyl below a certain size, may need simplifying for every machine.
- Emblem with an outer ring and text: needs a redrawn simplified version for small use, so budget for two marks, not one.

The question that settles it

> *If you could only ever use one version of this, in black, 12 mm wide, on a form — what would it be?*

Answer that with the client, and the type of mark usually selects itself.

One more distinction worth naming

A **mascot** — a drawn character — is sometimes asked for and is a different commission. It is expensive because a character needs multiple poses and expressions to be useful at all, it reproduces badly at small sizes, and it dates faster than any other type. It can be excellent for a food business or anything aimed at children, and it should be priced as illustration plus a logo, because that is what it is.

Similarly, a **lockup with a tagline** is not a logo with extra words. The tagline sets the minimum size, because it is the smallest text in the mark, and a tagline that is legible at 25 mm forces the whole lockup to be larger than it needs to be. Deliver the tagline version as a separate file with its own minimum size, and make clear that the mark without it is the primary.

BEFORE YOU MOVE ON

A new independent bakery with no marketing budget asks for a distinctive abstract mark with no name in it, because they admire a large sportswear brand's logo. What is the strongest professional objection?

A. Abstract marks cannot be registered as trademarks because they are not representational.

B. An abstract mark carries no meaning until sustained spending builds recognition, which this client cannot fund.

C. Abstract marks are harder to reproduce in one colour and at small sizes than wordmarks or emblems.

D. The sportswear mark is protected, so any abstract mark designed after seeing it risks being treated as derivative.

39 · 8 min

Thumbnails, and why you never show a concept you would not ship

THE ONE THING TO KEEP

Generate thirty thumbnails at 30 mm on paper before opening any software, develop six into real geometry, and present exactly three you would all be happy to ship — with rationale about function rather than story, because a functional reason cannot be argued with.

Quantity first, on paper

The first twenty ideas are the obvious ones, and they are obvious because everybody has them. Getting past them requires producing them and discarding them, which is faster on paper than on screen.

The method:

1. A sheet of A4, divided into 30 boxes about 30 mm square.
2. Fill every box. Thirty marks, in perhaps forty minutes. Nothing careful, nothing shaded, no lettering unless the lettering is the idea.
3. Do a second sheet. The second sheet is where the work is.
4. Go away for a few hours. Come back and circle six.

The box size is deliberate. At 30 mm you physically cannot draw detail, so you are forced to work in silhouette — which is the same discipline the 16-pixel test enforces later.

Everyone who does this seriously says the same thing: the good idea is somewhere between sheet two and sheet four, and it is never on sheet one.

Why not to start on screen

On screen, the tools push you towards refinement. You will spend twenty minutes adjusting one curve on a mediocre idea instead of generating the next fifteen ideas. Paper has no undo, no alignment guides and no temptation to pick a colour, so it produces more ideas per hour.

If you have no paper, use the crudest tool available — a brush in Krita at a large size, on a small canvas. The point is the constraint, not the material.

Developing six into three

Take the six circled thumbnails into vector. Build each one properly: primitives, booleans, real geometry, black only.

Then apply the tests you already know:

- Black on white, and white on black.
- 16 px on a phone.
- Flipped horizontally.
- Beside the competitor sheet at the same size.

Two or three will fail one of these outright. Good — that is the test doing its job, before anyone has seen them.

Show three, and mean all three

This is the rule that separates a professional presentation from a portfolio dump: **never show a concept you would be unhappy to see used.**

Clients choose. If you show one strong concept and two weak ones to make the strong one look better, you will sometimes get a weak one chosen — and then you have to produce it. Everyone who has done this has a logo in the world they are quietly ashamed of.

Three is the right number. One looks like you did not explore. Five or more makes the client feel responsible for design decisions they are not equipped to make, and the usual outcome is "can we combine the top of two with the bottom of three".

Presenting the three

Each concept gets:

- The mark, large, in black.
- The mark at 16 px and at 30 mm.
- The mark in its intended colour.
- One realistic application from the surfaces list — the actual shop board, the actual bag, not a generic mockup of a billboard in a European city.
- **Two sentences** of rationale, about function rather than story.

The rationale is where most presentations go wrong. "The upward stroke represents your journey of growth" invites an argument about whether the client feels they are growing. "This holds at 12 mm on your invoice and can be embroidered in one thread" cannot be argued with, and it is also true.

The uncomfortable part

Clients will ask to combine concepts, ask for a fourth direction, ask what their nephew thinks. This is normal and it is not disrespect; they are anxious about a decision they cannot evaluate.

The answer is to return to the brief you wrote down. "The brief says it must work embroidered in one colour at 8 mm. Concept two does; the combination you are describing does not." A written brief converts taste arguments into requirement arguments, and requirement arguments have answers.

Presenting remotely

Much of this work is now shown over a video call or as a PDF sent by email, and both weaken the presentation in the same way: the client sees the work without you beside it, at a size you did not choose, on a screen you have not seen.

Two defences. First, send a PDF rather than images, sized so each mark appears at a sensible physical size when the page is viewed whole — and say in the covering note "please view at 100%, not zoomed". Second, if it is a call, share your screen and present it rather than sending the file in advance. People form an opinion in the first four seconds of looking, and if those four seconds happen alone in an inbox on a phone, you were not there for them.

If the client insists on seeing files in advance, send one page: the recommended concept only. The other two follow in the call.

BEFORE YOU MOVE ON

A designer presents one strong concept alongside two deliberately weaker ones to steer the client towards the strong one. The client picks a weak one. Beyond the immediate awkwardness, what does this reveal about the method?

A. The rationale was written in terms of story rather than function, so the client had no basis for comparison.

B. Three concepts is too many for a client without design training, and one concept should have been presented instead.

C. Any concept shown can be chosen, so presenting work you would not ship guarantees some clients will ship it.

D. The weak concepts should have been shown at a much smaller size, so that their failure at 16 pixels was visible during the presentation.

40 · 8 min

Real construction, and the golden-ratio story

THE ONE THING TO KEEP

Real construction means one stroke unit, shared radii, aligned terminals and a grid of multiples, corrected optically by eye — and the golden-ratio circles laid over famous logos are almost always retrofitted decoration rather than how those marks were drawn.

What construction actually means

A well-built mark has internal consistency you can name:

- **One stroke weight**, or two in a deliberate relationship — say 1:2, never 1:1.13 because nobody measured.
- **Shared radii.** Every corner is 2 mm or 4 mm, not seven different values.
- **Aligned terminals.** Where strokes end, they end on the same line or at the same angle.
- **A repeating module.** The mark is built on a grid of some unit — the stroke weight itself is a good unit — so gaps are 1, 2 or 3 units rather than arbitrary.
- **Consistent angles.** If one diagonal is 30°, the others are 30° or 60°, not 31° and 58°.

You build this by drawing with primitives and typed numbers, which is what the earlier modules were for. Set the stroke weight, then make every gap a multiple of it. Set the corner radius once and reuse it.

How to make a construction sheet honestly

A construction sheet shows how the mark is built, so that somebody could re-build it. It is a real deliverable, useful for signage fabricators and for anyone redrawing it at 3 m.

1. Set the smallest meaningful unit — usually the stroke weight — as `x`.
2. Draw a grid at `x` intervals behind the mark.
3. Show every key dimension in units of `x`: "counter width 3x", "gap 1x", "corner radius 2x".
4. Show the circles and squares the mark was actually built from, with their centres marked.

A fabricator can work from that. They cannot work from a PNG.

The golden ratio story

You have seen the diagrams: a famous logo with fibonacci circles laid over it and arrows pointing at coincidences.

Almost all of them are retrofitted. The circles are placed to fit, not used to draw. Apple's designer Rob Janoff has said publicly that the golden ratio was not used in the Apple logo; the circles in those diagrams are somebody else's later decoration. You can overlay golden-ratio circles on almost any shape if you are permitted to choose where to put them and how large to draw them.

This matters for two reasons, and neither is about being clever.

It sets a false standard. Beginners believe good marks come from a mathematical construction they have not learned, and stop trusting their eye. They do not. Good marks come from consistent units, careful curves, and optical correction by eye.

Optical correction contradicts it. A circle beside a square of the same height looks smaller, so you overshoot the circle by 1–2%. A horizontal bar at the exact vertical centre looks low, so you raise it. Neither adjustment is mathematical; both are necessary. A mark built on a perfect ratio and then not optically corrected looks wrong, and a mark built on nothing in particular and optically corrected looks right.

What to do instead

Use maths where it genuinely helps:

- **Equal divisions.** Three bars, three equal gaps. Use the align and distribute tools.
- **Regular polygons and their angles.** A hexagon's internal geometry gives you 30° and 60° for free.
- **Circle packing.** Circles of equal radius touching each other place themselves; you do not guess.
- **Arithmetic relationships.** Cap height 10 units, x-height 7, stroke 2, gap 2.

Then correct by eye, and record what you did.

Presenting construction to a client

Show the construction sheet if the client is a fabricator or a large organisation with other suppliers. Do not show fibonacci overlays to justify a design decision — some clients now recognise it as decoration, and the ones who do not are being given a reason that is not the real reason.

The honest rationale is better and shorter: "Everything is built from one stroke weight and two corner radii, so it stays consistent when it is cut in vinyl, embroidered, or rebuilt at three metres by somebody who is not me."

Record your optical corrections

The one genuine problem with correcting by eye is that nobody else can see what you did, so the next person rebuilds the mark "correctly" and makes it look wrong.

Note the corrections on the construction sheet, in plain words: "circle overshoots square by 1.5% top and bottom — intentional"; "crossbar sits 2% above geometric centre — intentional"; "left counter is 0.2x wider than right to compensate for the diagonal — intentional".

Three lines, and they survive you. Without them, a well-meaning supplier will align everything mathematically and the mark will acquire the subtle wrongness that nobody can name and everybody notices.

BEFORE YOU MOVE ON

A designer builds a mark on a strict golden-ratio grid, with a circular element and a square element of mathematically identical height, and does no further adjustment. Viewers report the circle looks slightly small. What does this demonstrate?

A. The grid was applied at the wrong scale, so the circle's diameter does not correspond to the square's side in the intended ratio.
B. Circles rendered at small sizes lose apparent area to anti-aliasing, which does not affect straight-edged shapes equally.
C. Golden-ratio construction is incompatible with circular forms and should only be used for rectilinear marks.
D. Perception does not follow measurement: a circle of equal height reads smaller than a square, so it needs a 1–2% overshoot.

41 · 8 min

Lockups, clear space and the smallest it may ever be

THE ONE THING TO KEEP

Define separate lockups rather than letting somebody scale one to fit, express internal spacing and clear space as multiples of an internal measure so they stay true at any size, and arrive at minimum sizes empirically by printing on the worst printer available.

A logo is a small family, not one file

A mark that only exists in one arrangement will be rearranged by somebody else, badly. So you define the arrangements.

- **Primary lockup.** The one to use unless there is a reason not to. Usually icon plus wordmark, horizontal.

- **Secondary lockup.** Stacked, for narrow spaces — a shop board that is tall, a pull-up banner, a column on a page.
- **Icon only.** For app icons, profile pictures, favicons, embroidery on a cap.
- **Wordmark only.** For a sponsor strip or wherever the icon is too small to read.

Three or four files, each drawn and spaced deliberately. Scaling the horizontal one to fit a tall space is exactly what the stacked version exists to prevent.

Relative spacing is the whole trick

Within a lockup, space between the icon and the wordmark must be defined in terms of the mark itself, not in millimetres. Millimetres stop being true the moment somebody scales it.

The convention: express every distance as a multiple of some internal measure — the cap height, the stroke width, or the height of the icon. "Gap = 0.5 × icon height" is true at 12 mm and at 12 m.

Clear space

Clear space is the exclusion zone nobody may put anything into. Without it, your mark ends up 2 mm from a headline and looks like part of it.

Define it in the same relative way. The common conventions:

- **The height of a letter from the wordmark** — often the cap height, or the height of the "o". Easy for anyone to measure by eye.
- **A proportion of the icon** — 25% or 50% of its height.
- **The width of the widest stroke** × some number.

Draw it. In your usage sheet, show the mark inside a rectangle with the clear space marked out in the chosen unit, with the unit itself drawn beside it. A picture of the rule is understood; a sentence describing it is not.

Set it generously. People will encroach, and a generous zone encroached on still looks fine.

Minimum size, stated twice

Two different numbers, because two different worlds:

- **For print**, in millimetres of the lockup's width. "Minimum 25 mm wide."
- **For screen**, in pixels of the lockup's height or width. "Minimum 120 px wide."

They are not convertible, because screens vary and print does not. State both.

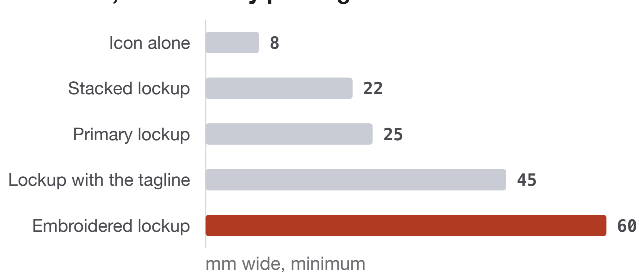
Arriving at the numbers is empirical, not theoretical:

1. Print the lockup at a range of widths on the worst printer you can find — an office laser on ordinary paper.
2. Look at it from 40 cm.
3. The smallest one where the smallest element is still clearly itself, plus 20%, is your minimum.

Do the same on a phone for the pixel figure. Actual devices, not a zoomed view on your monitor.

Then state the *separate* minimum for the icon alone, which will be much smaller, and for embroidery, which will be much larger.

Minimum sizes, arrived at by printing



These are not convertible from one another and none of them is a theoretical number. Print the lockup at a range of widths on the worst printer you can find, look from 40 cm, take the smallest one that still reads and add twenty per cent.

Responsive marks

Large organisations increasingly ship a graded set: full lockup for large use, reduced lockup for medium, simplified icon for small, single letter for tiny.

Each one is drawn for its size rather than scaled.

For a small client, do not build four. Build two — full lockup and icon — and say clearly which is used below the minimum size. Four versions that nobody maintains is worse than two that are used correctly.

What not to allow, and how to say it

The usage sheet needs a short "do not" list, and it should be short enough to read:

- Do not stretch or condense it.
- Do not rotate it.
- Do not change the colours.
- Do not add effects, shadows or outlines.
- Do not place it on a busy background without the plated version.
- Do not redraw or re-set the wordmark.

Six lines, each with a small crossed-out example beside it. People follow pictures; they skim rules.

Test the rules against a real layout

Before you publish the usage sheet, build one page using only what it permits: a letterhead, or a poster, or the shop board. If you find yourself wanting to break your own clear-space rule to make it fit, the rule is too generous and you should change it now rather than teaching the client that the rules are optional.

This also catches the common omission: a lockup that works on white but has no defined behaviour on the brand colour itself. If the primary colour is a deep blue and the wordmark is that same blue, the mark disappears on its own brand background. Somebody will try it in the first month. Decide now what happens, supply that file, and put it on the sheet.

BEFORE YOU MOVE ON

A brand sheet specifies clear space around a logo as "8 mm on all sides". The logo is later used at 2 m wide on a shop board and looks cramped against the fascia edge. What is wrong with the specification itself?

A. Clear space should be specified as a percentage of the available space rather than of the mark.
B. An absolute distance does not scale with the mark, so 8 mm is generous at 25 mm wide and negligible at 2 m.
C. Clear space rules do not apply to signage, which is governed by the fabricator's mounting requirements instead.
D. 8 mm is below the printing industry's standard minimum clear space of 10 mm for any applied mark.

42 · 8 min

Reversed, on a photograph, and in one thread

THE ONE THING TO KEEP

A knockout is transparent geometry rather than an inverted image, and a reversed mark may need its counters opened because light on dark closes them optically and ink closes them physically — so deliver designed versions for dark, photographic, single-thread and colourless surfaces rather than leaving them to be improvised.

The versions you deliver are not variations

A logo will land on a dark wall, a photograph, a coloured T-shirt, a brushed steel plate and a thermal-printed receipt. Each needs a version that was designed, not improvised by whoever is in a hurry.

The one-colour version

Build it first, as the reuse lesson insists. Then build the **knockout**: white where the mark was black, and nothing where the mark was white.

The knockout is not a colour inversion. Inverting turns the background into a solid block. A true knockout has a transparent background and white geometry, so the dark surface shows through the counters.

Check it by placing it on 100% black and looking at the counters — the hole in an `a`, the gap inside an `e`, the space between two close strokes. Light on dark visually closes gaps, and ink spread physically closes them, so a counter that measured 0.4 mm in the positive version may vanish. Open them by 10–15% in the knockout version if needed, and yes, that means two slightly different sets of geometry. Good type foundries do exactly this for their reversed cuts.

On a photograph

Three approaches, in descending order of preference:

1. **A clear area in the photograph.** Choose or shoot the image so the mark sits on a plain region. Costs nothing and always looks best.
2. **A plate.** A solid or semi-transparent rectangle behind the mark. Honest, legible, and looks deliberate if it is part of the design language.
3. **A scrim.** A gradient or a darkened band across part of the image, with the mark on top.

What not to do: put a drop shadow or an outline on the mark to make it survive. It works badly, it looks improvised, and it introduces transparency into a file that may be flattened later.

Define which of these is allowed in the usage sheet, and supply the plated version as a file so nobody has to build it.

Single-thread embroidery

Embroidery reduces the mark to a small number of thread colours and a minimum feature size — nothing below about 1.2 mm of satin width, and text below about 5 mm cap height becomes a smudge.

So a one-thread version usually needs to be **simplified**, not just recoloured: fine details removed, thin strokes thickened, small counters opened or closed deliberately. This is a separate drawing. Budget for it if the client has uniforms.

Embossing, engraving and etching

These have no colour at all. The mark reads purely as depth and edge, which means:

- Anything distinguished only by colour disappears. A two-tone mark becomes one tone.
- Very fine lines fill or fracture depending on the process.
- A mark with a heavy solid area embosses as a flat plateau and loses its interest.

A mark designed as a line drawing usually embosses well. A mark designed as a silhouette usually does not. If the client's list includes a foiled or embossed business card, check this at the design stage.

Vehicles and large-format

Vehicle graphics wrap over compound curves, so a mark placed across a body line is distorted and a mark across a door gap is cut. And a hoarding is usually printed in panels and joined — find out where the seams fall and keep the mark off them. Neither is your decision to make alone; both are a conversation with the fabricator that takes five minutes.

The test sheet

Build one file that contains the mark, at its minimum size, on nine backgrounds: white, black, 50% grey, each brand colour, a busy photograph, a coloured garment swatch, and a pale tint. Print it. Everything wrong with the mark's versions shows up on one sheet, and you can hand the same sheet to the client.

Print it twice: once on your own printer, and once on the cheapest copier you can reach. The copier version is a reasonable stand-in for a thermal receipt, a fax, a photocopied form and a newspaper advertisement, all of which are real destinations and all of which are more brutal than anything a design review will show you.

Naming the versions so they get used

The commonest failure is not that the versions do not exist. It is that they exist and nobody can tell which is which, so the person in a hurry picks the first file in the folder.

Name them by their job, not by their appearance: `acme-logo-mono-black`, `acme-logo-mono-white-knockout`, `acme-logo-on-photo-plated`, `acme-logo-embroidery-1colour`. "Positive" and "negative" are ambiguous to non-designers — half of them read "negative" as "the bad one". Spell it out.

Then say in one line where each belongs. That line is what the next lesson's usage sheet exists to carry, and it is the difference between a set of versions and a set of files.

BEFORE YOU MOVE ON

A mark that reads cleanly in black on white is supplied for a dark shopfront simply by inverting the artwork in an image editor. The installed sign has a visible rectangle of lighter material around the mark. What went wrong?

A. Inverting produced a raster file, and rasters cannot be cut in vinyl without a visible boundary.

B. The counters closed up when reversed, and the fabricator filled them, producing a solid block around the mark.

C. Inverting turned the transparent background into a solid field, so the mark arrived with a white rectangle behind it.

D. The dark shopfront material has a different gloss level, so the applied vinyl reflects differently around the mark's outer edges.

43 · 9 min

Names, trademarks and who owns the file

THE ONE THING TO KEEP

Copyright protects the drawing and a trademark protects the sign used in trade; ownership of commissioned artwork varies by country and must be settled in writing; and nobody can promise an enforceable copyright in generated output, so the client's real protection is registration and use.

This is not legal advice

Trademark and copyright law differ substantially between countries, and the position on several points below is genuinely unsettled or actively being litigated. What follows is the shape of the subject so you know when to ask a lawyer. It is not a substitute for asking one, and nothing here is advice for your situation.

Trademark and copyright are different things

They are confused constantly, and the confusion causes real problems.

- **Copyright** arises automatically in the artwork — the drawing itself — when it is created. It protects against copying the drawing. In most countries it lasts for decades.
- **A trademark** protects a sign used in trade to distinguish goods or services. It arises through use in some countries and through registration in others, and it is limited to the classes of goods and services it is registered for.

So: a client who commissions a logo may need both. Copyright in the drawing (which they may or may not own — see below) and a trademark registration for the mark as used in their sector.

Distinctiveness, which decides whether a name is worth protecting

Trademark registries everywhere apply a version of the same spectrum, strongest to weakest:

- **Invented** — a word that did not exist. Strongest.
- **Arbitrary** — a real word with no connection to the goods.
- **Suggestive** — hints at a quality without describing it.
- **Descriptive** — describes the goods. Registrable only after long use establishes that people associate it with one trader.
- **Generic** — the name of the thing itself. Never registrable by anyone.

A client set on "Quality Bakery" is choosing something they will not be able to protect and which will be invisible in search. Saying so early, once, is part of the job.

Searching before you draw

Do this before spending days on a mark:

- Search the relevant national registry. India's is the **IP India** public search; the UK, EU, US, Australia and most others have free online searches.
- Search for the name plainly, and for the design if the registry supports image search.
- Search domain availability and the main social platforms at the same time.
- Search ordinary web results for the name in the client's sector.

You are not clearing the mark — that is a professional search, and there are firms that do nothing else. You are avoiding the obvious collision that would waste a week.

Who owns the artwork

This is jurisdiction-dependent and it catches freelancers constantly.

In several common-law countries, copyright in a commissioned work **vests in the creator by default** unless assigned in writing — so the designer owns the drawing and the client has an implied licence to use it. In others, and in employment situations almost everywhere, it vests in the commissioner or employer. Some jurisdictions also recognise **moral rights** that cannot be assigned at all.

The practical consequence is the same everywhere: **write it down**. A single clause in your quotation settles it:

> *On payment in full, copyright in the final selected mark transfers to the client. Copyright in unselected concepts and in working files remains with the designer. Source files for the selected mark are supplied.*

Decide what you are actually selling. Some designers retain source files and supply outputs; some transfer everything. Both are legitimate. What is not legitimate is leaving it unsaid and arguing about it later.

Generated artwork

If any part of the mark came from a generative tool, two separate issues arise and both are unresolved.

Ownership. Several jurisdictions, including the US Copyright Office, have taken the position that purely machine-generated output without sufficient human authorship is not protected by copyright. Other jurisdictions differ, and the boundary of "sufficient human authorship" is exactly what is being argued about. The honest statement to a client is that you cannot promise them an enforceable copyright in generated output.

Similarity. A generator trained on existing logos produces marks resembling ones it has seen, and resemblance to a registered mark is a trademark problem regardless of how it was drawn or who owns the copyright.

The practical route around both: use generated output as a sketch and redraw the final mark yourself, deliberately, from primitives. Then the drawing is yours, and the client's protection comes from trademark registration and use, which is what they are actually buying.

What to put in writing

Three sentences in every logo quotation: what transfers, when it transfers, and what a trademark search was or was not done. That is the whole of this lesson's practical output.

BEFORE YOU MOVE ON

A client asks for written confirmation that they will own exclusive rights in a mark, part of which was produced by an AI image generator and then redrawn by hand. What can the designer honestly confirm?

A. Exclusive copyright, because redrawing the generated sketch by hand makes the final artwork an original human work in every jurisdiction.
B. Nothing, because no rights can subsist in any work that involved a generative tool at any stage.
C. That copyright is unsettled for generated material, and that the dependable protection is trademark registration for the mark as used.
D. Exclusive rights in the trademark sense, since trademark registration automatically confers copyright in the registered artwork.

44 · 8 min

The folder you hand over

THE ONE THING TO KEEP

Hand over a numbered folder with print, screen, source and usage material plus a plain-text README that tells a printer, a developer and an embroiderer which file each of them needs — and include a Lab value, because it is the only colour notation that means the same thing at two different suppliers.

The deliverable is a folder, not a file

A client who receives one PNG will, within six months, have somebody redraw the mark in Word, stretch it on a banner, and put it on a dark background with a white box behind it. Every one of those is preventable by what you hand over.

The folder

```
acme-logo/
  README.txt
  01-usage/
    acme-logo-usage.pdf
  02-print/
    acme-logo-primary-cmyk.pdf
    acme-logo-primary-cmyk.eps
    acme-logo-stacked-cmyk.pdf
    acme-logo-mono-black.pdf
    acme-logo-mono-white-knockout.pdf
    acme-icon-cmyk.pdf
  03-screen/
    acme-logo-primary.svg
    acme-logo-primary-1024.png
    acme-logo-primary-512.png
    acme-icon-256.png  128.png  64.png  32.png  16.png
    acme-favicon.ico
  04-source/
    acme-logo-master.ai        (or .cdr / .svg / .afdesign)
    acme-logo-master-outlined.pdf
  05-fonts/
    fonts.txt
```

Five folders, numbered so they sort in the order somebody needs them. The numbering is not decoration; it is the difference between a client finding the right file and picking whichever is first alphabetically.

What each file is for

- **PDF** is the universal print deliverable. Vector, fonts outlined, opens everywhere, and a printer with nothing else can use it.
- **EPS** because some older shops and some embroidery digitisers still ask. No transparency in it, which is fine for a flat mark.
- **SVG** for the web and for developers.
- **PNG with transparency** at declining sizes, because the client will need them and will otherwise make them badly.

- **ICO** for the favicon, containing 16 and 32 px, because browsers still want one.
- **The source file** in your working format, clearly named as the master.
- **An outlined PDF of the master** as a format-independent archive, so the mark survives even if nobody can open your `.cdr` in ten years.

The usage PDF

Four to six pages, no more. Anything longer is not read.

1. The mark, large, and the lockups.
2. Clear space and minimum sizes, shown as diagrams with the unit drawn.
3. Colours: hex, RGB, CMYK, **Lab**, and the spot reference if there is one. Lab because it is the only value that means the same thing at two different printers.
4. The one-colour and knockout versions, with a line saying when each is used.
5. Backgrounds: what is allowed, what needs the plated version.
6. The "do not" page, with six crossed-out examples.

The README

A plain text file, ten lines, at the top level. It is read by the person who opens the folder in a hurry.

```
ACME logo pack — delivered 2026-09-21, v1.0

If you are a printer: use 02-print/acme-logo-primary-cmyk.pdf.
If you are a web developer: use 03-screen/acme-logo-prima-
ry.svg.
If you are an embroiderer: use 02-print/acme-logo-mono-
black.pdf and read
  page 4 of 01-usage/acme-logo-usage.pdf for minimum size.

Colours:  Primary  #0B5FA5  /  C90 M60 Y0 K10  /  Lab 38 -3 -44
          Black    100K only for small text

Do not redraw or re-set the wordmark. The source file is in 04-
source/.
Contact: <your name and email>
```

Naming that survives

Lowercase, hyphens, no spaces, no version numbers in the filename except in the source. `acme-logo-primary-cmyk.pdf` tells you what it is. `Logo FINAL final2 (1).pdf` does not, and you will be sent it by somebody within the year.

Send it as a zip, and keep a copy

Zip the folder and send it once, with the README's contents pasted into the email so it is read before anything is downloaded.

Then keep your own copy, with the date, indefinitely. You will be asked for it again — after a hard disk fails, after the marketing person leaves, after the client changes printers. Being the person who still has the file three years later is worth more business than most portfolios.

What to leave out

A delivery pack gets worse as it gets larger, because the client stops being able to find anything in it. Three things people include that they should not:

- **Every intermediate version.** Ship the final. Keep the rest in your own archive.
- **Mockups.** The photograph of the mark on a mug is a presentation asset, not a deliverable, and somebody will send it to a printer.
- **A forty-page brand manual** for a business with four employees. They will not read it and its existence makes the six rules that matter harder to find.

Walk the pack before you send it

Open the zip on a machine that is not yours — a different laptop, a phone, a cyber café computer. Then answer three questions as if you were the client.

1. Can I find the file a printer needs, from the README alone, in under thirty seconds?
2. Does the SVG open and render correctly in a browser?
3. Do the PNGs have genuine transparency, or a white background somebody will notice on a dark slide?

Those three checks catch nearly every packaging failure, and each takes a few seconds. The pack is the last thing the client sees of your work, and it is the only part of it that their next three suppliers will ever see at all.

BEFORE YOU MOVE ON

A client's new printer emails asking for "the logo in high resolution". The designer's delivery pack contains PDFs, an SVG, PNGs and the source file. What is the most useful reply?

A. Send the 1024 px PNG, since that is the highest-resolution raster file in the pack and resolution is what was asked for.
B. Send the source file, since the printer can then adjust the artwork for their own press requirements.
C. Send the print PDF and note that vector artwork has no resolution, so they need the finished size instead.
D. Ask the printer to specify a dpi figure so the correct PNG can be exported from the source file at that resolution.

CASE STUDY · MODULE 5

The peacock the uniforms could not hold

A new English-medium school in Rajkot: six hundred pupils, a shop board, and an embroidered pocket on every shirt.

The trust running the school approved a mark in February. It is a peacock's eye feather: fourteen fine barbs radiating from an eye, with the school's name set below in a light sans. At 80 mm on the prospectus cover it is genuinely good, and it won partly because the chairman's daughter draws peacocks.

The designer had shown three concepts and would have been content to ship any of them, which is the rule he works to. This was the one chosen.

In May the uniform order went out to a supplier in Rajkot: eighteen hundred shirts, chest pocket embroidery, to be placed within eleven days so the garments arrive before the school year starts.

The supplier's digitiser looked at the artwork and made three observations, none of them a matter of opinion.

At the pocket size of 45 mm, the barbs are 0.7 mm wide. A satin stitch works between about 1.2 mm and 12 mm; below 1.2 mm the needle simply perforates the fabric into a line, and on a knit polo the fabric pulls inward as the thread is drawn, so the drawn width is not the finished width. The school's name sits at 3.8 mm cap height in the lockup, and embroidered text below about 5 mm turns into a smudge. And the mark uses five colours, each of which is a machine stop on every single garment.

He could digitise it. He said so. He also said what would come back.

The designer had three options and eleven days.

Simplify the primary mark. Remove the barbs, thicken the strokes, drop the name to a heavier weight. The result works everywhere, including on the shirt. It is also visibly not the mark the trust approved in February, and the chairman will be looking at a different peacock.

Draw a second mark for embroidery and small use. Keep the approved mark for the prospectus, the board and the website, and draw a separate single-thread version at 45 mm: fewer barbs, thicker strokes, no tagline. Two days of drawing, a fee the trust has not budgeted, and two marks in the world, which the stationery shop near the gate will certainly confuse.

Run it as drawn. The supplier will do it if instructed. Eighteen hundred shirts, and the smudge arrives on every child's pocket for the year.

The trust's position, stated by the honorary secretary on the phone, was that the logo had been approved, paid for and finished, and that a designer asking for more money in May for something approved in February looked like a designer who had not done his job in February.

That is an uncomfortable position to argue against with words, because it is half true. The surfaces list should have been collected before anything was drawn, and embroidered pockets on eighteen hundred shirts is not an obscure requirement for a school. Had the question been asked in January, the peacock would have been drawn with four barbs from the start and nobody would have experienced it as a loss.

There was also a quieter argument against the second mark, and it came from the school's office rather than from the trust. The stationery shop outside the gate prints the diaries, the receipt books and the identity cards, and whoever is in a hurry there opens the folder and takes the first file. A school with two peacocks in one folder will get the wrong peacock on something within a term. Large organisations do ship graded sets of marks, drawn separately for each size, but they also have somebody whose job it is to police them, and a six-hundred-pupil school does not.

The designer's own view, which he kept to himself during the call, was that he would rather have one mark that worked everywhere than two that were correct and confusable. What stopped him proposing that outright was that the chairman had chosen the fourteen-barb version in a room full of people.

STOP HERE

Before turning the page: what would you have done, and what would it have cost if you were wrong? Write it down. The point of the next section is to compare.

WHAT HAPPENED

He stopped arguing and bought an object.

The supplier charged six hundred rupees to run two sew-outs on the actual poly-cotton the shirts are made from: the approved mark at 45 mm exactly as drawn, and a simplified version he drew that evening. Both came back in two days.

He took them to the trust meeting and put them on the table without a slide deck. The approved mark's barbs had closed into a dark blur and the school's name was a line of texture. The simplified one read cleanly at arm's length. Nobody argued, including the chairman.

The trust approved a second mark as a separate line on the invoice, described as uniform mark, single thread, 45 mm minimum. He drew it properly rather than thinning the original: four barbs instead of fourteen, strokes at 1.6 mm at pocket size, and the name dropped entirely, because the shirt already says the school on the collar label.

He then did the thing that makes it stick. The usage sheet gained a page naming which file goes where, in plain words: the uniform supplier gets the embroidery file, the printer gets the primary, the sign fabricator gets the primary, and the primary is not to be used below 25 mm. The files were renamed by job rather than by appearance.

Two years on, the prospectus and the board still carry the full peacock, and the shirts carry the other one, and nobody at the school has ever needed to think about it.

Why is a sew-out more persuasive than exporting the mark at 16 pixels and showing it on a phone?

Both are honest tests and the 16-pixel export is free, so it should be run first. The sew-out wins the argument because it is the actual process on the actual fabric, and it is an object the client is holding. A screen test invites the reply that it will look better when it is real. Thread on cloth does not leave that reply available, and in this case the decision was blocked by a person rather than by a fact.

The trust's objection was that the mark had already been approved. What earlier step would have removed the objection entirely?

The surfaces list, collected before any drawing began. Asking where the mark will appear, specifically, and writing the answers down would have produced embroidered shirt pockets as a named requirement, and a requirement in a written brief is something both sides agreed to. Design decisions defended by reference to a brief are requirement arguments, which have answers; design decisions defended by taste are not.

If the trust had refused to pay for a second mark, what is the least bad option and why?

Simplify the primary mark itself and deliver one mark that works everywhere. A mark that only holds at 80 mm is not really a logo, and the school's most numerous application is eighteen hundred shirt pockets. Running the approved artwork as drawn is the one option to refuse: it produces a visible failure on every garment for a year, and it would be the designer's name attached to it.

MODULE 5 · TEST

Check what stuck

6 questions, one right answer each. This one is for you, not for a record: answers and the reason behind each are at the back. If two questions from the same module go wrong, re-read that module before the exam rather than after.

1 Why is a mark built in flat black at small size before anything else?

A. Black ink is cheaper, so a one-colour version saves the client money
B. Screens render black more accurately than colour at small sizes
C. Embroidery, stamps, vinyl and one-colour print can hold nothing else
D. A black mark can be inverted automatically to give the dark-background version

2 A client says the mark will be used 'everywhere, I don't know'. What have they told you?

A. Nothing useful, so work from the adjectives in the brief instead
B. To deliver a graded responsive set of four separate marks
C. To choose an abstract mark, since abstract suits any context
D. To design flat, simple, and capable of working in one colour

3 Why is an abstract mark a poor recommendation for a small business with no marketing budget?

A. It cannot be registered until long use has established it
B. It reproduces badly at small sizes and in a single colour
C. It carries no meaning until somebody spends years building it
D. It is by a wide margin the most expensive type of mark to draw

4 Why is clear space expressed as a multiple of something inside the mark rather than in millimetres?

A. Millimetres stop being true the moment somebody scales the mark
B. Printers cannot measure millimetres accurately on a running press
C. Relative units allow the mark to be stretched to fit its space
D. Millimetres are ambiguous between the two definitions of an inch

5 What is wrong with producing the reversed version of a mark by inverting the positive one?

A. Inversion changes the colour values, so the white is no longer 0/0/0/0
B. Inverting gives a solid block; a knockout is transparent geometry
C. Inverted artwork cannot be saved as a PDF carrying transparency
D. The counters open rather than close, so the mark reads too light

6 Why does a usage sheet carry a Lab value beside the hex, RGB and CMYK figures?

A. Lab has a wider gamut, so it prints more saturated colours
B. Lab converts straight into a spot ink formula without needing a profile
C. PDF/X-4 requires a Lab value in the output intent
D. Lab is the only one of the four that means the same at two suppliers

MODULE 6

Documents that survive a late change

Multi-page work, where the difference between a tool and a toy is what happens when the client changes their mind on page 60. Choosing a layout application honestly, deriving a grid from the type rather than a template, masters and sections, threaded text and overset, linked images and effective resolution, tables that behave like tables, automatic contents and cross-references, data merge for five hundred cards, and how a book is actually bound.

> BY THE END OF THIS MODULE YOU CAN
>
> Build a multi-page document that survives a late change — masters, a style family, a grid derived from the body text, linked images at verified effective resolution, an automatic contents list and a data merge — and specify a bound job with the right page count, spine width and gutter allowance.

45 · 9 min

Eighty pages, one font change, six hours of your life

THE ONE THING TO KEEP

Styles exist so a change is made once in a named place rather than eighty times by hand, and a style that does not update is almost always carrying a local override.

The afternoon this lesson exists to prevent

A designer lays out an 80-page college prospectus. Each page is built by hand: text boxes placed individually, the running head copied onto every page, the page number typed. It looks fine.

Then the client says the body font should be Lato, not Poppins.

If the file was built with styles, that is one dialog and about forty seconds. If it was not, it is six hours of clicking, and three paragraphs somewhere in the middle will be missed and go to press wrong.

The whole reason layout applications exist as a separate category from drawing applications is this problem, and the features that solve it.

Why layout is a separate category of tool

A layout application

- Text flows from frame to frame across pages
- Masters carry the running head and the folio
- Page numbers renumber themselves
- Preflight reports overset text and broken links
- Package gathers images and fonts into one folder

A drawing application

- Every text box is an island
- Repeating furniture is copied by hand
- Page numbers are typed
- Nothing tells you a paragraph is missing
- Perfectly reasonable up to about twenty pages

None of the left-hand column is about drawing better. All of it is about surviving a change made after the document is built, which is where design work actually happens.

What a layout application gives you that a drawing application does not

- **Linked text frames.** Text flows from one frame to the next and across pages, so adding a paragraph on page 3 reflows everything after it. In a drawing program, every text box is an island.
- **Master pages.** Running head, folio, margin guides and any repeating furniture live on a master. Change it once, it changes on all 80 pages.
- **Automatic page numbers**, which renumber when you insert a page.
- **Styles**, which is the rest of this lesson, and **tables** that behave like tables.
- **Preflight**, a live check for missing images, overset text, RGB pictures and missing fonts.
- **Package / Collect for Output**, which gathers the document, every linked image and every font into one folder to send.

The tools:

- **Adobe InDesign** — the industry reference. Subscription.
- **Affinity Publisher** — very capable, was a one-off purchase, terms changed after the Canva acquisition in 2025; check.
- **Scribus** — free, open source, mature, and its PDF/X export is excellent. The interface is dated and some things are clumsy. It has laid out a great many real books.
- **CorelDRAW** does multi-page and has paragraph styles, and is perfectly reasonable up to about twenty pages. Its text flow model is weaker, so a long document is harder work than it needs to be.
- **LibreOffice Writer** with proper styles will produce a decent 200-page interior for a novel, and it is free and runs on anything.

A style is a name for a set of decisions

That is the whole idea. "Body" means 10/14 pt Lato, ragged right, 0 space before, 3 mm space after, hyphenation on. Once it has a name, you can change the decisions behind the name and every paragraph carrying it updates.

The value is not that applying a style is fast. It is that **the change is centralised**. Design work is mostly revision, and revision is where unstyled documents fail.

Three kinds:

- **Paragraph styles** for whole paragraphs: body, heading, caption, list item.
- **Character styles** for spans inside a paragraph: bold run-in lead, a different font for a botanical name, a colour on a cross-reference.
- **Object styles** for frames: fill, stroke, text inset, corner radius.

And **based-on** inheritance is what makes it worth doing properly. Define "Body". Then define "Body Indent" as *based on Body*, changing only the first-line indent. Then "Body Small" as *based on Body*, changing only the size. Change Body's typeface and all three follow. Build your styles as a family, not as a list.

Colour works the same way: define a swatch, name it, use it everywhere, and a brand-colour change becomes one edit.

The mistake that makes people distrust styles

You change the style. Thirty paragraphs do not update.

Almost always the cause is a **local override**. Someone selected that text and set the font directly, on top of the style. The direct formatting wins, and the style change never reaches it.

InDesign shows a + next to the style name in the Paragraph Styles panel when the selection has overrides. **Alt-click** the style name to apply it and clear overrides at the same time. Scribus and Affinity Publisher have equivalent behaviour with different names.

The discipline: format by applying a style, never by reaching for the font dropdown. If you need a variation, make a style for it. This feels slow for the first ten minutes of a document and saves the whole afternoon described at the top.

Two more things worth setting up on day one

A baseline grid. Set it to your body leading — if the body is 10/14, the grid is 14 pt — and align the body style to it. Text in adjacent columns then sits on exactly the same lines, and a spread looks composed rather than approximately aligned.

Linked, not embedded, images. A linked image stays on disk and the layout holds a reference, so the document stays small and updating the photograph updates the layout. Embedding makes the file self-contained but bloated and stale. Use links, then Package before sending.

The one that pays

Data merge. Five hundred visiting cards with different names, or 300 certificates, or price labels for a whole shop, from a CSV file. InDesign has Data Merge, CorelDRAW has Print Merge, Scribus does it with the Generator script. Set up the template once, point it at the spreadsheet, produce 500 personalised pages in a minute.

This is well-paid, unglamorous, repeatable work that almost no one advertises being able to do. If you learn one thing from this lesson beyond styles, learn this.

Leave imposition alone

Do not build your own printer's spreads or gang up pages unless you have been asked to. Supply single pages at trim size in reading order, with bleed. The press imposes for its own machine, paper and folding, and a helpfully pre-imposed file usually has to be taken apart again.

Today

Open any multi-page document you have made. If there are no named paragraph styles in it, rebuild the first three pages with them. You will finish the remaining pages faster than the ones you already did.

BEFORE YOU MOVE ON

An 80-page catalogue is built with paragraph styles. The client changes the body typeface, the designer edits the Body style, and about thirty paragraphs stay in the old font. What is the most likely reason?

A. Those paragraphs carry local overrides applied on top of the style — someone selected the text and set the font by hand — so the style change never reached them; clearing the overrides restores the link.
B. Those paragraphs use a different style that was never based on the Body style, so they were never going to inherit the change.
C. The new typeface lacks the weights those paragraphs used, so the application silently kept the old font for them.
D. Those paragraphs sit on a master page, and master pages do not inherit document styles.

46 · 8 min

Choosing a layout tool, honestly

THE ONE THING TO KEEP

Choose the layout tool from the document's requirements — press colour and bleed, structural features like contents and cross-references, and budget — and accept that Scribus is a genuine free answer for press work rather than a consolation prize.

The question is what the document is

Nobody needs the "best" layout application. They need the one whose limits do not sit in the middle of their job. So start from the document.

InDesign

The industry reference, and what nearly every publishing and agency job advertisement names. Threaded text, full style system with nesting and GREP styles, tables with styles, data merge, cross-references, indexing, footnotes and endnotes, live preflight, package for output, and the best justification engine available outside academic typesetting.

Subscription only. If you are working for agencies or publishers, you will need it eventually, and there is no substitute that reads and writes `.indd`.

Affinity Publisher

Genuinely capable: masters, styles, threaded text, tables, data merge, and the ability to open the other Affinity apps' tools inside it. Weaker on very long documents — indexing and footnotes arrived later than the rest and are less developed — and it cannot open `.indd`. It was a one-off purchase for years; after Canva's acquisition of Serif the apps were relaunched in 2025 with free access and a paid tier for some features, so confirm the current terms rather than planning around a remembered price.

Scribus

Free, open source, mature, and it has laid out a great many real books. Its **PDF/X export and colour management are the best of the free tools**, which is why it is the export stage of the free press chain. Real styles, masters, threaded text, spot colours, and a preflight verifier.

The honest limits: the interface is dated and some operations take more clicks than they should; table handling is weaker than InDesign's; there is no paragraph-level composer, so justified text is line-by-line; and complex OpenType features are less accessible. None of these stops a book.

CorelDRAW

Multi-page, paragraph styles, print merge, and an entire trade that uses it for eight-page brochures and menu cards. Its text flow model is weaker than a dedicated layout application's, so a 96-page document is genuinely harder work than it needs to be. Up to about twenty pages it is perfectly reasonable, and it is what the shop already has.

LibreOffice Writer

Free, runs on anything, and with real paragraph styles it will produce a decent 300-page novel interior with an automatic contents list and an index. What it cannot do is press-grade colour, bleed with crop marks, or precise image placement. For text-only interiors it is a legitimate answer and it is often the right one for a self-publishing author with no budget.

LaTeX

Free. Sets the best justified text of anything on this list, handles footnotes, cross-references, bibliographies and mathematics with no effort, and produces identical output on any machine. It is the correct tool for a thesis, a technical manual or a mathematics textbook.

The cost is that visual control is indirect: you describe structure and the system decides placement, and fighting it is unpleasant. If you want to move something 2 mm to the left because it looks better, LaTeX is the wrong tool.

HTML and CSS with Paged Media

Write the content as HTML, style it with CSS including `@page` rules, and render it to PDF with **WeasyPrint** or **Paged.js**, both free. Page breaks, running headers, page numbers, footnotes and a generated contents list are all CSS features now.

This is the right answer when the same content must become a web page *and* a PDF, when the content is generated from a database, or when you want the layout under version control. It is the wrong answer when somebody needs to nudge things by eye, because there is no canvas to nudge on.

The decision, in three questions

1. **Does it have to be press-grade colour with bleed?** If no, Writer or LaTeX are open to you. If yes, the shortlist is InDesign, Publisher, Scribus or CorelDRAW.
2. **Is it over about thirty pages with structure — contents, cross-references, an index?** If yes, drop CorelDRAW.
3. **Is there any budget?** If no, the answer is Scribus, and it is a real answer rather than a consolation.

The one combination to avoid is a word processor producing a press PDF for a job with bleed. It does not do bleed, and every workaround is worse than switching tools.

BEFORE YOU MOVE ON

A self-publishing author needs a 280-page novel interior: text only, no colour, no images, but with an automatic contents list and consistent running heads. They have no budget and a five-year-old laptop. What is the most defensible recommendation?

A. Scribus, because its PDF/X export and its colour management are by some way the strongest of any of the free tools.

B. LibreOffice Writer, because a mono text interior needs styles and running heads, not press colour.

C. HTML and CSS rendered with WeasyPrint, because the content can then also be published as a web page.

D. CorelDRAW, because it handles multi-page documents and is what most local print shops already run.

47 · 8 min

Deriving a grid from the type, not from a template

THE ONE THING TO KEEP

Derive the grid from the body text — size and leading first, then a measure of 45 to 75 characters, then columns, and the margins are what remains — and add the binding's gutter allowance to the inner margin before anything else is placed.

The four rectangles

Every page has them, and confusing them produces most layout problems.

- **Trim** — the finished page. A5 is 148 × 210 mm.
- **Bleed** — artwork extending past trim, normally 3 mm.
- **Margin** — where the text block starts.
- **Column** and **gutter** — the subdivisions of the text block and the gaps between them.

Margins are a design decision and there is no universal correct answer, but there is a useful starting point: **make the outer margin larger than the inner, and the bottom larger than the top**. The page then looks optically centred rather than sitting low, and the reader's thumb has somewhere to go.

A workable default for a book page: top 15 mm, outer 18 mm, bottom 20 mm, inner 20 mm. The inner is larger because binding eats some of it.

The gutter allowance, which depends on binding

On a bound book, part of the inner margin disappears into the spine. How much depends on how it is bound:

- **Saddle stitch** (stapled through the fold) — very little loss, but pages near the centre creep outwards, which the last lesson of this module covers.
- **Perfect binding** (glued spine, as on most paperbacks) — the page curves into the spine and roughly **3–6 mm** becomes hard to read. Add that to the inner margin.
- **Wire-o or spiral** — the punch holes need **8–12 mm** clear.
- **Case binding** (hardback) — similar to perfect binding plus the hinge.

Ask the binder before setting the margins. It is a one-line question and it is not recoverable afterwards.

Deriving the grid from the body text

Most people pick a grid from a template and then fit the type to it. That is backwards, and it is why so many layouts have awkward line lengths.

Do it the other way:

1. **Choose the body size and leading.** Say 10 pt on 14 pt.
2. **Set the measure from the type.** 45–75 characters. At 10 pt in a typical text face that is roughly **80–110 mm**. Take 95 mm.
3. **That is your column width.** Now decide how many columns fit the page, with gutters wide enough to keep columns separate — a gutter of about 5 mm, or one to two times the type size, is normal.
4. **The margins are what is left.** Adjust them to the proportions above, then let the column width shift a millimetre or two to make the arithmetic close.

The result is a grid where the text is comfortable to read, because the reading came first.

Columns and the modular grid

A single text column suits a novel. A magazine or a report usually wants more flexibility, and the standard technique is to divide the page into more columns than you use — a **12-column grid** — and then span them: text across 8 columns, a sidebar across 4, an image across all 12, a caption across 3.

Twelve is popular because it divides by 2, 3, 4 and 6, so you can make halves, thirds, quarters and sixths without a new grid. The same reasoning is why web layouts use it.

Add **horizontal** divisions too — a modular grid — if the document has repeated card-like elements. Set the row height to a multiple of the baseline leading so type still lands on the baseline grid.

Setting it up

- **InDesign**: File ▸ Document Setup for trim and bleed; Layout ▸ Margins and Columns for the rest; Layout ▸ Create Guides for a modular grid.
- **Scribus**: File ▸ Document Setup ▸ Document for size and margins, Page ▸ Manage Guides ▸ Column/Row for an automatic grid, and the Bleeds tab under Document Setup.
- **Affinity Publisher**: Document Setup, then View ▸ Guides Manager and the Columns section.

Set all of it on the master page, not on page one. A grid that exists only on the first spread is a suggestion.

The check

Print a spread at 100% and hold it at reading distance. If the text block looks low on the page, your bottom margin is too small. If your eye loses its place returning to the start of a line, the measure is too long. Both are faster to diagnose on paper than on screen, because on screen you cannot see the page edges honestly.

BEFORE YOU MOVE ON

A 200-page perfect-bound manual is set with equal 15 mm margins on all four sides. The printed book reads awkwardly and text near the spine is hard to see. Which two decisions caused this?

A. The margins were set on individual pages rather than on a master page, so they drifted through the whole document.
B. Equal margins make the text block sit optically low, and no allowance was made for the spine.
C. The measure is too long for the page size, so the return sweep loses its place near the binding edge.
D. Bleed was not set, so the trimming cut into the inner margin on every second page.

48 · 8 min

Masters, running heads and why folios drift

THE ONE THING TO KEEP

Put all repeating furniture on masters, number pages with a marker rather than by typing, use sections for front matter, and when a folio drifts the cause is almost always a local override on that page rather than a problem with the master.

What a master is

A master page is a template applied to real pages. Anything on it — margin guides, the running head, the folio, a rule, a background panel — appears on every page using that master, and changing it changes all of them.

You will normally need three or four:

- **A-Master** — the standard text page, with running head and folio.

- **B-Master** — the chapter opener, usually with no running head and a folio in a different position, or none.
- **C-Master** — full-bleed image pages, with no furniture at all.
- **Front matter** — title page, contents, imprint. Often no folio.

Masters can be based on other masters, exactly like paragraph styles. Build B based on A, delete the running head, and a later change to A's folio position still reaches B.

Automatic page numbers

Never type a page number. Insert a **current page number marker**, which renders as the master's prefix on the master and as the real number on the page.

- **InDesign**: Type ▸ Insert Special Character ▸ Markers ▸ Current Page Number (**Ctrl+Alt+Shift+N**).
- **Scribus**: Insert ▸ Character ▸ Page Number (**Ctrl+Alt+Shift+P**).
- **Affinity Publisher**: Text ▸ Insert ▸ Fields ▸ Page Number.
- **LibreOffice Writer**: Insert ▸ Field ▸ Page Number, inside a header.

Insert one page and add a **next/previous page number** marker too if the document has continued articles.

Sections, which is what people miss

A book's front matter is numbered i, ii, iii and the body restarts at 1. That is two **sections**, and every layout application supports them.

- **InDesign**: Pages panel ▸ select the first page of the section ▸ panel menu ▸ Numbering & Section Options. Set the style (Roman, Arabic) and the starting number.
- **Scribus**: File ▸ Document Setup ▸ Sections. Add a section, set its page range, style and start.

Without sections, people fake it by typing numbers, and every insertion afterwards breaks them silently.

Why folios drift

The commonest bug in long documents: page numbers or running heads that are 3 mm out on some pages and not others.

Three causes, in order of frequency:

1. **The master item was overridden.** Somebody clicked the folio on a page and nudged it. Most applications require a modifier — **Ctrl+Shift+click** in InDesign — to select a master item on a page, precisely so this is hard to do by accident. When it happens, the item is now local and no longer follows the master. InDesign's Pages panel menu has *Remove All Local Overrides*; use it and the folios snap back.
2. **The wrong master is applied.** A page using B-Master where A was intended. The Pages panel shows the master letter on each page thumbnail; scan down the column.
3. **Two masters with the folio in slightly different places.** Fix it by basing one on the other rather than building them separately.

Running heads that know their chapter

A running head saying "Chapter 4: Materials" on every page of chapter 4, changing by itself at chapter 5, is a **text variable** driven by a paragraph style.

- **InDesign**: Type ▸ Text Variables ▸ Define ▸ New ▸ Running Header, set to the chapter-title paragraph style. Insert the variable on the master.
- **LibreOffice Writer**: Insert ▸ Field ▸ More Fields ▸ Document ▸ Chapter.
- **Scribus** has no equivalent; running heads are set per section by hand or by a script.

This is one of the strongest arguments for InDesign on a long structured document, and it is worth knowing it exists even if you are working in something else, because it tells you what to budget time for.

Left and right are not the same page

A facing-pages document has recto (right-hand, odd) and verso (left-hand, even) pages, and they are mirror images: the folio sits on the outer edge on both, which means right on a recto and left on a verso. Set the master as a spread, not as a single page, and set the margins as *inside* and *outside* rather than left and right. Every application offers this and it is a checkbox at document creation that people miss.

Page 1 is always a recto. If your document opens on a left-hand page, the page count or the start position is wrong.

BEFORE YOU MOVE ON

In a 90-page catalogue, the folio sits 4 mm higher on eleven scattered pages. The master page is correct and all eleven pages use it. What is the most likely cause, and the fastest fix?

A. The folio is a text variable that resolves to a different string length on those particular pages, shifting the baseline.

B. Those pages sit in a second section with a different numbering style, which repositions the marker.

C. Master items on those pages were overridden locally; removing local overrides restores them.

D. The pages use a master based on the correct one, and the derived master has its own folio position.

49 · 8 min

Threaded text, overset, and copy that does not fit

THE ONE THING TO KEEP

Threaded frames let text reflow through a document, and overset text is the silent failure that hides surplus copy where no proof will show it — so check the preflight for overset before every export, and fit copy by editing it or by changing a style rather than by distorting type.

The feature that defines a layout application

In a drawing program, every text box is an island. In a layout application, frames are **threaded**: text flows from one into the next, across columns and across pages, and inserting a paragraph on page 3 reflows everything after it automatically.

This is the single feature that makes long documents possible, and it is why CorelDRAW becomes painful past twenty pages.

Threading, unthreading, and seeing the thread

- **InDesign**: click the out-port (the small box at the bottom right of a frame), then click the next frame — or click an empty area to draw a new one. View ▸ Extras ▸ Show Text Threads (**Ctrl+Alt+Y**) draws the links. Shift-click the out-port with a loaded cursor to **autoflow**, creating pages as needed until the story ends.
- **Scribus**: select the first frame, click the link-text-frames button (or press **N**), then click the target. The unlink button breaks it.
- **Affinity Publisher**: the same out-port model as InDesign, with *Auto-flow* on the Text Frame panel.

Threading is directional. Moving a frame does not change its position in the thread, which is why a frame on page 12 can be the eighth in a story that started on page 3.

Overset text

When a story is longer than the frames holding it, the surplus is **overset**: still in the document, not displayed anywhere.

This is the most dangerous failure in layout, because it is silent on a printed proof. The paragraph simply is not there, and nobody notices until a reader does.

Every application marks it:

- **InDesign**: a small red + at the frame's out-port, and a count in the Preflight panel.
- **Scribus**: a red cross in the bottom-right corner of the frame, plus an entry in the Preflight Verifier.
- **Affinity Publisher**: a red triangle on the frame edge.

Check for overset before every export, every time. It costs two seconds and it is the reason the preflight panel exists.

Copy fitting is a design decision

When copy will not fit, there are five moves and they are not equally honest.

1. **Edit the copy.** Cut forty words. Almost always the best answer, and the one nobody wants to make.
2. **Add a page.** Cheap in digital printing, and in a bound job the count must be a multiple of four, so it may mean four pages.
3. **Change the style globally** — 9.5/13 instead of 10/14 throughout. Legitimate, provided it is applied through the style so the whole document stays consistent.
4. **Track the offending paragraph** by −5 or −10 thousandths of an em. Acceptable in small doses; visible as a change in colour if overdone.

5. **Scale the type horizontally.** Never. It distorts the letterforms and is visible beside any undistorted paragraph on the same spread.

InDesign's *Story* panel and the Keep Options in a paragraph style let you automate some of this. Whatever you do, do it through a style, not by selecting text — otherwise you have created a local override, and the earlier lesson explained where that ends.

Smart text reflow and the pages that vanish

InDesign's *Smart Text Reflow* (Preferences ▸ Type) adds and deletes pages automatically as a story grows or shrinks. It is useful for a long text-only document and actively dangerous in a designed one, because it can delete pages you placed images on.

If you use it, set *Limit to Primary Text Frames* and *Preserve Facing-Page Spreads*. If your document is designed rather than flowed, switch it off.

Placing text without losing control

Use **Place** (**Ctrl+D**) rather than copy-paste for any substantial copy. Placing lets you keep or discard the source formatting, map Word styles onto your paragraph styles, and re-link to the original file.

Mapping Word styles onto InDesign styles is the single biggest time saving available on a long editorial job. Show Import Options on Place, then *Customise Style Import*, and map "Heading 2" to your "B-Head" once. Every subsequent import is instant.

Before any of that, run a clean-up pass on the source text: find-and-replace double spaces, empty paragraphs, manual line breaks and tabs used as indents. Five find-and-replaces on the raw text will save you an hour of fighting the layout later.

BEFORE YOU MOVE ON

A 48-page brochure is proofed, approved and printed. A reader points out that a paragraph referenced in the contents list does not appear anywhere in the book. The file opens with no error and the text is still in it. What happened?

A. Keep Options on its style pushed it past the last frame in the thread, leaving it overset.
B. The paragraph was placed on a master page that was never applied to any document page at all, so it never rendered.
C. The frame containing it was set to non-printing, which hides it from export but keeps it in the file.
D. Smart Text Reflow deleted the page during export because the story had shrunk since it was last edited.

50 · 8 min

Linked images, and the number that actually matters

THE ONE THING TO KEEP

Effective PPI — native pixels divided by the placed physical size — is the only resolution figure that means anything, a stored dpi tag is just a note about intended size, and packaging exists because a linked file that was not sent is the commonest reason a job cannot be opened.

Link or embed

A **linked** image stays on disk; the layout holds a path and a low-resolution preview. A **embedded** image is copied inside the document file.

Link, nearly always:

- The document stays small and opens fast.

- Updating the photograph in Photoshop or GIMP updates the layout.
- The press's RIP receives the full-resolution file through the PDF export, not a re-compressed copy.

Embed only for a tiny document you will send as a single file, and even then, package instead.

The cost of linking is that a link can break. Move the folder, rename a file, or send only the layout document, and the links go missing. That is what packaging exists for.

Effective PPI, which is the number to check

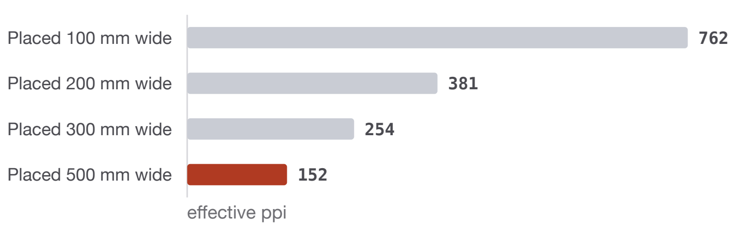
A photograph has a **native** resolution: 3000 × 2000 pixels. It has no physical size until you place it.

Place it and scale it to 100 mm wide. Its **effective PPI** is:

```
3000 pixels ÷ (100 mm ÷ 25.4) = 3000 ÷ 3.937 = 762 ppi
```

Plenty. Scale it up to 500 mm wide and the effective PPI drops to 152, which is marginal for good commercial print.

One 3,000-pixel photograph, placed four ways



The pixels never change. Only the physical size you placed them at does, which is why a file's stored dpi tag is a note about intention rather than a property of the image. Sort the Links panel by effective ppi and read down the column.

Effective PPI is the only resolution figure that means anything. A file's stored "300 dpi" tag is a note about intended size, not a property of the pixels; scaling in the layout is what decides the real number.

Where to read it:

- **InDesign**: the Links panel, expanded, shows *Actual PPI* and *Effective PPI* for the selected image. Sort the panel by Effective PPI and scan the column.
- **Scribus**: the Image tab of the Properties palette shows the DPI; the Preflight Verifier flags images below a threshold you set.
- **Affinity Publisher**: the Resource Manager lists placed images with their effective resolution.

The target, and the exceptions

The general rule is about **twice the line screen**: 300 ppi for 150 lpi commercial print. The exceptions matter as much as the rule:

- **Line art and scanned text** — 1200 ppi bitmap, not 300. A 300 ppi scan of a fine line drawing has visibly ragged edges because there is no halftone to hide them.
- **Newsprint at 85 lpi** — 170 ppi is fine.
- **Large format at viewing distance** — a hoarding seen from 30 m is normal at 20–40 ppi at full size.
- **Digital toner presses** — often happy with 240 ppi, because there is no plate.

Ask, and if nobody answers, 300 is safe and 200 is usually survivable.

Why upsampling does not help

Resampling a 150 ppi image to 300 ppi in an editor adds pixels by interpolation. The detail is not there to be recovered; you have made a larger file with the same information and softer edges.

There is one nuance worth knowing: modern machine-learning upscalers genuinely invent plausible detail, and the result can look better than the original enlarged. It is invented detail, which is fine for a decorative background and not fine for a product photograph, a face or anything where the invented detail might be wrong. Say which you have used.

Package, collect, and the file that arrives incomplete

Before sending anything, use the application's collect-for-output function. It copies the document, every linked image, and every font into one folder.

- **InDesign**: File ▸ Package.
- **Scribus**: File ▸ Collect for Output.
- **Affinity Publisher**: File ▸ Package.

Send that folder, or the PDF you exported from it. Sending only the layout file is the single most common cause of "we cannot open your job" emails.

Why the file is 400 MB

Two reasons, and both are fixable.

Cropped images are still whole. A 40 MB photograph cropped to a thumbnail carries all 40 MB into the PDF unless you crop image data on export. Most export dialogs have the option.

Nothing is downsampled. A 762 ppi image does not print better than a 300 ppi one; it just takes longer. Set the export to downsample colour images above about 450 ppi to 300 ppi, using bicubic. The visual result is identical and the file is a fraction of the size.

Check the exported PDF, not the layout

The layout tells you what it intended. The PDF tells you what the press will receive, and the two can differ if an export preset downsampled more aggressively than you expected.

```
pdfimages -list brochure.pdf
```

That lists every image with its pixel dimensions and, in most builds, its effective x-ppi and y-ppi on the page. Scan the ppi columns for anything under your target, and scan the pixel dimensions for anything enormous, which tells you a crop was a clip. Two seconds, no software to install beyond poppler-utils, and it catches both failures in this lesson at once.

BEFORE YOU MOVE ON

A 4000 × 3000 pixel photograph tagged "300 dpi" is placed in a layout and scaled to fill a 420 mm wide A3 spread. Is it adequate for 150 lpi commercial print?

A. Yes — the file is tagged 300 dpi, which is the standard requirement for commercial print.
B. No — 4000 pixels across 420 mm is about 242 effective ppi, which is below but close to the 300 ppi target.
C. Yes — 4000 pixels exceeds the 3543 pixels an A3 width needs at 300 ppi, so there is headroom.
D. No — the image must be resampled up to 300 ppi in an editor before placing, which will restore the missing detail.

---

51 · 8 min

Tables, and why tabs are not one

THE ONE THING TO KEEP

A real table object supports repeating headers, cell styles, alternating fills and central restyling where tabbed text supports none of them — and a table is made readable by removing rules and adding space rather than by boxing every cell.

Tabs are not a table

A great many price lists, timetables and specification sheets are built by typing text and pressing Tab. It looks like a table and it behaves like nothing.

What goes wrong:

- A cell whose text is longer than its column pushes everything after it out of line, with no warning.
- A row cannot have a background fill, a border, or vertical alignment.
- Adding a column means retyping every row.
- The whole thing breaks when the typeface changes, which is the exact scenario the styles lesson was about.

A real table object fixes all four, and every layout application has one.

Building one properly

- **InDesign**: Table ▸ Insert Table, or select text with tabs and Table ▸ Convert Text to Table, choosing the column and row separators. Then Table Styles and Cell Styles, which work exactly like paragraph styles: define once, apply everywhere, change centrally.
- **Affinity Publisher**: the Table tool, with Table Formats and Text Styles.
- **Scribus**: Insert ▸ Table. Scribus's tables are usable but noticeably less developed than InDesign's — cell styles arrived late and merging behaviour is fiddly. For a complex table in Scribus, many people build it in LibreOffice Calc, export a PDF, and place that.
- **LibreOffice Writer**: real tables with styles, and genuinely good at long data tables that run over pages.

The features worth knowing exist

Repeating header rows. A table spanning four pages should carry its column headings onto each. In InDesign: select the header row, Table ▸ Convert Rows ▸ To Header. Doing this by hand means the headings are wrong the moment a row is inserted.

Cell styles inside a table style. The table style names which cell style applies to the header row, the body rows, the first column and so on. One definition for the whole document's tables.

Alternating row fills. Table Options ▸ Alternating Fills. Set it in the style, never by colouring rows individually — inserting a row then breaks the alternation everywhere below.

Tabular figures. Covered in the type module, and this is where it matters: a column of numbers in proportional figures will not align however you set the tabs or the cell alignment, because the digits themselves are different widths.

Decimal-aligned tabs inside cells. For a price column, set a right-aligned or decimal tab inside the cell rather than right-aligning the whole cell, so `1,250.00` and `95.50` line up on the point.

Designing a table people can read

The default table — a box around every cell — is the hardest to read, because the rules compete with the data.

- **Delete most of the rules.** A line under the header and a line at the foot is usually enough. Vertical rules are almost never needed; the space between columns does that job.
- **Use space, not lines.** Increase the cell inset before you add a border.
- **Align numbers right, text left, headers with their data.** A centred column of numbers is unreadable.
- **Do not centre headers over left-aligned text.** Align the header the same way as the column beneath it.
- **A light alternating tint** at 5–8% beats ruled rows for a long table, and remember from the colour module that 5% on uncoated stock may print at 12%.

The free path, end to end

For a specification sheet or a price list with no budget:

1. Keep the data in a spreadsheet — LibreOffice Calc or a CSV.
2. Either place the exported PDF into Scribus as a picture, or use Scribus's table and paste the data column by column.
3. For anything genuinely complex, generate it as HTML with CSS and render with WeasyPrint. Tables are what HTML was invented for, and CSS

gives you repeating headers (`<thead>`), page-break control and alternating rows for nothing.

That third route sounds like more work and is less work as soon as the table has more than about forty rows or changes more than once.

The habit

Keep the data in one place and pull it into the layout, rather than retyping it. Whenever a table exists in both a spreadsheet and a layout, they will disagree within a month, and the version that goes to press will be the wrong one.

BEFORE YOU MOVE ON

A 12-page price list is built with tab stops. The client asks to change the body typeface, and afterwards several columns no longer line up although every tab stop is unchanged. What is the underlying reason a table object would not have this problem?

A. Table objects store their column positions in absolute units, while tab stops are always stored relative to the current type size.

B. A table object applies tabular figures automatically, so numbers always align regardless of typeface.

C. A table's columns own their width and wrap long content; tabbed text pushes past the next stop.

D. Tab stops are a character-level property and are lost when a paragraph style changes typeface.

52 · 8 min

Contents lists and cross-references that renumber themselves

THE ONE THING TO KEEP

Generated contents lists, cross-references, numbered captions and running heads all derive from paragraph styles and update when pages move — so anything that refers to a position in the document should be generated rather than typed, including the leader dots.

Never type a page number into a contents list

The contents list is the first place a long document goes wrong, because it is built early, the document shifts, and nobody rechecks it. A generated contents list cannot be wrong, because it is derived from the pages themselves.

How a generated contents works

It reads paragraph styles. Every paragraph carrying "Chapter Title" becomes a level-1 entry; every "A-Head" becomes level 2; the page number is read from where the paragraph actually is.

Which means two things follow. **The styles have to be applied properly** — a heading formatted by hand rather than by style will be missing from the contents, and that is the commonest failure. And **you have to regenerate** after the pages move.

- **InDesign**: Layout ▸ Table of Contents. Choose which paragraph styles map to which levels, assign a paragraph style to each level's entry, and tick *Include Book Documents* for a multi-file book. Update with Layout ▸ Update Table of Contents.
- **Scribus**: Insert a text frame, give it a name, then File ▸ Document Setup ▸ Table of Contents and Indexes, and set which attribute to collect. Generate

from Extras ▸ Generate Table of Contents. Scribus's implementation is more limited and works on an attribute you attach to frames.

- **LibreOffice Writer**: Insert ▸ Table of Contents and Index. Update by right-clicking it.
- **LaTeX**: `\tableofcontents` and two compilation passes.

Leader dots done properly

The row of dots between an entry and its page number is a **tab leader**, not typed full stops.

Set a right-aligned tab at the right edge of the column, and in the tab dialog set the *Leader* character to a full stop, often with a space after it so the dots are not crowded: type `. ` as the leader string. Then the dots regenerate at the right length whenever an entry changes.

Typed dots are the giveaway of an amateur document, and they are also wrong the moment a chapter title gets longer.

Cross-references

"See figure 12 on page 47" is a reference that will be wrong within a week unless it is live.

InDesign: Type ▸ Hyperlinks & Cross-References ▸ Insert Cross-Reference. Point it at a paragraph (usually a figure caption or a heading) and choose a format — "Full Paragraph", "Page Number Only", or a custom build. When the figure moves, the reference updates.

The Cross-References panel flags any reference whose target has been deleted, which is a genuine safety net on a technical manual.

LaTeX: `\label{fig:pump}` and `\ref{fig:pump}` / `\pageref{fig:pump}`, and this is where LaTeX is unbeatable — a document with four hundred cross-references is no harder than one with four.

LibreOffice Writer: Insert ▸ Cross-reference, with reference targets set on headings, bookmarks or numbered items.

Scribus has no cross-reference system. On a document that needs them, this is a real reason to choose something else.

Numbered figures, tables and equations

Set the caption paragraph style to use an **automatic numbered list**. Insert a figure in the middle and everything after it renumbers, and every cross-reference follows.

InDesign: define a numbered list under Type ▸ Bulleted & Numbered Lists ▸ Define Lists, then attach it to the caption style with the format `Figure ^#:`. LibreOffice: Insert ▸ Caption, which does this for you and creates the reference target at the same time.

Indexing, briefly

An index is built from **index markers** placed in the text, not from a search. You mark a term, optionally with a subentry, and the application collects the markers and their page numbers.

It is slow, careful work — a 300-page book's index is one to three days — and it is genuinely valuable in a reference book. InDesign and LibreOffice both support it; Scribus's support is minimal; LaTeX has `makeindex`.

If the budget does not allow an index, say so rather than producing a bad one. A concordance of every occurrence of every word is worse than no index, because it wastes the reader's time convincingly.

The rule

Anything in a document that refers to a position in the document — a page number, a figure number, a chapter name in a running head — should be generated. If you are typing it, you are creating something that will become untrue without telling anybody.

BEFORE YOU MOVE ON

A generated contents list is produced for a 120-page report. Three second-level headings are missing from it, although they look identical to the ones that appear. What is almost certainly true of those three?

A. They fall on pages using a different master, which the contents generator excludes by default.
B. They were formatted by hand to look like the heading style rather than having it applied.
C. They sit inside anchored frames, and text in anchored objects is not collected.
D. The contents list was generated before those three headings were written, and regenerating is the only fix.

53 · 9 min

Five hundred different cards from one template

THE ONE THING TO KEEP

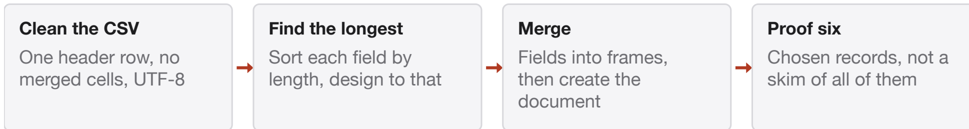
Variable data printing is a template plus a clean UTF-8 CSV, and nearly every failure is a data problem — so design to the longest record, generate QR and barcode images beforehand as a merged image field, and proof by sampling six deliberately chosen records rather than skimming all of them.

The work nobody advertises

Five hundred visiting cards with different names. Three hundred certificates. Price labels for every item in a shop. Name badges for a conference. Numbered raffle tickets. Barcode or QR labels for a warehouse.

This is **variable data printing**, it is well paid, it is repeatable, and remarkably few designers offer it. The whole skill is setting up a template once and pointing it at a spreadsheet.

Five hundred cards, four steps



Nearly every merge failure is a data failure, which is why two of the four steps happen in the spreadsheet. Nobody checks 500 records, so the proof is six chosen ones: first, last, longest, shortest, one with an odd character, one at random.

Prepare the data first

Most merge failures are data failures, so clean the CSV before you open the layout.

1. **One header row**, with short names and no spaces: `name`, `designation`, `mobile`, `email`, `photo`.
2. **No merged cells, no blank rows, no totals row** at the bottom.
3. **Consistent case and spacing.** Run a trim on every field; trailing spaces produce mysterious alignment.
4. **Encoding: UTF-8.** If the data has any Indic script, a name with an accent, or a rupee sign, a file saved as Latin-1 will arrive as mojibake. In LibreOffice, Save As ▸ Text CSV ▸ tick *Edit filter settings* and choose UTF-8.
5. **Image fields** hold a file path, and InDesign requires the header to begin with `@` — `@photo` — with the paths either absolute or relative to the data file.
6. **Look at the longest record.** Sort by the length of each field and design the template around the longest name, not the first one.

Doing it in each program

InDesign — Window ▸ Utilities ▸ Data Merge. Select the data source, place the field placeholders into the frames, then *Create Merged Document*. Options that matter: *Records per Document Page* (one card per page, or a grid of ten on an A4 sheet), *Multiple Record Layout* with spacing, and **Preview** before merging, which is how you catch overflow.

CorelDRAW — File ▸ Print Merge. It is genuinely good, it handles a grid of cards on a sheet, and it is why so many Indian print shops do this work in CorelDRAW rather than anything else.

Scribus — the **ScribusGenerator** Python script, free on GitHub. Put `%VAR_name%` placeholders in the text, run the script against a CSV, and it produces one output file per record or a combined document. Less polished than the commercial tools and entirely capable.

Free, no layout application at all — generate HTML from the CSV with a short script and render with WeasyPrint. For a thousand labels this is faster than any GUI, and the template is a text file you can version.

Overflow is the failure mode

A template that fits "Anil Kumar" will not fit "Lakshmi Narayanan Venkataraman". The merged document will either overset that record silently or push text outside the frame.

Three defences:

- Design to the longest record, found by sorting the data.
- Set the frame to **auto-size** where the application supports it, or set a minimum type size and allow the style to shrink by a defined amount.
- **Check the merged output**, not the template. Which is the next section.

QR codes and barcodes per record

A QR code per row is the most common request after names.

- **InDesign** has a built-in QR code generator (Object ▸ Generate QR Code) but it does not read from a data field, so the practical route is to generate the images first and merge them as an `@image` field.
- **Generate them free** with the Python `qrcode` library or `qrencode` on the command line, writing one SVG per row named after a key in the data:

```
while IFS=, read -r id url; do qrencode -t SVG -o "qr/$id.svg"
"$url"; done < links.csv
```

- **Barcodes** — `zint` is free, handles EAN, Code 128, DataMatrix and the rest, and outputs vector. Get the check digit right; a barcode with a wrong check digit scans as nothing.
- Always **test a printed sample with a real scanner** before running 5,000. Quiet zone, contrast and minimum module size all matter, and a QR code printed too small or over a pattern will not read.

Proof the merge by sampling

Nobody checks 500 merged records. Check these six:

1. The first record.
2. The last record.
3. The longest name.
4. The shortest name.
5. Any record with an unusual character — an apostrophe, a hyphen, a non-Latin script.
6. One at random from the middle.

If all six are right, the merge is almost certainly right. If any is wrong, the fault is systematic and you have found it before printing.

Pricing it

Charge for the setup, not for the records. The template is two to four hours; the merge itself takes minutes. A client comparing you against "someone who types 500 cards" is comparing against three days of labour, and the honest pitch is that you are faster, cheaper and will not make a typo on card 312.

BEFORE YOU MOVE ON

A merge of 500 name badges produces correct output except that about thirty records show their names as strings of question marks and boxes. The template is fine and those thirty names are all in Devanagari. What is the most likely cause?

A. The Devanagari text needs a shaping engine the merge process bypasses, so conjuncts fail to form.
B. The CSV was saved in a single-byte encoding, so the Devanagari code points were lost before the layout saw them.
C. The template's font has no Devanagari glyphs, so every character falls back to the missing-glyph box.
D. The image field paths for those records were absolute rather than relative, so the merge substituted placeholder characters.

54 · 8 min

Page counts, creep and the spine

THE ONE THING TO KEEP

Folded work comes in multiples of four and often of sixteen, creep pushes the inner pages of a thick saddle-stitched booklet outward by several millimetres, and a perfect-bound cover cannot be laid out until the printer gives you the spine width measured on the actual stock.

Why the page count is a multiple of four

A book is not printed page by page. Sheets are printed with several pages on each side, folded, and trimmed. Fold one sheet once and you get four pages — two on each side. So **every folded job has a page count that is a multiple of four.**

Larger presses print 8, 16 or 32 pages per sheet, and a 16-page folded sheet is called a **signature** or a **section**. A perfect-bound book is a stack of signatures glued at the spine.

The practical consequences:

- A 62-page manuscript becomes a 64-page book, with two blank pages, or it becomes 60 by cutting content.
- If the press runs 16-page signatures, going from 64 to 68 pages may cost a whole extra signature, not four pages' worth of paper. **Ask what the signature size is** before promising a page count.
- Blank pages at the back are normal and are not a mistake. They can carry "Notes" if that suits the book.

Imposition is not your job

Imposition is arranging the pages on the sheet so they come out in order after folding. For a simple saddle-stitched 16-page booklet, page 1 sits beside page 16 on one side of a sheet.

Do not do this yourself. Supply **single pages, at trim size, in reading order, with bleed and crop marks**. The press imposes for their sheet size, their folding machine and their binding, and a pre-imposed file has to be taken apart again before it can be used.

The only exception is if the printer explicitly asks for printer's spreads, which small shops occasionally do. Ask; do not assume either way.

Creep, and the booklet that loses its margins

On a saddle-stitched booklet, the inner sheets are wrapped around by the outer ones. Each sheet's fold sits slightly further out, so the inner pages project further and are trimmed more at the fore-edge.

That shift is **creep** (or push-out). On a 64-page saddle-stitched booklet on normal paper it can be 3–5 mm at the centre spread.

Two consequences:

- **Keep important content well inside the fore-edge margin** on a thick saddle-stitched job — page numbers especially.
- The press usually compensates by shifting pages inward progressively during imposition. Ask whether they do. If they do, you do nothing. If they do not, you widen the margins.

Creep is why thick booklets are usually perfect-bound instead: beyond about 64 pages of normal stock, saddle stitching stops working well.

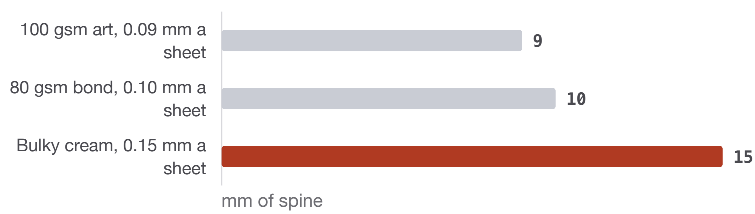
Spine width

For a perfect-bound book, the cover is one piece: back cover, spine, front cover. You need the spine width before you can lay out the cover, and it depends on the paper's **bulk**, not only its weight.

```
spine (mm) = (pages ÷ 2) × caliper (mm per sheet)
```

Caliper depends on the stock. A rough guide: 80 gsm uncoated bond is about 0.10 mm per sheet, 100 gsm art paper about 0.09 mm, and a bulky cream book paper can be 0.15 mm at the same weight — which is why novels are thicker than their page count suggests.

Spine width of a 200-page book, by stock



Half the page count multiplied by the caliper of one sheet. Weight is not thickness, which is why a bulky cream book paper at the same grammage makes a spine half as wide again and why the number has to come from the printer on the actual paper.

Get the number from the printer, on the actual paper. They will measure it or quote the mill's figure. A cover laid out to a guessed spine width will have the fold falling in the wrong place, and a fold 2 mm out is visible on every copy.

Allow for the cover's own thickness and the hinge on a hardback, which the binder will specify.

The binding methods, and what each demands of you

- **Saddle stitch** — stapled through the fold. Multiples of 4, up to about 64 pages, cheap, lies reasonably flat, subject to creep.
- **Perfect binding** — glued spine. Multiples of 4 (or of the signature), from about 40 pages, needs 3–6 mm extra inner margin, does not lie flat.
- **Wire-o and spiral** — punched and coiled. Any page count, lies completely flat, needs 8–12 mm clear at the binding edge and the holes destroy any artwork crossing it.
- **Section sewn / case bound** — hardback. Expensive, durable, needs endpapers and a case specification from the binder.

The conversation to have before you start

Four questions, one email:

> *Binding method? Signature size? Spine caliper for this stock at this page count? Do you compensate for creep during imposition?*

Everything in this lesson is in those four answers, and they take a printer under a minute to give.

BEFORE YOU MOVE ON

A designer lays out a perfect-bound book cover using a spine width calculated from the page count and the paper's gsm figure. The printed covers have the spine fold 3 mm off centre on every copy. What was the error?

A. Perfect-bound spines require an allowance for the glue layer, which is a fixed 3 mm on any book.
B. The page count should have been divided by the signature size rather than by two before multiplying by the caliper.
C. Weight in gsm does not determine thickness; bulk varies by stock, so the caliper must come from the printer.
D. The cover was supplied as three single pages rather than as one continuous spread, so the press positioned the spine by estimate.

CASE STUDY · MODULE 6

Fourteen hundred certificates and a question mark in every name

A polytechnic in Nagpur: convocation on Saturday, a digital press booked for ten on Friday morning, and a merge run at eleven at night.

The polytechnic redesigned its convocation certificate because the principal wanted each student's name printed in Marathi as well as in English. Fourteen hundred and six students, one certificate each, personalised.

The designer built the template properly. One page, at trim size, with the student name, the Marathi name, the programme, the roll number and the year as merge fields. The registrar's office exported the data from the student information system to a spreadsheet, and the clerk sent it across on Thursday evening with an apology for the delay.

She ran the merge at eleven that night, which was already later than she wanted, and previewed before creating the merged document.

Record one was fine. Record two was fine. From record three onwards, every Marathi name was a row of question marks.

The cause took four minutes to find and could not be fixed in the layout. The student system had exported the file in a legacy single-byte encoding rather than UTF-8, so every Devanagari character had been replaced on the way out. The English names were intact because they are ASCII. There was nothing in the layout file to repair; the data had lost the characters before it arrived.

While she was in there she found a second fault. Three names overflowed the frame, one of them badly, because she had built the template around a record she had picked at random from the top of the file rather than around the longest one.

The digital press was booked for ten the next morning. The convocation was on Saturday, the Governor was attending, and the certificates are handed over on stage.

Three options, at eleven at night.

Drop the Marathi line. Print English only. This can be done immediately and the registrar could authorise it by phone. It also removes the one thing the principal asked for, on the day it was meant to appear, in front of the Governor.

Get the data re-exported as UTF-8. Somebody with access to the student system has to do it. The registrar's office opens at ten on Friday, which is when the press slot starts. Asking the press to slip a day puts the printing into Saturday morning, which is the convocation.

Retype fourteen hundred Devanagari names from the original register overnight. Nobody seriously proposed this, and it is worth noticing why: it would introduce fourteen hundred fresh opportunities for a name to be wrong on a document a person keeps for life.

The press slot mattered more than it looks. The polytechnic had booked a four-hour window on a digital press that was running an election job either side of it, and the shop had made it clear that a slot given up was a slot gone. Digital printing has no plates, so a short notice change is cheap in materials and expensive in scheduling, and the scheduling was the whole constraint.

Dropping the Marathi line was also not as simple as it sounded. The certificate had been redesigned around two names, and the English name alone would sit in the middle of a space designed for two lines, which anybody on stage would read as a mistake rather than as a decision. Fixing the layout for a single line was another half hour she did not have, on top of a change the principal had not agreed to.

The clerk who had sent the file had gone home at half past five. The designer had his mobile number, because he had rung her earlier in the week about the column headings, and she had saved it.

STOP HERE

Before turning the page: what would you have done, and what would it have cost if you were wrong? Write it down. The point of the next section is to compare.

WHAT HAPPENED

She rang him at twenty past eleven and he answered.

He still had the original export on his office machine, but he also had a copy he had mailed to himself. His son's laptop had LibreOffice on it. She talked him through opening the file, then Save As, Text CSV, ticking Edit filter settings, and choosing UTF-8 in the character set dropdown. Twenty-five minutes end to end, most of it spent finding the file.

The re-merge took four minutes.

Then she did not skim fourteen hundred pages. She proofed six records, chosen rather than sampled: the first, the last, the longest Marathi name found by sorting the column by length, the shortest, one containing a nukta, and one at random from the middle. Two of the six were wrong. The longest name still overflowed, which she fixed in the paragraph style by allowing a defined one-step reduction rather than by scaling the type, and the nukta record showed a stray trailing space that was pushing the name off centre, which she cleared with a trim across the whole column.

Both faults were systematic, which is the point of choosing those six. Having fixed them, she had reasonable grounds to believe the other 1,400 were right.

The press ran at ten. The certificates were trimmed by four on Friday afternoon.

She now sends one line with every merge quotation: data must arrive as UTF-8, and I will ask for a sample of twenty records before the template is built.

Why proof six chosen records rather than skim all fourteen hundred?

A merge fault is almost always systematic rather than random, because every record goes through the same template and the same data pipeline. The six are chosen to hit the places faults live: the ends of the file, the extremes of field length, and an unusual character. Skimming a thousand near-identical pages reliably finds nothing, because attention fails long before the pages do, and it takes far longer.

What should have been settled before the template was built, and what would it have cost?

Two things, each worth a single line in an email. The encoding, stated as a requirement to the registrar's office before they exported anything. And the longest record, found by sorting each field by length, so the template was designed around it instead of around whichever name happened to be at the top. Together they are perhaps twenty minutes at the start of the job, against a telephone call at twenty past eleven at night.

One name still overflowed after the data was fixed. Which ways of making it fit are legitimate, and which is not?

Editing the copy is not available, because it is somebody's name. Changing the style globally is legitimate, as is allowing the style a defined reduction, as is enlarging the frame if the design permits. Tracking slightly is acceptable in small doses. Scaling the type horizontally is not, because it distorts the letterforms and is visible beside any undistorted certificate. Whatever is done must be done through the style, or it becomes a local override that the next change will not reach.

MODULE 6 · TEST

Check what stuck

6 questions, one right answer each. This one is for you, not for a record: answers and the reason behind each are at the back. If two questions from the same module go wrong, re-read that module before the exam rather than after.

1 You change a paragraph style and thirty paragraphs in the middle of the document do not update. What is the cause?

A. The style is based on another style, which takes precedence over it
B. Those paragraphs carry local formatting applied on top of the style
C. They sit in threaded frames, which each hold their own copy of the style
D. The style was edited on the master page rather than in the document

2 In a grid derived from the body text rather than from a template, what sets the column width?

A. The page size divided by the number of columns you want
B. A measure of 45 to 75 characters at the chosen size
C. The baseline grid, which is set to the body leading
D. The binding method, which sets the gutter allowance

3 A 3,000 pixel wide photograph is placed 500 mm wide. What is its effective ppi, and what does the file's stored dpi tag tell you?

A. About 152 ppi, and the stored tag is only a note about intended size
B. 300 ppi, because that is what the file itself was saved at
C. About 762 ppi, since effective ppi is read at the native size
D. It depends on the export preset, which is what sets the real resolution

4 Why is overset text described as the most dangerous failure in a layout?

A. It reflows the rest of the document without any warning
B. It is silent: the copy is in the file and on no page
C. It prevents the PDF from exporting until it is resolved
D. It pushes the page count past the next multiple of four

5 Why is a generated contents list preferred to typing the page numbers?

A. It formats the row of leader dots automatically at exactly the right length
B. It renumbers the pages themselves whenever a chapter moves
C. It derives its entries and pages from the styles, so it cannot go stale
D. It is required before a PDF can carry navigation bookmarks

6 Why can a 62-page booklet not simply be printed as 62 pages, and what sets the spine width of a perfect-bound book?

A. Folding gives four pages a sheet, and the spine comes from the caliper
B. Presses run 60 or 64 only, and the spine comes from the paper's gsm
C. Bleed adds two pages, and the spine comes from the cover's weight
D. Creep adds pages at the centre spread, and the spine comes from the trim size

MODULE 7

The handover

Everything between "the design is finished" and "the job is on press". What a RIP does with your file, which PDF/X flavour forbids what, the five rectangles inside a PDF, how to preflight your own export with free command-line tools, how to read separations, what to ask about paper and finishing, and how to run a proof approval so that a dispute has an answer.

> BY THE END OF THIS MODULE YOU CAN
>
> Export a file a press will run without a phone call — bleed and boxes correct, the right PDF/X flavour, fonts and images accounted for — and verify every one of those claims from the exported PDF itself using free tools, then specify the stock and finishing and record the approval.

The blue you chose on screen cannot be printed

THE ONE THING TO KEEP

Choose colours from a printed chart rather than a screen, keep small black text at 100K only, and run overprint preview before every export because a white object set to overprint vanishes entirely.

Why the client says the colour is wrong

Your screen makes colour by emitting red, green and blue light and adding them. Paper makes colour by absorbing it: cyan, magenta, yellow and black inks each subtract part of the spectrum, and what bounces back is what you see. Two different physical processes, and the printable set is smaller.

So some colours have no CMYK recipe. Pure RGB blue (0, 0, 255), saturated orange, vivid green and every neon come back dull and grey. Nothing warned you; the file simply printed wrong.

What to do:

- **Soft proof.** Illustrator and Photoshop: View > Proof Colors, **Ctrl+Y**, with a CMYK profile set; Photoshop's Gamut Warning greys out unprintable areas. Inkscape's press colour management has historically been thin, so proof in Scribus instead.
- **Better: choose colours from printed paper.** Pick from a swatch book or the shop's printed CMYK chart and note the values. That removes the screen from the decision entirely.

Spot colours, and when they are worth it

A spot colour is an ink mixed to a recipe and printed from its own plate rather than simulated from four process inks. Use one when a brand colour must match across ten suppliers, when the job is one or two colours and plates beat full process, or when you need a metallic, fluorescent or very saturated colour CMYK cannot reach. Each spot is another plate, so on a full-colour job convert spots to process unless the client is paying.

Pantone's libraries left Adobe in 2022 for a paid plug-in, and the books are expensive. Most small shops work from their own printed CMYK charts, which is legitimate and costs nothing.

Bleed, trim, safe area

Three rectangles, and confusing them is the commonest print mistake.

- **Trim** is the finished size. A visiting card's trim is 90 × 54 mm.
- **Bleed** is artwork extended *past* the trim, normally **3 mm** in India and Europe or **0.125 in** in the US, because a guillotine cutting 500 sheets drifts by a millimetre or two. Without it, that drift shows as a white sliver along one edge — so pull a background photograph out to the bleed line, never the trim line.
- **Safe area** is 3–5 mm *inside* the trim. Nothing important goes there, because the same drift cuts into it. A border sitting exactly on the trim is the worst decision in print: any drift makes it visibly uneven on one side.

300 dpi, and when it is wrong

"Always 300 dpi" is a rule of thumb for images reproduced at **final size** at reading distance. It is wrong in both directions.

The real number comes from the halftone screen. Print reproduces tone as dots at a **line screen** in lines per inch, and images want roughly **twice** that: newspaper at about 85 lpi is fine on 170 dpi, commercial print at 150 lpi wants 300, and a hoarding seen from 50 metres is normal at 20–40 dpi full size — the same thing as building it at one-tenth scale at 300.

And **vector art has no dpi at all.** Do not let anyone ask you for a 300 dpi logo; ask what the finished size is.

Rich black and 100K

A large area of 100% K alone prints a washed-out dark grey, obviously weaker than a photograph's black. **Rich black** adds inks underneath — a common build is 60C 40M 40Y 100K.

- **Total ink coverage.** Add the four percentages. Presses cap it, often near 300% on coated stock and 240% on uncoated and newsprint; too much ink does not dry and offsets onto the next sheet. Ask the shop.
- **Never use rich black for small text or thin lines.** Four plates never register perfectly, and a fraction of a millimetre of drift puts a coloured fringe around every letter. Small text is 100K only.

Overprint, and the object that disappears

By default an object **knocks out** what is beneath it: the press does not lay background ink where the object sits, so the inks do not mix. **Overprint** switches that off and prints the object on top.

Black text is normally set to overprint, so misregistration leaves no white gap around letters.

The expensive failure: **a white object accidentally set to overprint disappears completely.** White in CMYK is 0/0/0/0, the absence of ink. Overprinting nothing on top of a photograph prints nothing, and the photograph shows straight through where your white logo should have been. Perfect on screen, wrong on paper.

Turn on **overprint preview** before every export: Illustrator's View menu, Acrobat's Output Preview, Scribus's Preview Mode.

Making the PDF the press will accept

PDF/X is a PDF with the dangerous parts forbidden: fonts embedded, colour defined, an output intent named.

- **PDF/X-1a:2001** — conservative. Everything CMYK or spot, all transparency flattened. Still the default request at many shops.

- **PDF/X-4** — transparency live and colour-managed, better if the shop's RIP handles it.

Ask which.

Illustrator and InDesign: Save As or Export > Adobe PDF, the `[PDF/X-1a:2001]` preset, then Marks and Bleeds for 3 mm bleed and offset trim marks. CorelDRAW: Publish to PDF, the PDF/X-1a or Prepress preset, bleed and crop marks on the Prepress tab. Scribus: Export > Save as PDF, bleed on the Pre-Press tab, PDF/X-3 and an output intent in the Colour tab.

The one-minute check before you send

Open your own exported PDF as if a stranger had made it:

1. Page size equals **trim**, bleed past the crop marks on all four sides, nothing important within 4 mm of trim
2. Zoom to 400% on small black text — 100K, not four-colour
3. Overprint preview on: has anything vanished?
4. Right pages, right order, fonts embedded

Then send it **with a plain-language note** — shops read this faster than any file:

> *Finished size 90 × 54 mm, 3 mm bleed, 4/4 colour, 300 gsm matt art card, matt lamination both sides, 500 pieces.*

BEFORE YOU MOVE ON

A flyer comes back from the press and a white logo that sat over a photograph has vanished — the photograph shows through where the logo should be. What most likely happened?

A. Transparency was flattened during PDF export and the logo was dropped from the flattened region.
B. The logo sat outside the bleed area and was trimmed off during cutting.
C. The logo was set to 100% K by mistake and printed dark against the dark parts of the photograph.
D. The white object was set to overprint, so instead of knocking out the photograph beneath it, its 0/0/0/0 ink values printed on top — which is no ink at all. Overprint preview before export catches this.

56 · 8 min

What a RIP actually does with your file

THE ONE THING TO KEEP

A RIP interprets, flattens, colour-converts and screens, and each stage has its own characteristic failure — so a job that times out is almost always node explosion, nested transparency, oversized clipped images or too many clipping paths, and none of it is visible in a screen view of the PDF.

Four stages, and a failure at each

A **RIP** — raster image processor — is the software that turns your PDF into the pattern of dots a press, platesetter or large-format printer lays down. Knowing its four stages tells you where a job can fail.

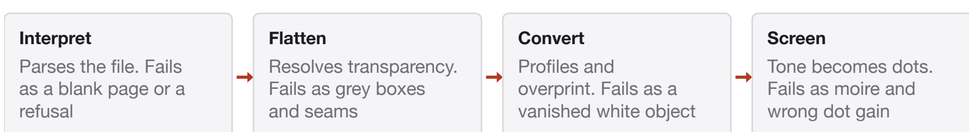
1. Interpret. The RIP parses the PDF: page tree, fonts, images, vector operators, colour spaces. Failures here are structural — a corrupt file, a font it cannot decode, a PDF feature its version does not know about. Symptom: the job refuses outright, or one page comes out blank.

2. Flatten transparency. If the workflow cannot carry live transparency to the end, this is where it is resolved into opaque regions. In a PDF/X-1a workflow your application did it already; in PDF/X-4 the RIP does it, with full knowledge of the device. Failures here are the grey boxes and hairline seams from the earlier module.

3. Colour convert. Source colours are converted to the device's space through profiles, honouring the output intent, spot colour definitions and overprint settings. Failures here are everything in the colour module: wrong profile, spot converted to process, white set to overprint and vanishing.

4. Screen and render. Continuous tone becomes halftone dots at the line screen and angles for that device, and the result is written as a raster for the imaging engine. Failures here are moiré, and dot gain if the curves are wrong.

Four stages, and the failure at each



None of these four stages is visible in a screen view of the PDF, which is why a file that looks right in Acrobat can still come back wrong. Every check worth running is run against the exported file rather than against the design document.

Why a job times out

A RIP has finite memory and a time budget. Files that break it are almost always one of four things:

- **Node explosion.** A traced logo with 40,000 points, or a page where somebody outlined an entire book's text.
- **Deeply nested transparency.** Groups inside groups with blend modes, each requiring its own compositing pass.
- **Enormous embedded images**, especially ones clipped down to a small frame so the RIP decodes 200 MB to paint 20 mm.

- **Hundreds of clipping paths**, each of which constrains the rendering of everything below it.

The fix is the same for all four: simplify before you export. Simplify traced paths, flatten what does not need to stay live, crop image data on export, release clips you do not need.

"It looks fine in Acrobat" is not proof

Acrobat renders your PDF with its own engine, on a screen, at low resolution, with anti-aliasing. The RIP renders it with a different engine, at 2400 dpi, with no anti-aliasing, converting colour and applying overprint.

Two things Acrobat hides unless you ask it to show them:

- **Overprint.** Off by default in the free Reader on some platforms. Turn on Acrobat's *Output Preview* and the white-object-set-to-overprint failure becomes visible.
- **Separations.** The composite view looks correct even when your black text is made of four plates.

The honest test is not a screen view. It is the checks in the next few lessons, run against the exported file.

PostScript and why it still matters

Before PDF, jobs went to press as **PostScript**: a programming language the RIP executed. PostScript is Turing-complete, which meant a file could take unbounded time to render, and a printer could genuinely hang on a valid file.

PDF is a *restricted* subset with the programmability removed — a static description rather than a program. That is the main reason it replaced PostScript for delivery, and the reason it is safe to send a stranger a PDF.

You will still meet PostScript in two places: old workflows where a driver produces a `.ps` and a distiller makes the PDF, and EPS files, which are PostScript fragments. Both are worth avoiding when a direct PDF export is available, because the extra conversion is one more place for transparency to be flattened without your knowledge.

What to ask your printer

> *What RIP do you run, and does it handle live transparency? Do you want PDF/X-1a or PDF/X-4? Do you apply trapping in-RIP?*

Three questions. The answers determine your export settings for every job you send them, so you ask once and write it down.

One more thing the RIP decides

The RIP, not your file, applies the device's **tone reproduction curve** — the compensation for dot gain discussed in the colour module. This is why the same PDF sent to two printers with different presses produces two correctly different sets of plate values and, ideally, the same printed result.

It also means a file you have "corrected" by lightening the midtones because the last job printed dark is now doubly corrected, and will print washed out. If a job prints dark, the answer is the output intent or the press calibration, not an adjustment layer. Fix it upstream and tell the printer what you observed, rather than compensating blindly in artwork that will be reused.

BEFORE YOU MOVE ON

A 16-page brochure returns from the printer with a note that the RIP timed out on page 9. That page carries a company logo traced from a scan, a photograph clipped into a small circular frame, and body text. What should be checked first?

A. The traced logo's node count and the clipped photograph's real pixel dimensions.
B. Whether the page was exported as PDF/X-1a, since flattening transparency during export is what consumes all of the RIP's memory.
C. Whether the body text was outlined, since outlined text is the usual cause of RIP timeouts on text-heavy pages.
D. The colour profile assigned to the photograph, since an unexpected profile forces the RIP into a slow per-pixel conversion.

57 · 8 min

PDF/X, and what each flavour forbids

THE ONE THING TO KEEP

Every PDF/X flavour embeds fonts, defines a TrimBox and names an output intent, and they differ in what they allow: X-1a forbids RGB and flattens transparency, X-4 keeps transparency live for the RIP to flatten once at device resolution.

PDF/X is a PDF with the dangerous parts removed

An ordinary PDF can contain RGB images, missing fonts, JavaScript, video, layers, comments, form fields and an unstated colour intent. A press cannot do anything sensible with most of that.

PDF/X is a family of ISO standards defining subsets suitable for exchange. Every flavour requires:

- All fonts embedded.
- No JavaScript, no video, no form fields, no encryption.
- A **TrimBox or ArtBox** defined, so the finished size is unambiguous.
- An **output intent** — a named characterisation of the intended printing condition, such as `FOGRA39` or `Japan Color 2001 Coated`.

The last one is the point of the standard. A PDF/X file says what printing condition its colours were prepared for. A plain PDF does not, and everything else is guesswork.

The flavours you will meet

PDF/X-1a:2001 and :2003. The conservative one. Everything must be CMYK or spot — **no RGB at all** — and **all transparency is flattened** before export. No live transparency, no colour management inside the file beyond the output intent. Still the default request at a large number of shops, because it cannot surprise their RIP.

PDF/X-3:2002/2003. Permits colour-managed data — RGB and Lab with embedded profiles — leaving conversion to the RIP. Transparency still flattened. Used mostly in Europe, and Scribus's default export target.

PDF/X-4:2010. Permits **live transparency and layers**, and colour-managed content. The RIP flattens once, at device resolution, with full knowledge. This is the better standard where the shop's RIP supports it: no atomic regions in your file, no hairline seams, smaller files.

PDF/X-5 and PDF/X-6. X-5 adds external references (a file that points at another file), rarely used in general commercial work. X-6 (2020) is the modern successor built on PDF 2.0. Neither is what a local shop will ask for.

What each PDF/X flavour forbids

- **2001** — PDF/X-1a. CMYK and spot only, no RGB at all, and every piece of transparency flattened before it leaves your machine
- **2002** — PDF/X-3. Colour-managed RGB and Lab allowed, with conversion left to the RIP. Transparency still flattened
- **2010** — PDF/X-4. Live transparency and layers. The RIP flattens once, at device resolution, knowing the device
- **2020** — PDF/X-6, built on PDF 2.0. Correct, modern, and not what a local shop is going to ask you for

Every one of them embeds the fonts, defines a TrimBox and names an output intent. They differ only in what they allow through, so the question to the shop is two words long: X-1a or X-4?

Choosing, in two questions

1. **"Do you want X-1a or X-4?"** Ask. If they say X-1a, send X-1a. If they say "either", send X-4 for anything with transparency and X-1a for flat line work.
2. **"Which output intent?"** Their answer names the profile you convert to. If they cannot say, ISO Coated v2 (FOGRA39) for coated European-style work and Japan Color 2001 Coated for a great deal of Asian work are defensible defaults — but the question is better than the default.

Exporting it

- **InDesign / Illustrator**: File ▸ Export ▸ Adobe PDF (Print), then the `[PDF/X-1a:2001]` or `[PDF/X-4:2008]` preset. Marks and Bleeds ▸ *Use Document Bleed Settings* and *Crop Marks* with a 3 mm offset. Output ▸ set the destination and the output intent profile.
- **CorelDRAW**: File ▸ Publish to PDF ▸ Settings ▸ the PDF/X-1a or PDF/X-4 preset; bleed and crop marks on the Prepress tab.
- **Scribus**: File ▸ Export ▸ Save as PDF. Choose PDF/X-1a, X-3 or X-4 in the General tab's compatibility dropdown, set bleed on Pre-Press, and **set the output intent profile on the Colour tab** — Scribus will refuse to export a PDF/X file without one, which is the standard being enforced correctly.
- **Affinity Publisher**: File ▸ Export ▸ PDF, with the PDF/X presets in the preset list.

Where the shop's answer comes from

A shop that says "X-1a only" is usually running an older RIP, or has been burned by a transparency job and does not want the argument. Neither is unreasonable. A shop that says "X-4 preferred" has a modern workflow and is telling you they would rather flatten it themselves, correctly, than receive somebody's pre-flattened approximation.

If you have a choice and the job has soft shadows, transparency or placed PSDs, X-4 is materially better and the file will be smaller. If the job is flat line work with spot colours, the flavour makes almost no difference and X-1a is one less thing to discuss.

Verifying you got what you asked for

Never trust the preset name. Check the file:

```
pdfinfo job.pdf | grep -i -E 'page size|pdf version'
pdffonts job.pdf
```

For the X-conformance claim itself, Acrobat Pro's Preflight has PDF/X profiles, and the free **veraPDF** validator, though aimed primarily at PDF/A, also validates PDF/X conformance levels. Ghostscript will tell you about structure but not about conformance.

The honest limitation

A file can be technically valid PDF/X and still be wrong for the job: correct structure, wrong output intent, images at 90 ppi, artwork 4 mm off the trim. **Conformance is a floor, not a check on your work.** The standard guarantees the press can process the file. Everything about whether the file is *right* is the next three lessons.

BEFORE YOU MOVE ON

A job containing several soft drop shadows is exported as PDF/X-1a because the shop's preset list offered it first. What is the most significant consequence of that choice?

A. The output intent is omitted, since X-1a defines colour purely by device values rather than by any printing characterisation.
B. Fonts will be embedded as full sets rather than subsets, increasing the file size substantially.
C. Layers are preserved, so the shop can switch the shadows off if the RIP struggles with them.
D. Transparency is flattened on export, splitting the page into atomic regions where seams can appear.

58 · 8 min

The five rectangles inside a PDF

THE ONE THING TO KEEP

The press cuts from the TrimBox, so set the document page size to trim and bleed in the document setup rather than exporting a page at trim-plus-bleed — and verify with `pdfinfo -box`, which is arithmetic anybody can check in ten seconds.

Your page has five sizes, not one

A PDF page carries up to five rectangles, and most disagreements with a printer are one of them being missing or wrong.

- **MediaBox** — the sheet the page is described on. Always present. Everything else is defined inside it.
- **CropBox** — what a viewer displays. Defaults to the MediaBox.
- **BleedBox** — the area including bleed, where artwork may extend.
- **TrimBox — the finished size after cutting.** This is the one that matters.
- **ArtBox** — the meaningful content area. Rarely used in commercial print.

The press positions and cuts from the **TrimBox**. A file with no TrimBox forces the operator to guess, and "no trim box defined" is one of the most common preflight rejections there is.

A worked example

A 90 × 54 mm visiting card with 3 mm bleed and crop marks:

- **TrimBox** — 90 × 54 mm. The card.
- **BleedBox** — 96 × 60 mm. Trim plus 3 mm on each side.

- **MediaBox** — larger again, typically 106 × 70 mm or so, to make room for crop marks and any colour bars outside the bleed.

A very common error is exporting the page at 96 × 60 mm with no boxes defined. The printer then sees a 96 × 60 mm page and either cuts to it — producing an oversized card with no bleed — or asks you to re-export.

A 90 by 54 mm card, in boxes

MediaBox, about 106 by 70 mm	The sheet the page is described on, with room for crop marks
BleedBox, 96 by 60 mm	Trim plus 3 mm on every side. Artwork may run out to here
TrimBox, 90 by 54 mm	The card. The press positions and cuts from this one

Set the document page size to trim and the bleed in the document setup, and these fall out correctly. Export a page at 96 by 60 mm with no boxes defined and the press either cuts to that size or sends the job back.

Reading them

```
pdfinfo -box job.pdf
```

Output looks like:

```
Page size:      300.47 x 198.43 pts
MediaBox:       0.00  0.00  300.47 198.43
CropBox:        0.00  0.00  300.47 198.43
BleedBox:       8.50  8.50  291.97 189.93
TrimBox:       17.01 17.01  283.46 181.42
ArtBox:        17.01 17.01  283.46 181.42
```

Check the arithmetic. TrimBox width here is 283.46 – 17.01 = 266.45 pt = **93.99 mm**. BleedBox width is 291.97 – 8.50 = 283.47 pt = **100 mm**. So trim is 94 mm and bleed extends 3 mm each side. If those numbers are not the sizes you agreed, stop before sending.

`mutool info` and Acrobat's Document Properties give you the same information.

Acrobat also draws the boxes on screen: Preferences ▸ Page Display ▸ *Show art, trim & bleed boxes*. Turn it on once and every PDF you open shows you its geometry without a command line.

Setting them correctly

The boxes come from how you set up the document and the export, not from a separate dialog.

- Set the **document page size to trim**, not to trim-plus-bleed. This is the single mistake that generates most of the box problems.
- Set **bleed in the document setup** (3 mm all round), not by drawing a larger rectangle.
- On export, tick **use document bleed settings** and turn on crop marks with an **offset** of at least the bleed amount — so the marks sit outside the bleed rather than printing into it.

Scribus, InDesign, Affinity and CorelDRAW all follow this pattern. In Inkscape, which has no native bleed concept, set the SVG page to the trim size and use the PDF export's bleed/margin field — or, better, do the final assembly in Scribus.

Fixing a file you cannot re-export

Occasionally you receive a PDF with wrong boxes and no source. Ghostscript can rewrite them:

```
gs -o fixed.pdf -sDEVICE=pdfwrite \
   -dUseTrimBox=true -dFIXEDMEDIA \
   -c "[/CropBox [17 17 283 181] /PAGES pdfmark" -f job.pdf
```

`pdftk`, `qpdf` and `mutool` can also manipulate page geometry. Treat all of it as a rescue, not a workflow — the right answer is to fix the source document and export again.

Crop marks, registration marks and colour bars

- **Crop marks** show where to cut. Offset them beyond the bleed.
- **Registration marks** are printed in all plates and let the press check alignment. The press adds these; you normally do not.
- **Colour bars** are strips of known patches the press measures during the run. Also theirs, not yours.
- **Page information** — filename, date, plate name — is optional and genuinely useful on a multi-page job.

Do not draw any of these by hand. Every application generates them, aligned to the boxes, and a hand-drawn crop mark in the wrong place is worse than none.

BEFORE YOU MOVE ON

A designer sets up a visiting card document at 96 × 60 mm so the 3 mm bleed is "already included", draws the card's trim area as a guide, and exports without crop marks. What will the printer see?

A. A 96 × 60 mm TrimBox and no BleedBox, so the card is cut at 96 × 60 mm with no bleed.

B. A 90 × 54 mm TrimBox derived from the drawn guides, with a correct 3 mm BleedBox surrounding it.

C. No boxes at all, because a PDF only carries box definitions when crop marks are enabled on export.

D. A correct BleedBox but no TrimBox, so the press will default to cutting 3 mm inside the page edge.

59 · 9 min

Preflighting your own PDF, free

THE ONE THING TO KEEP

Run seven checks against the exported PDF rather than the design file — boxes, fonts, image ppi, ink coverage, separations, overprint simulation and structure — all of which poppler-utils and Ghostscript do for nothing, and none of which will catch a misspelled name.

Check the file you are sending, not the file you made

The design document is your intention. The PDF is the artefact. Every check in this lesson runs against the PDF, because that is what the press receives.

You do not need Acrobat Pro. Two free packages cover nearly everything: **poppler-utils** (`pdfinfo`, `pdffonts`, `pdfimages`, `pdftoppm`) and **Ghostscript**. Both install on Windows, macOS and Linux, and both run on a laptop that could not open Illustrator.

The seven checks

1. Page size and boxes.

```
pdfinfo -box job.pdf
```

Confirm TrimBox equals the agreed finished size and BleedBox extends past it. Previous lesson.

2. Fonts.

```
pdffonts job.pdf
```

Every row must show `yes` under `emb`. A `no` means the font is referenced but not embedded, and the press will substitute. A six-letter prefix like `ABCDEF+` means a subset, which is normal and good. A font you did not choose appearing in the list means something substituted silently.

3. Image resolution.

```
pdfimages -list job.pdf
```

Read the `x-ppi` and `y-ppi` columns. Anything well below your target is a problem; anything enormous in pixel dimensions next to a small placed size means a clipped image was not cropped on export.

4. Colour and ink coverage.

```
gs -q -o /dev/null -sDEVICE=inkcov job.pdf
```

You get four numbers per page — C, M, Y, K coverage. A page you believe is mono showing cyan, magenta and yellow means your "black" is a four-colour build. A page showing coverage far above the press's ink limit needs attention.

5. Separations.

```
gs -o sep-%02d.tif -sDEVICE=tiffsep job.pdf
```

One TIFF per plate, spot plates named. If you expected two plates and get five, you have accidental spots or RGB that separated. Next lesson covers reading these.

6. Overprint. Ghostscript can simulate it:

```
gs -o overprint.png -sDEVICE=png16m -dSimulateOverprint=true -
r150 job.pdf
```

Compare against a normal render. Anything that disappears was set to overprint and should not have been. This catches the vanishing white object.

7. Structure.

```
qpdf --check job.pdf
```

Reports damage, broken cross-reference tables and recoverable errors before the RIP finds them.

A script you can keep

```
#!/bin/sh
f="$1"
echo "== BOXES ==";  pdfinfo -box "$f"
echo "== FONTS ==";  pdffonts "$f"
echo "== IMAGES =="; pdfimages -list "$f"
echo "== INK ==";    gs -q -o /dev/null -sDEVICE=inkcov "$f"
echo "== CHECK ==";  qpdf --check "$f" 2>&1 | head -5
```

Save it as `preflight.sh`, make it executable, and run it on every file before you send one. Thirty seconds, and it has caught every failure in this module at least once.

The GUI options

- **Scribus** has a built-in *Preflight Verifier* that runs before export and reports missing images, low resolution, overset text, non-embedded fonts and transparency in a flavour that forbids it. It is genuinely good and it is free.
- **Acrobat Pro** has the most complete preflight available, with editable profiles and fixups.
- **PDF24, Ghostscript front-ends and online validators** exist. Be careful about uploading a client's unreleased artwork to a free web service; read what the service says it does with your file, and default to local tools.

What preflight cannot tell you

It cannot tell you the phone number is wrong, the client's name is misspelled, the photograph is of the wrong building, or the price is out of date. Those are the errors that actually get reprinted, and the only defence is a human reading the proof.

Print the PDF at 100%, read every word out loud, and have somebody who did not design it read it too. Both of those catch things no software will.

Rendering a page to look at

Sometimes you want to see what the file really contains rather than what a viewer chooses to show you.

```
pdftoppm -r 300 -png -f 4 -l 4 job.pdf page
```

That writes page 4 at 300 ppi as a PNG, rendered by poppler rather than by Acrobat. Open it in GIMP and zoom in on the small text, the hairlines and the edges of any shadow. Rendering artefacts that were hidden by a viewer's anti-aliasing become visible, and you are looking at something much closer to what the RIP will produce.

For a long document, render every page at a low resolution and flick through the images. Missing content, a page in the wrong orientation and an accidentally blank page all show up in a contact sheet faster than in a scroll through a viewer.

BEFORE YOU MOVE ON

`pdffonts job.pdf` lists a font called `Helvetica` with `no` in the `emb` column, although the designer never used Helvetica. What does this most likely indicate?

A. The PDF was saved from an older application that always lists the base fonts, and it can safely be ignored.
B. Helvetica is one of the PDF base fonts, so it is resolved by the RIP and does not need embedding.
C. A font used in the document was unavailable and was silently substituted.
D. The font is embedded as a subset, and subset fonts are reported as not embedded because their names carry a prefix.

60 · 8 min

Reading separations, and the fifth plate you did not order

THE ONE THING TO KEEP

Separations answer questions a composite cannot: whether your black text is really 100% K, whether a spot survived as a spot, whether there is an unordered fifth plate, and whether total ink exceeds what the press can dry.

What separations show you

A composite view shows the four inks combined and looks like the finished job. **Separations** show one ink at a time, and they reveal things a composite hides entirely.

Four questions that only separations answer:

1. Is my black text actually 100% K, or is it four colours?
2. Is my spot colour still a spot, or did it convert?

3. Are there plates I did not ask for?
4. Is anything set to overprint that should not be?

Looking at them

Acrobat Pro: Print Production ▸ Output Preview. Tick and untick each separation, hover to read ink percentages under the cursor, and set *Total Area Coverage* to a number — 300%, say — to highlight every area over the limit in a warning colour.

Free, with Ghostscript:

```
gs -o sep-%02d.tif -sDEVICE=tiffsep -r300 job.pdf
```

You get `sep-01.Cyan.tif`, `sep-01.Magenta.tif` and so on, plus a file for every spot colour, named after the spot. Open them in GIMP or any viewer.

The file list alone answers question 3 before you look at a single image. Two plates expected, five files produced: something converted.

Free, for ink coverage:

```
gs -q -o /dev/null -sDEVICE=inkcov job.pdf
```

Prints four numbers per page as fractions of full coverage. Useful as a quick screen across a long document.

The black-text check

Open the Cyan separation and look at your body text. If you can read it, your black is not 100% K.

This happens in three ways, all common:

- **Text drawn in RGB black** (`#000000`) converted with a standard profile, which becomes a rich black of roughly 90/70/70/100 rather than 0/0/0/100.

- **Text copied from another document** carrying its own colour definition.
- **A logo whose black was defined as rich black**, then reused at 8 mm on a form.

The consequence, from the colour module: four plates never register perfectly, so small text built from four inks gets a coloured fringe. Fix it in the source by setting the text to a 100% K swatch, not by adjusting the export.

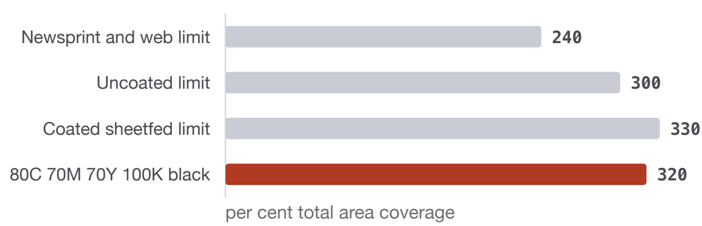
Total area coverage

Add the four percentages at any point. Presses cap the total, typically:

- Coated sheetfed: **300–330%**
- Uncoated: **260–300%**
- Newsprint and web: **240%**

Above the limit, ink does not dry, sheets offset onto each other in the stack, and the press slows down or refuses.

Three ink limits, and one rich black



A 60/40/40/100 rich black totals 240% and is safe anywhere. The 80/70/70/100 build shown here totals 320%: comfortable on coated sheetfed, over the limit on uncoated, and impossible on newsprint, where it will not dry and the stack offsets.

Where does too much ink come from? Usually a rich black built too heavily (60/40/40/100 is 240%, which is fine; 80/70/70/100 is 320%, which is not on uncoated), or a dark photograph converted with a profile whose ink limit is higher than the press's.

Acrobat's TAC warning finds it instantly. Free, the `inkcov` output gives you a page-level total, and a conversion to the correct destination profile prevents the problem rather than detecting it.

The accidental fifth plate

Common sources, in order:

- An **RGB object** that nobody converted — often a logo pasted from a website, or a chart from a spreadsheet.
- A **spot swatch** used once, on a hidden or off-page object.
- A **Registration** swatch used as a black. Registration prints on every plate and is for crop marks only; a headline in registration black prints 400% ink and will not dry.
- A **duplicate spot** with a slightly different name — `PANTONE 2925 C` and `Pantone 2925 C` are two plates to a RIP.

The swatches panel in your layout application usually has a *Select All Unused* command. Delete them before export, and check the separations after.

Separations on a one-colour job

Worth doing even when you think it cannot go wrong. A "black only" job that separates into four plates is quoted as four-colour, and the difference on a 5,000-sheet run is substantial. The usual culprit is a logo supplied as an RGB or CMYK file and never converted to 100% K, or a greyscale photograph that carries an RGB profile.

The fix is upstream: convert the image to greyscale, set every swatch to K-only, and check again. A two-minute check that removes three plates from a quotation pays for itself immediately, and it is the kind of thing a client notices you doing.

The habit

Open the separations on every job that matters, look at four things — black text, plate count, spot names, overprint — and close them. Two minutes, and it catches the failures that produce a reprint rather than a complaint.

BEFORE YOU MOVE ON

A two-colour letterhead is exported and Ghostscript's `tiffsep` produces five files: Cyan, Magenta, Yellow, Black and one named after the brand spot. The visible design uses only black and the spot. Where should the designer look first?

A. The output intent profile, since an incorrect intent will cause the RIP to generate process plates alongside the spot ones.
B. The trapping settings, since in-RIP trapping generates additional process separations around spot colour edges.
C. The PDF/X flavour, since X-4 preserves live transparency by splitting spot colours into process equivalents.
D. For an RGB or process-coloured object in the file — a pasted logo, a hidden object, a registration black.

61 · 8 min

Paper, and everything done after printing

THE ONE THING TO KEEP

Weight is not thickness, coated and uncoated differ in gamut and dot gain by a wide margin, grain direction decides whether a fold cracks, and every finishing process — varnish, foil, emboss, die-cut — is supplied the same way: its own layer, a named spot swatch, set to overprint.

Paper is a design material

Two jobs printed identically on different stocks are different objects. Knowing the vocabulary lets you specify rather than accept.

Weight is grammage, in grams per square metre. Rough anchors: ordinary copier paper 80 gsm; a good letterhead 100–120 gsm; a leaflet 130–170 gsm; a visiting card 300–350 gsm; a folder 350 gsm plus.

Weight is not thickness. Bulk varies by stock, which is why the spine calculation in the previous module needs a caliper from the printer rather than a gsm figure.

Surface decides how ink sits:

- **Coated art**, gloss or matt — a clay coating holds ink on the surface. Sharp images, strong colour, larger gamut, low dot gain. What most brochures are printed on.
- **Uncoated** — ink soaks in. Softer image, muted colour, 20–25% dot gain, lovely to write on and to hold. Letterheads, forms, literary books.
- **Maplitho** — the uncoated wood-free paper that runs an enormous share of Indian commercial and book printing.
- **Newsprint** — cheap, absorbent, grey, and a completely different set of numbers.
- **Textured, laid, recycled** — specify by sample, not by name, because names vary by mill.

Grain direction is the direction the fibres lie. Paper folds cleanly *along* the grain and cracks *across* it, and a folded job with the grain running the wrong way produces a ragged, cracked fold on every copy. Tell the printer where the folds are; they choose the grain.

Finishing, and what each one demands of your file

Lamination — a plastic film over the sheet. Gloss, matt or soft-touch. Adds durability and stiffness, and shifts colour slightly: matt lamination noticeably dulls and darkens. If colour is critical, ask to see a laminated sample.

Spot UV — a glossy varnish on selected areas only. Needs a **separate spot colour layer** in your file, usually named `SpotUV`, containing solid shapes where the varnish goes, set to overprint. Keep it at 100% of a spot; it is a presence-or-absence mask, not a tint.

Foiling — metallic foil pressed on with a heated die. Same idea: a separate spot layer, solid shapes, named `Foil`. Fine detail and small type do not foil well; ask for a minimum stroke width.

Embossing and debossing — raised or recessed. Another separate layer. Remember from the logo module that a mark distinguished only by colour disappears entirely here.

Die-cutting — a shaped cut. Supply the **die-line** as a separate spot-coloured layer, named `Dieline` or `CutContour`, drawn as thin outlines with no fill, and set to overprint so it does not knock out the artwork beneath.

Creasing, perforating and folding — usually shown on the same technical layer as the die-line, with different spot names (`Crease`, `Perf`) so the finisher can separate them.

Numbering, scoring, round-cornering, drilling, padding, gumming — all normal, all cheap, and all things to ask about rather than to discover.

The technical-layer convention

Every one of the above shares a pattern, and if you learn it once you can prepare any of them:

1. Put the marks on their own layer, above the artwork.
2. Colour them with a **named spot swatch** — `Dieline`, `SpotUV`, `Foil`.
3. Set the objects to **overprint**, so they do not knock a hole in the artwork.
4. Confirm the layer appears as its own separation, using the checks from the previous lesson.
5. Tell the printer the layer names in your covering note.

Names vary by shop. Ask what they use; `CutContour` is the convention in large-format and vinyl work, `Dieline` in packaging.

The spec you send

One paragraph that answers everything a shop needs:

> *Finished size 90 × 54 mm, 3 mm bleed, printed 4/4, on 350 gsm matt art card, matt lamination both sides, spot UV on the logo (layer `SpotUV`), round corners 3 mm radius, quantity 1,000.*

Every number in it is something you had to decide. Being able to write that paragraph is most of what a print buyer does.

Ask for samples before you specify

Paper names mean different things at different mills, and a stock you liked at one shop may not exist at another. Ask for a **swatch book** or a set of printed samples early in a relationship; most suppliers give them away because a designer who specifies confidently is a designer who orders.

Then keep them. A drawer of real paper, labelled with weight, name and supplier, is worth more than any online chart, because the two things that decide a stock — how it feels and how ink sits on it — cannot be photographed.

BEFORE YOU MOVE ON

A folded A4 leaflet is printed on 170 gsm uncoated stock and every copy has a ragged, cracked white line along the fold. The colour and registration are correct. What is the most likely cause?

A. The fold was placed over a solid dark area, and ink always cracks when folded across a heavy ink film.

B. The stock's grain runs across the fold rather than along it, so the fibres break instead of bending.

C. 170 gsm is too heavy to fold without creasing first, regardless of grain direction.

D. Uncoated stock absorbs ink into the fibres, which stiffens the sheet locally and causes it to split at the fold.

62 · 8 min

Approvals, and who pays for the reprint

THE ONE THING TO KEEP

Keep four artefacts on every job — the exact PDF with a checksum, the message that sent it, the signed proof and the preflight output — because a reprint dispute is only answerable by whichever party can show what was actually supplied and approved.

The question every dispute reduces to

Five thousand copies are wrong. Somebody pays. The only question that matters is whether the supplied file was wrong or the press deviated from it, and it is answerable only if you kept the evidence.

What to keep, on every job

- **The exact PDF you sent.** Not "the final version" — the actual bytes. Archive it alongside the job, and record a checksum:

```
shasum -a 256 job-press-v3.pdf > job-press-v3.sha256
```

Thirty seconds, and it converts "this is the file I sent" from an assertion into a fact.

- **The email or message that sent it**, with the spec paragraph in the body.
- **The proof**, signed and dated, or the email in which the client approved.
- **Your preflight output**, saved as a text file.

Four artefacts. Together they answer the question before it becomes an argument.

The four artefacts, and when each is made

Before sending	Run the preflight and save its output beside the job as a text file
On sending	Archive the exact bytes you sent and record a checksum of them
With the file	Keep the message carrying the specification paragraph
On approval	The proof, signed and dated, photographed, held by both sides
Months later	Five thousand copies are wrong, and those four answer who pays

None of this is bureaucracy. A reprint dispute reduces to one question, whether the supplied file was wrong or the press deviated from it, and it is answerable only by whichever party kept the evidence.

Approving a proof properly

1. **View under a known light.** D50 if you have it, north daylight otherwise. Never a warm tube.
2. **Compare against the reference**, not against memory or a screen.
3. **Read every word out loud.** Names, phone numbers, prices, dates, the word "public". Have a second person read it independently.
4. **Mark corrections physically**, in a colour not present in the job, with a number beside each so they can be counted.
5. **Sign and date it**, photograph it, and send the photograph back so both parties hold the same record.

What a signature means

Signing a proof normally means: *the content is correct and I accept this as the standard for the run*. It transfers responsibility for anything visible on the proof.

So read what you are signing for. A **digital proof** shows content, position and approximate colour. It does not show the stock, the lamination, the fold, the trim accuracy or the exact ink density. Signing one does not mean you accepted a 4 mm trim drift.

Where colour matters and the budget allows, ask for a **press pass**: be there when the run starts, look at the first good sheet, and sign that. It is the strongest approval available and it costs an hour.

The disputes that actually happen

Colour is not what I expected. Answer: what was it compared against? If a printed reference was agreed and a ΔE tolerance was named, this is measurable. If the reference was a screen, nobody can win, which is why the colour module insisted on a printed reference.

Trim is off. A 1–2 mm drift is within normal guillotine tolerance and is why safe areas exist. A border sitting on the trim line will look uneven and that is a design decision, not a press fault. Beyond about 2 mm, it is worth raising.

A typo. If it was on the signed proof, it is the approver's. This is uncomfortable and it is why the read-aloud step exists.

The file was rejected or changed. If a shop altered artwork — converted a spot, substituted a font, rebuilt a die-line — ask for the file they output. A reputable shop will send it.

It came back late. Agree the schedule in writing including who does what by when, because a proof sat on for four days is not the printer's delay.

How to handle it when it is your fault

Say so quickly, in plain words, and propose a remedy before you are asked. "The die-line layer was set to knock out rather than overprint. That is my error. I will supply a corrected file today and I will cover the plate cost."

This is not only decent; it is commercially sound. Printers and clients both remember who was straight with them about a mistake, and in a trade that runs largely on repeat work and word of mouth, that memory is worth more than the cost of a set of plates.

Setting expectations at the start

Two sentences in your quotation prevent most of this:

> *Proofs will be supplied for approval before printing. Changes after approval are chargeable, and colour is matched to the supplied printed reference within normal commercial tolerance.*

Say it once, in writing, and the awkward conversation mostly does not happen.

BEFORE YOU MOVE ON

A client approves a digital contract proof. The delivered job has the correct content and colour but is trimmed 4 mm off centre, cutting into a border. The client claims the proof approval covers it. What is the reasonable position?

A. Approval covers everything the job contains, so the trim was accepted along with the content when the proof was signed.
B. The trim is within normal guillotine tolerance, so a 4 mm drift is not raisable regardless of what was approved.
C. The proof covers content, not physical trimming, and 4 mm exceeds normal tolerance.
D. The border was placed too close to the trim line, so the design is at fault and no remedy is available.

---

63 · 9 min

CDR is the most common file in Indian print and the least portable

THE ONE THING TO KEEP

Keep your working file native, send a press-ready PDF by default, and ask the shop which software, version and PDF/X standard they want before you start rather than after you finish.

The situation you will be in

You designed in Illustrator. The shop runs CorelDRAW. Or you designed in CorelDRAW 2024 and the shop runs X7. Or you used Inkscape and the client's other vendor wants "the AI file". Every one of these conversations happens weekly, and getting it wrong costs a reprint.

Here is what each format actually is, and what survives the trip.

CDR

CorelDRAW's native format. It is the working currency of the Indian print trade, of signage, of garment and screen-print units across South and Southeast Asia, and of thousands of small shops elsewhere. Learning CorelDRAW is a genuinely good career decision for that reason alone, and almost no free course teaches it.

It is also the least portable format in the business. It is proprietary and undocumented, and the internal structure changes between versions. **An older CorelDRAW generally cannot open a file saved by a newer one.** That is the single most common file failure in the trade.

The fix is in your Save As dialog: there is a **version dropdown**. Ask the shop what they run and save to that version. Saving down may drop newer effects, so check the result rather than assuming.

Inkscape can import many CDR files through the libcdr library. It usually gets the geometry and often gets the colours; it frequently gets text, effects and some fills wrong. Useful for rescuing artwork, not for production.

AI

Illustrator's native format. Since version 9, an `.ai` file is a PDF with Illustrator's private data attached — but only if **"Create PDF Compatible File"** is ticked in the save dialog. That checkbox is why other programs can open your AI files at all. Untick it and the file gets smaller and becomes unreadable outside Illustrator. Leave it on.

EPS

Old, still requested, and worth understanding for one reason: **EPS does not support transparency.** Export something with a soft shadow, a blend mode or a partially transparent object to EPS and it is flattened — chopped into regions, some of them rasterised. That is where the mysterious grey rectangle behind a logo comes from. EPS is fine for flat line art and a poor choice for anything modern. If someone asks for EPS, ask whether a PDF would do.

SVG

Open, text-based, a W3C standard, and genuinely good. It is the right format for the web and for handing artwork to a developer. But it was designed for screens: **no CMYK, no spot colours, no bleed, no overprint, no output intent.** A print shop asking for SVG usually just means "send something I can open". Send a PDF as well.

PDF

This is the actual lingua franca, and it is the answer most of the time. A PDF exported with vectors intact opens in Illustrator, Inkscape, Affinity and CorelDRAW with varying degrees of editability, and goes straight to a press when exported as PDF/X — which is the next lesson.

Practical rule: send a press-ready PDF unless someone specifically asked for something else. Keep your working file in your own program's native format. Ask the shop what they can open before you start designing, not after you finish.

What round-tripping loses

Moving a file between programs is lossy in ways that are quiet:

- Inkscape SVG filters do not survive into Illustrator
- Illustrator gradient meshes and some live effects do not survive into Inkscape
- CorelDRAW's lens, envelope and contour effects do not survive anywhere unless converted first
- Text keeps its characters but loses OpenType feature settings, and reflows if the font is missing

- Live blends, symbols and linked images become embedded or vanish

If artwork must survive a trip intact, do three things before exporting: convert effects to curves or objects, outline the strokes, and outline the display text. You lose editability and you gain certainty. That is usually the right trade for a file leaving your hands.

The tools, honestly

- **CorelDRAW** — its real advantage is that the shop already has it, but it is also legitimately good at the job: multi-page documents, print merge, imposition, and a workflow built around getting things onto material rather than onto a screen. Paid; Corel has offered both subscription and one-off perpetual licences, so check what is currently sold.
- **Illustrator** — the reference tool, what agency job ads name, the deepest type and colour handling. Subscription only.
- **Inkscape** — free, GPL, mature. Drawing, node editing, booleans, type on a path, tracing, DXF and PDF export are all there and all good. Weaker points: colour management for press has historically been thin because it works in RGB internally, though recent releases have added CMYK handling, so check your version rather than trusting this sentence. It also slows down on very heavy files. For the CMYK gap, pair it with Scribus.
- **Affinity Designer** — reads PSD, AI and PDF, has live non-destructive booleans, and had a one-off price for years. After the Canva acquisition the apps were relaunched in 2025 with free access and a paid tier for some features. Confirm current terms before planning around it.
- **Penpot** — free, open source, runs in a browser, aimed at UI work rather than print, but usable and it opens on a machine that cannot install anything.

The fully free press chain works: **draw in Inkscape, place in Scribus, export PDF/X from Scribus.** Nothing in this course requires a payment.

The message to send the shop

Before you design anything:

> *Finished size, number of colours, material, quantity — and: which software and version do you run, and do you want PDF/X-1a or X-4?*

Four lines. It prevents most of what this lesson is about.

BEFORE YOU MOVE ON

A designer emails a CorelDRAW 2024 file to a print shop, which replies that the file will not open. The shop runs an older CorelDRAW. What is the best response?

A. Rename the file extension to .cdr from the newer save format, since Corel uses the same container across versions.

B. Export to EPS, which every version of every application can read without loss.

C. Ask which version the shop runs, use Save As with that version selected, and in the same message send a press-ready PDF with fonts outlined or embedded so the job can proceed even if the CDR still misbehaves.

D. Send the SVG instead, since it is an open standard and carries the print settings with it.

64 · 8 min

The covering note, and the file a shop can open

THE ONE THING TO KEEP

A nine-line covering note in ordinary words answers everything a shop would otherwise email about and doubles as your record of the specification — and the final look is run on the exported PDF as though a stranger had sent it.

The note is read before the file

A printer opens ten jobs a day. The ones that go smoothly are the ones where a human being said, in ordinary words, what the job is. Attach the file, and put this in the message body:

> ***Job:*** *ACME visiting cards*

> ***Finished size:*** *90 × 54 mm, 3 mm bleed*

> ***Printing:*** *4/4 process, no spot colours*

> ***Stock:*** *350 gsm matt art card*

> ***Finishing:*** *matt lamination both sides, round corners 3 mm*

> ***Quantity:*** *1,000*

> ***File:*** *acme-cards-press-v3.pdf — PDF/X-1a, FOGRA39, fonts embedded, crop marks with 3 mm offset*

> ***Notes:*** *small black text is 100K only. No manual trapping applied.*

> ***Delivery:*** *by 28th, to the Andheri office.*

Nine lines. It answers every question they would otherwise email you about, and it is also a record of what you specified.

Naming files so the right one gets printed

Filenames end up on job dockets, in folders with six other versions, and on a shop floor.

```
acme-cards-90x54-press-v3.pdf
acme-cards-90x54-EDITABLE.indd
```

Include the client, the item, the size, the state and the version. Never use `final`, `final2`, `FINAL-new`, `use this one`, or a space in a filename. On some shop systems a space still breaks a hot folder, and everywhere it makes a command line awkward.

Put the version in the filename, and increment it for every file that leaves your machine. A shop asking "which one did you want printed?" is a shop about to print the wrong one.

How to send it

- **Under 20 MB** — email, and say the file is attached in the first line.
- **Anything larger** — a link from Drive, Dropbox, WeTransfer or your own hosting. Set the link so it does not expire in three days; a job on hold for a fortnight will outlive a default expiry.
- **Confirm receipt and openability.** "Please confirm you can open it and that the trim box reads 90 × 54." That one sentence moves the check from your assumption to their confirmation.

If a shop uses a hot folder or an upload portal, use it, and still send the note.

What to send besides the PDF

Normally, nothing. A press-ready PDF is the deliverable.

Send more only when asked, and know what you are handing over:

- **The layout source plus packaged links and fonts** — only if they will be making changes, and remember from the type module that handing over font files is a licensing act.

- **A low-resolution reference PDF or JPEG** marked `REFERENCE — DO NOT PRINT`, which is genuinely useful on a complex job so the operator can see what it should look like.
- **A separate die-line PDF** if the finishing is complicated, even though it is also a layer in the main file.

The last look

Before it goes, open your own PDF as if a stranger sent it to you.

1. Every page present, right order, right orientation.
2. TrimBox correct, bleed on all four sides, nothing important within 4 mm of trim.
3. Zoom to 400% on small black text; it should be 100K.
4. Overprint preview on; nothing has vanished.
5. Read every phone number, price and name out loud.
6. Check the page count is what the binding needs.

Five minutes. This list, run honestly, is the difference between a career of smooth jobs and a career of awkward phone calls — and every item on it is something that has gone to press wrong for somebody this week.

When the shop comes back with a query

Answer the question they asked, quickly, and do not be defensive. Shops query files all day; a query is not an accusation. If they are wrong, say so politely with the evidence — "`pdfinfo -box` shows the TrimBox at 90 × 54, here is the output" — and if they are right, thank them, because they have just saved you a reprint.

The people who run the machines see more print in a month than most designers see in a year. Treat their queries as free expertise, because that is what they are.

Keep the sent file separate

Once a file has been sent, do not overwrite it. Move it into a `sent/` folder inside the job and make the next export `v4`. The moment a press PDF is edited in place, you can no longer answer the question in the proofs lesson — which file did they print — and that question arrives without warning, months later, usually by phone.

BEFORE YOU MOVE ON

A designer emails a press PDF with the message "final file attached, please print 1000". The shop prints it on 250 gsm gloss rather than the 350 gsm matt card that was discussed on the phone two weeks earlier. Where does the failure actually sit?

A. With the shop, since a verbal specification agreed on a call is binding and should have been recorded on their job docket.

B. With the PDF, which should have carried the stock specification in its document metadata for the shop's workflow to read.

C. With the designer, for not restating the specification in writing with the file.

D. With the client, who should have confirmed the stock directly with the printer before the file was sent.

CASE STUDY · MODULE 7

Six at night, plates at eight, and a die-line set to knock out

A tea packer in Colombo: three thousand cartons, a trade fair on the Thursday, and a preflight run at the last minute.

The job was a folding carton for a single-estate tea, three thousand pieces, printed four colour on 260 gsm uncoated board with a die-cut window and a spot varnish. The trade fair the cartons were for opened on Thursday morning. The printer's platemaker starts his run at eight in the evening to hold that delivery.

The designer exported the press PDF at ten past six and ran the five-line preflight script he keeps on his machine. It takes about thirty seconds. He had been sending files to this shop for two years and had never once had a file rejected, which is the state of mind that makes people skip the check.

Four things came back.

The boxes were right: pdfinfo reported a TrimBox at the agreed carton size with the BleedBox 3 mm outside it on every edge.

The fonts were all embedded, every row showing yes, though one carried an unfamiliar name with an axis value appended, which is what an instanced variable font looks like in a listing.

Ink coverage was not right. Ghostscript's inkcov reported a page total of 341 per cent. The printer's stated limit on that uncoated board is 300. The excess came from a dark photograph of the estate at dawn, which had been converted with a coated profile because that was the last setting the document had been given.

And the separations were wrong in a way that would have been invisible on screen. Running tiffsep produced five files where four were expected: cyan, magenta, yellow, black and a spot named Dieline, which is correct, because the die-line should travel as its own separation. But the die-line objects were set to knock out rather than to overprint, which means the cut path would punch a white outline straight through the artwork it sits on. In a normal render of the PDF it looks like a thin line on top of the picture. On the sheet it is a hole.

It was now twenty-five past six.

The die-line fault is certain: it will print, on every one of the three thousand cartons. The ink coverage is a judgement: 341 per cent on uncoated board may dry given time, or may set off onto the sheet above it in the stack, and the job is going straight into a finishing line.

The options were the obvious three.

Send it and add a note asking the RIP operator to force the Dieline separation to overprint. Many RIPs can. It puts the fix in somebody else's hands, at eight at night, on a job he has not seen.

Hold the plates and fix both faults properly: convert the photograph with the uncoated profile and set the die-line objects to overprint, re-export, re-preflight. Perhaps forty minutes of work, but the platemaker would then be working past ten, which is overtime the shop charges for, and the two-hour slip eats the float before Thursday.

Fix only the die-line and let the coverage ride, on the grounds that one is certain and the other is a risk.

> **STOP HERE**
>
> Before turning the page: what would you have done, and what would it have cost if you were wrong? Write it down. The point of the next section is to compare.

WHAT HAPPENED

He telephoned the shop rather than deciding alone, which took four minutes and changed the answer.

The platemaker said two things. He could force the die-line to overprint in the RIP, but he would rather not, because he had once forced it on a job where the same setting also overprinted a white panel and the panel vanished. And 341 per cent on that board, going into a finishing line the next morning, would not dry; he had seen the same photograph problem twice that month, both times from designers converting with a coated profile out of habit.

They agreed to hold. The conversion with the correct uncoated profile brought the page total to 287 per cent. Setting the die-line objects to overprint took two minutes. The re-export and the second preflight ran clean, and the file was with the shop at five past nine. Plates were made by a quarter past nine and the overtime came to less than a quarter of a set of plates.

The cartons reached the fair on Wednesday evening.

A month later, on a different job, the same shop queried a file and asked whether he had sent the version with the corrected spine. He had the archived PDF, its checksum and the preflight output saved beside it, and the question was answered in a minute rather than becoming a conversation about who remembered what.

The PDF looked correct in a normal viewer. Why is that not evidence, and what two things must be asked for explicitly?

A viewer renders with its own engine, on screen, with anti-aliasing and usually with overprint simulation switched off, and it shows a composite. The two things it hides are overprint, which is what made the knocked-out die-line look like a harmless line, and separations, which is the only view that shows plate count, whether black text is really 100 per cent K and whether a spot survived. Both are available free: Ghostscript can simulate overprint and write one file per plate.

He corrected the ink coverage by converting with the right profile rather than by lightening the photograph. Why does the distinction matter?

The RIP applies the press's own tone reproduction curve, so the correction for how that press behaves already happens downstream. Lightening the artwork by hand corrects a second time, and the result prints washed out. It also travels: the lightened photograph goes into the next job and carries an adjustment that belonged to one press on one board. Converting to the correct destination profile fixes the cause and leaves the artwork reusable.

Which of the four artefacts earned its keep a month later, and what does that suggest about when to create them?

The archived PDF with its checksum, because the question was which file was printed, and a checksum turns an assertion into a fact. The saved preflight output helped too, because it recorded what was true about that file at the time it was sent. Both are worth creating at the moment the file leaves, when they cost seconds, because the question they answer always arrives later, without warning, and usually by telephone.

MODULE 7 · TEST

Check what stuck

6 questions, one right answer each. This one is for you, not for a record: answers and the reason behind each are at the back. If two questions from the same module go wrong, re-read that module before the exam rather than after.

1 A job times out on the press's RIP. Which of these is by far the likeliest cause?

A. The file was exported as PDF/X-4 rather than PDF/X-1a
B. A traced logo carrying tens of thousands of nodes
C. Crop marks placed inside the bleed area rather than outside it
D. A spot colour that was defined in the swatches and never used

2 What does PDF/X-1a forbid that PDF/X-4 permits?

A. Embedded fonts and a defined TrimBox
B. Spot colours and a named output intent
C. Crop marks and bleed beyond the trim
D. RGB content and live transparency

3 Which box does the press cut from, and what is the usual way of getting it wrong?

A. MediaBox; people set the page size to trim instead of to trim plus bleed
B. BleedBox; people forget to offset the crop marks past the bleed
C. TrimBox; people export a page at trim plus bleed with no boxes set
D. CropBox; people leave the viewer's display box at the sheet size

4 You open the cyan separation of a page and can read the body text in it. What does that mean?

A. The text is 100 per cent K, and the cyan plate is showing it as a ghost image
B. The cyan plate has been trapped into the text to strengthen it
C. The separation was rendered as a composite rather than as a plate
D. The text is a four-colour build and will fringe at the slightest drift

5 Why does a rich black of 80C 70M 70Y 100K cause trouble on uncoated stock?

A. It totals 320 per cent, above the limit, so it will not dry
B. It exceeds the gamut, so the black clips back to a dark grey
C. Four plates cannot register, so every edge of it will fringe
D. Uncoated dot gain means it reads as more than 400 per cent

6 The shop runs CorelDRAW X7 and you work in CorelDRAW 2024. What is the reliable answer?

A. Export EPS, which every version of every application can open
B. Send the CDR, because Corel files are backward compatible
C. Send a press-ready PDF, and save down only if they must edit
D. Send SVG, which is an open standard that any version imports

MODULE 8

Machines that cut, stitch and burn

The paid work almost nobody teaches. Preparing a file for a physical machine rather than for a sheet of paper: vinyl plotters and weeding, print-and-cut registration, embroidery stitch files, laser kerf and material safety, CNC tool radius, screen-printing separations and underbase, the newer garment processes, and large-format signage at real viewing distances.

> BY THE END OF THIS MODULE YOU CAN
>
> Prepare a vector file for a plotter, needle, laser, router, screen or large-format printer — stating the minimum feature the material can hold, the kerf, pull or choke the process introduces, and the file format, layer convention and spot-colour name the operator's software expects.

65 · 10 min

The paid vector work is mostly machines, not posters

THE ONE THING TO KEEP

For any machine — plotter, needle, laser, screen — the file must describe a physical action, which means closed paths, outlined strokes, welded overlaps and a minimum feature size the material can actually hold.

Where the money actually is

Designers imagine vector work means logos and posters. A great deal of the paid work is preparing files for machines: a plotter that cuts vinyl, a needle that stitches thread, a laser that burns a line, a screen that pushes ink. The mark is the same, the constraints are completely different, and almost nobody teaches them — which is why the work pays.

The through-line: **the file tells the machine what to physically do.** Not what to look like.

Cutting plotters and vinyl

A plotter drags a blade along your paths: signage, vehicle graphics, shop shutters, stickers. Roland and Graphtec at the top, and many cheaper machines running ArtCut, SignMaster or FlexiSign. What that means for the file:

- **No fills, no gradients, no images.** The machine cuts outlines; colour comes from the vinyl you load.
- **Strokes must be converted to outline objects.** A stroke is a rendering instruction, not a shape. The plotter follows the path's centreline, so a 0.5 pt outline around a letter is cut as a single line through the middle of it. Convert first: **Ctrl+Shift+Q** in CorelDRAW, Object > Path > Outline Stroke in Illustrator, **Ctrl+Alt+C** in Inkscape.

- **Overlapping shapes must be welded.** Two overlapping letters are two closed paths, so the blade cuts the buried edges too and script lettering falls apart.
- **Weeding sets the minimum size.** A person picks the waste vinyl out by hand with a hook, and anything under about 2 mm tears while weeded. Hairline serifs and tiny counters are not a stylistic risk here, they are physically impossible.
- **Counters come out loose.** The hole in an "a" is a separate piece held in place by the application tape, not by the design, so small counters get lost.

Free path: Inkscape exports SVG and DXF, which most cutting software imports, and **Inkcut**, a free extension, drives many plotters directly.

Embroidery digitising

Embroidery does not use vectors at all. The machine reads a stitch file — DST (Tajima), PES (Brother), EXP (Melco), JEF (Janome) — storing needle positions, stitch types, jumps, trims and colour stops. Your vector is a reference the digitiser works over, deciding stitch direction, density, underlay and pull compensation. It is a skilled trade of its own, and units in Surat, Tiruppur and Ludhiana do it all day. What it means for your design:

- Detail below about **1.5 mm** disappears into the thread, and text below **5 mm cap height** turns to a smudge.
- **No gradients.** Colours become a short list of thread codes from one brand's chart.
- Fabric **pulls** as the needle works, so the design distorts and the digitiser widens shapes slightly across the pull direction. Knitwear pulls more than woven cloth.
- Every colour change is a machine stop, so five colours cost more than two.

Free tool: **Ink/Stitch**, an Inkscape extension that genuinely produces usable DST and PES files. It will not replace a good digitiser on a complex chest logo, but for a simple one- or two-colour mark it works.

Laser and CNC

Vector paths become toolpaths. A common laser convention is a hairline red stroke for cut and black fill for engrave, but it varies by machine and driver, so **ask the operator rather than trusting any rule you read**, including this one.

Two things are always true:

- **A cut path must be closed.** An open path leaves the piece attached.
- **Kerf.** The beam or bit removes material as it travels — roughly 0.1–0.3 mm for a laser, the full bit diameter for a CNC router — so an interior hole comes out that much larger and a tab that much smaller. If two pieces must slot together, design the clearance in rather than drawing them touching exactly.

Free path: Inkscape plus Gcodetools or another CAM extension. **LightBurn** is paid, cheap and one-off, and what most small laser shops run.

Screen printing separations

Each colour is a separate screen, so the design is built as a few flat colours from the start. That constraint is why so much good T-shirt and poster art looks the way it does.

- **Separations**: one plate per colour. For flat spot colours, put each colour on its own layer in Inkscape and export the layers.
- **Halftones** for tonal areas, at a line screen the mesh can hold — ask the printer, a coarse mesh cannot hold a fine screen.
- **Underbase** on dark garments: a white layer printed first so the colours read on top. It must be **choked**, shrunk slightly, so it does not peek out around the edges.
- **Registration marks** on every separation, and **trapping** — adjacent colours overlapped slightly so misregistration shows an overlap, not a gap.

Simulated-process separations are usually done in Photoshop with channels. **Photopea** does the same in a browser, free, on a laptop that could never run Photoshop.

Large format and signage

Vector all the way to the printer, and the RIP handles resolution. Check minimum text size at the real viewing distance, and ask the question people forget: **hoardings are usually printed in panels and joined.** Find out where the seams fall and keep faces, text and the logo off them.

The habits that make all five work

Before any file goes to a machine:

1. All strokes converted to outlines, all text converted to curves
2. All overlapping shapes welded, all cut paths closed
3. No stray objects outside the artboard, no hidden layers, no zero-length paths
4. Node count sane — run a simplify pass over any traced artwork
5. Units and finished size in the filename: `shutter-sign-1200x600mm-CUT.svg`
6. A phone call to the operator before you send

That last one is not padding. The person running the machine knows their material, their kerf, their mesh and their limits, and five minutes with them prevents most reprints. They are also, in this trade, the people who hand out the next job.

BEFORE YOU MOVE ON

Artwork goes to a vinyl cutter: text with a 0.5 pt outline stroke, and several overlapping letters. The blade cuts through the middle of the letters, and the thin outlines tear during weeding. What went wrong at the file stage?

A. The file was in CMYK, and a cutter needs a single named spot colour to know which contour to follow.
B. The strokes were never converted to outline objects and the overlapping letters were never welded, so the plotter followed each stroke's centreline and the buried edges of the overlaps as real paths — and the resulting hairline shapes were too thin to weed.
C. The artwork was too low in resolution for the cutter's tracking and should have been exported at 300 dpi.
D. The vinyl was the wrong weight for the design, so the file was fine and the material choice caused the tearing.

66 · 8 min

A vinyl job from file to wall

THE ONE THING TO KEEP

A plotter cuts outlines, so text must be curves, strokes must be outlined and overlaps welded — and the real constraint is weeding, which sets a minimum feature of about 2 mm and makes every counter a separate piece held only by the transfer tape.

What the machine is

A cutting plotter holds a small blade in a swivelling holder and drags it over material on a roll while the roll moves back and forth under it. It does not print. It cuts outlines, and the colour comes from the vinyl you loaded.

Roland and Graphtec are the machines most signage shops aspire to; a large number of small shops run cheaper Chinese plotters with ArtCut, SignMaster or FlexiSign. Cricut and Silhouette are the hobby end of exactly the same technology.

The materials, because they decide the design

- **Cast vinyl** — thin, conformable, 5–10 year outdoor life. For vehicles and curved surfaces. Expensive.
- **Calendered vinyl** — thicker, cheaper, 3–5 years, shrinks a little over time. Flat surfaces: shop boards, shutters, windows.
- **Reflective** — for road and safety signage. Thick and hard to weed.
- **Heat transfer vinyl (HTV)** — cut, weeded, then heat-pressed onto fabric. **Must be cut mirrored**, because it is applied face down.
- **Etch or frosted film**, **one-way vision** perforated film for vehicle windows, **floor graphics** with a textured laminate.

Preparing the file

Five steps, in this order, and skipping one costs a sheet of vinyl:

Preparing a file for the blade

In this order, and skipping one costs a sheet of vinyl. The plotter cuts what is in the file rather than what is on screen, so a hidden layer and a stray zero-length path are both cut.

1. **Convert all text to curves.** The shop's software will not have your font.
2. **Convert all strokes to outlines.** The blade follows the path. A stroked letter is cut as a single line down the middle of the outline, which is the commonest beginner failure.
3. **Weld all overlaps.** Two overlapping script letters are two closed paths, so the blade cuts the buried edges too and the join falls apart. Union everything that visually touches.

4. **Remove everything invisible.** Stray points, zero-length paths, objects outside the artboard, hidden layers. The plotter cuts what is in the file, not what is on screen.
5. **Set the real size.** Build at 1:1 if the software allows. Put the size in the filename.

Then check the node count and simplify any traced artwork. The plotter decelerates at every node; a 4,000-node letter takes minutes instead of seconds and comes out rougher.

Weeding sets the minimum feature size

After cutting, a person picks out the waste vinyl by hand with a hooked tool. This is **weeding**, and it is where design meets physical reality.

- Anything narrower than about **2 mm** tears while being weeded.
- Fine serifs, thin script connectors and small counters are not a stylistic risk, they are physically impossible.
- The counter of an `a` or the middle of an `o` comes out as a **separate piece**, held in place only by the transfer tape. Below about 5 mm across, they get lost.
- A design with hundreds of small holes takes an hour to weed, and that hour is on the quotation.

For a complex job, add **weed lines** — extra cut lines through the waste, dividing it into manageable sections — and **weed boxes** around groups of letters.

Application

Worth understanding even if somebody else does it, because it changes the file.

1. Cut, then weed.
2. Lay **application tape** (a low-tack transfer paper) over the design and squeegee it down.
3. Peel the design away from its backing; it comes up on the tape.

4. Position on the surface — wet application with soapy water gives you repositioning time on large pieces.
5. Squeegee from the centre out, then peel the tape back at a sharp angle.

Which is why the counters matter: the tape holds them, and the design must keep them large enough to survive.

The free path

- **Inkscape** exports SVG and DXF; most cutting software imports both. Set the document to millimetres at 1:1.
- **Inkcut**, a free Inkscape extension, drives many plotters directly over USB or serial, handling HPGL output and weed boxes.
- **LibreCut** and `vpype` are other free options for plotter workflow.
- Hobby machines (Cricut, Silhouette) mostly need their own software, which will import SVG.

The question to ask before you draw

> *What is the smallest stroke you can weed in this material, and what is the sheet width?*

Sheet width decides whether a 1.4 m sign is one piece or three with a join, and the smallest weedable stroke decides your typeface. Both are one sentence, and both change the design.

BEFORE YOU MOVE ON

A shop cuts a script wordmark in vinyl. The letters were drawn with a 3 mm stroke on a single connected path and the file was sent without further preparation. What comes off the machine?

A. A single cut line down the centre of each letterform, because the blade follows the path.
B. The letters cut correctly, but with the overlapping joins cut through, so the script separates into pieces.
C. Nothing, because plotter software rejects files containing strokes rather than closed paths.
D. Each letter cut twice over, once along each side of the stroke, leaving nothing but a thin ribbon of vinyl.

67 · 8 min

Print and cut, and the marks the machine looks for

THE ONE THING TO KEEP

Print-and-cut aligns the two machines with optical registration marks the RIP places, and the cut line travels as an outline on a named spot colour such as `CutContour`, set to overprint so it is passed to the cutter rather than printed.

Two machines, one file

A **print-and-cut** job is printed on a large-format printer onto adhesive vinyl, then moved to a cutting plotter which cuts around the printed artwork. Stickers, labels, decals, shaped window graphics, floor markers.

The problem is alignment. The plotter has no idea where the print landed on the roll, and a 1 mm error around a printed sticker is obvious.

The solution is **registration marks**: small printed targets that the plotter finds with an optical sensor before it cuts, using them to calculate the print's position, rotation and any stretch.

How the marks work

The printer prints three or four marks — usually small squares, crosses or right-angle corners — at known positions around the artwork. The plotter's sensor is driven over each in turn, detects the edge, and builds a transform.

- **Three marks** give position and rotation.
- **Four marks** additionally correct for the material stretching along the roll, which it does.
- **Segmented marks** along a long job let the machine re-register periodically, so a 3 m banner does not drift at the far end.

The marks must be: the shape and size the machine expects, at the distance it expects from the artwork, printed in solid black on an unprinted white area, with a **quiet zone** of bare white around each. Artwork touching a mark, or a coloured background under it, stops the sensor finding it.

The mark specification comes from the machine. Roland, Graphtec, Mimaki and Summa all differ. Your cutting software — VersaWorks, CutStudio, FlexiSign, Onyx — places them correctly. Never draw them by hand.

The cut line is a spot colour

This is the convention that matters, and it is the same across the industry:

Draw the cut path as an outline with no fill, colour it with a spot swatch named `CutContour`, and set it to overprint.

The RIP recognises the name, does not print it, and passes the geometry to the cutter. Other names are used — `Thru-cut`, `Kiss-cut`, `CutLine`, `Crease`, `Perf` — and **the shop will tell you which** because it is configured in their RIP.

In Illustrator: Swatches ▸ New ▸ Colour Type: Spot, name it exactly `CutContour`, and use 100% magenta as its appearance so you can see it. In CorelDRAW: define a spot colour with the same name. In Scribus: Edit ▸ Colours ▸ New ▸ tick Spot Colour. Inkscape cannot make a true spot, which is why print-and-cut work is one of the places the free chain needs Scribus for the export.

Bleed for a contour cut

A contour cut is not a guillotine cut, and it still needs bleed. The plotter is accurate to a few tenths of a millimetre, plus whatever the registration leaves.

- **1–2 mm of bleed** beyond the cut line, on all sides.
- Keep text and important detail **at least 3 mm inside** the cut line.
- Round the corners of the cut path rather than leaving sharp points — a sharp corner lifts at the edge and the sticker peels.

Blade offset, and why corners come out rounded

The blade is not at the centre of its holder; it trails behind, swivelling to follow the direction of travel. That distance is the **blade offset**, typically 0.25–0.5 mm.

The machine compensates by overcutting slightly at corners. If the offset is set wrong for the blade fitted:

- **Too small** — corners come out rounded, because the blade has not finished turning.
- **Too large** — corners overshoot, leaving small tails past each corner.

This is the operator's calibration, not your file, but recognising the symptom tells you whose problem it is. Every machine has a test-cut pattern for setting it.

Kiss cut and through cut

- **Kiss cut** — through the vinyl, not the backing. Stickers on a sheet.

- **Through cut** — through everything. Individual die-cut stickers.

A sheet of stickers usually wants both: kiss-cut shapes and a through-cut around the whole sheet, on two differently named spot layers. Say which is which in the covering note.

Before you send

1. Cut path on its own layer, named spot colour, no fill, overprint on.
2. Artwork bleeds 2 mm past the cut path everywhere.
3. Nothing important within 3 mm of the cut.
4. Corners of the cut path rounded.
5. Registration marks left to the RIP.
6. The finished size and the quantity per sheet in the filename.

BEFORE YOU MOVE ON

A sheet of printed stickers comes back with every cut about 1.5 mm off, consistently shifted in the same direction. The registration marks printed cleanly and the plotter found all four. What does the consistent direction of the error point to?

A. The blade offset is set wrong for the blade that is fitted, which would show as rounded or overshooting corners rather than a uniform shift.
B. The material stretched along the roll, which four-mark registration exists specifically to correct.
C. The cut path was drawn with a stroke rather than as a fill-less outline, so the plotter followed one edge of it.
D. A fixed offset between the sensor's reading position and the blade's, which is a machine calibration.

68 · 9 min

Embroidery, which does not use vectors at all

THE ONE THING TO KEEP

Embroidery reads a stitch file, not vectors, so a digitiser must choose stitch type, direction, density and underlay — and pull compensation means a shape comes out narrower across the stitch direction and longer along it unless the artwork is deliberately widened.

The file is a list of needle positions

This is the single most misunderstood thing in the trade. An embroidery machine does not read vectors. It reads a **stitch file** — DST (Tajima), PES (Brother), EXP (Melco), JEF (Janome), VP3 (Husqvarna) — which stores a sequence of needle coordinates, jumps, trims and colour stops.

Your vector artwork is a **reference**. Somebody, or some software, must decide stitch type, direction, density, underlay and compensation for every area. That process is **digitising**, and it is a skilled trade employing a great many people in Surat, Tiruppur, Ludhiana and Karur.

The three stitch types

- **Run stitch** — a single line of stitches. Outlines, fine detail, stems.
- **Satin stitch** — zigzag back and forth across a shape, giving the glossy raised look. Used for lettering and borders. Works between about **1.2 mm and 12 mm** wide. Narrower than 1.2 mm and the needle perforates the fabric into a line; wider than 12 mm and the threads snag and sag, so a wide area is split or filled instead.
- **Fill stitch (tatami)** — rows of run stitches covering a large area, with a stitch direction and a pattern of offsets to stop a visible line forming.

Direction is a design decision. A fill's angle changes how light catches it, and a good digitiser varies the angle between adjacent areas so they read as separate shapes without an outline.

Underlay, density and pull

Three parameters that make the difference between a clean badge and a puckered mess.

Underlay is a first, sparse layer of stitching that stabilises the fabric and gives the top stitches something to sit on. Without it, satin stitches sink into the pile of a towel or a fleece and disappear.

Density is the spacing between stitches, typically 0.35–0.45 mm for a satin. Too loose and the fabric shows through; too dense and you get a stiff plate that perforates the cloth and breaks needles.

Pull compensation is the one nobody expects. As the needle pulls thread through, the fabric is drawn inwards across the stitch direction. A shape therefore comes out **narrower** than drawn, and the digitiser widens it — typically 0.2–0.3 mm on each side — to compensate. Knitwear pulls far more than woven cloth, so the same design needs different compensation on a polo shirt and on a cap.

Push is the complement: the shape grows slightly along the stitch direction. A circle digitised without compensation comes out as an oval.

What this means for your design

- **Nothing below about 1.5 mm** survives. It disappears into the thread.
- **Text below about 5 mm cap height** becomes a smudge. For lowercase, 6 mm.
- **No gradients, no tints.** Colours become a short list of thread codes from one brand's chart — Madeira, Isacord, Robison-Anton. Ask which the shop uses and pick from their chart.
- **Every colour change is a machine stop.** Five colours cost more than two, in time and in thread changes, on every single garment.
- **Stitch count is the price.** A shop quotes per thousand stitches. A detailed chest logo at 12,000 stitches costs roughly twice a simple one at

6,000, on every unit.

- **Small text inside a fill** needs to be a gap in the fill, not stitched on top, or it fills in.

Doing it yourself, free

Ink/Stitch is an Inkscape extension that genuinely produces usable DST, PES and other formats. It is free, actively developed, and it will take you from a vector shape to a stitch file with satin columns, fills, underlay and manual stitch direction.

The workflow: draw the shape in Inkscape, use Ink/Stitch's *Params* dialog to set stitch type and density per object, use the satin-column tools to control direction, simulate the result, then export. It has a real learning curve and it is a complete tool.

It will not replace a good digitiser on a complex chest logo, where experience with a specific fabric is what you are paying for. For a simple one- or two-colour mark, it works.

The test that settles everything

Ask for a sew-out. A single stitched sample on the actual fabric, before the run. Most shops will do one for a small fee or free on a decent order, and it shows you pull, density, colour and legibility in one object.

Nothing on a screen predicts embroidery. The sew-out is the proof, and skipping it on a 500-shirt order is the most expensive economy in this module.

BEFORE YOU MOVE ON

A circular logo is digitised without pull compensation and sewn onto a knitted polo shirt. What comes out?

A. A circle with ragged edges, because knit fabric cannot hold a smooth curve without underlay.
B. A circle of the correct shape but visibly smaller overall, because the thread tension shrinks the whole design uniformly.
C. A correct circle, since pull compensation only matters for satin columns and not for fills.
D. An oval, narrowed across the stitch direction by pull and lengthened along it by push.

69 · 8 min

Laser cutting, kerf, and the material you must never use

THE ONE THING TO KEEP

A laser distinguishes cut from engrave by stroke and colour on named layers, removes a kerf of 0.1–0.3 mm that must be measured on the real material before parts will fit, and must never be used on PVC or any other chlorine-bearing plastic.

Two operations, one machine

A laser cutter does two different things and your file must say which is which.

- **Vector cut** — the head follows a path at speed, the beam cutting through. Driven by paths.
- **Raster engrave** — the head sweeps back and forth like an inkjet, pulsing the beam to burn a surface image. Driven by a bitmap, generated from your fill.

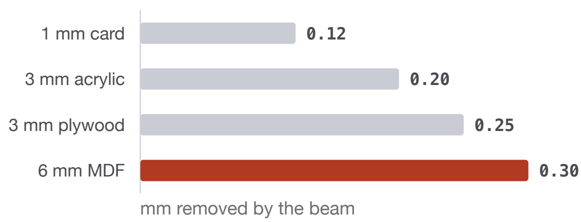
Most machines distinguish them by **colour convention**: a hairline stroke of one colour means cut, another means score, and black fill means engrave. **The convention varies by machine and driver**, so ask the operator rather than trusting any rule you read, including this one. What is universal is that the distinction is made by colour and stroke, so keep them on separate, clearly named layers.

Kerf, and why the parts do not fit

The beam removes material as it travels. That removed width is the **kerf**, typically **0.1–0.3 mm** depending on material, thickness and focus.

The consequence:

Kerf, measured on the material itself



Cut a 20 mm square from a scrap, measure the square and the hole with callipers, and half the difference is your kerf. A hole comes out one kerf larger and the part one kerf smaller, so two pieces drawn to slot together exactly are loose by two.

- A cut-out hole comes out **larger** than drawn, by one kerf.
- A cut-out part comes out **smaller** than drawn, by one kerf.
- So two parts drawn to slot together exactly will be loose by two kerfs — about 0.4 mm, which is a wobbly joint.

Measure it rather than guessing. The standard test: cut a 20 mm square from a scrap of the actual material, measure the square and the hole with callipers, and the difference divided by two is your kerf.

```
hole measured  20.22 mm
part measured  19.78 mm
difference      0.44 mm  →  kerf 0.22 mm
```

Then offset your paths by half a kerf in the appropriate direction for a press fit, or design the clearance in deliberately.

Closed paths, always

A cut path must be **closed**. An open path leaves the piece attached by whatever gap remains, and it has to be snapped or cut free by hand, leaving a mark. This is the same discipline as the plotter: weld overlaps, close every loop, remove duplicates.

Duplicate paths matter more here than anywhere. Two identical paths stacked means the laser cuts the same line twice, which on thin material burns a wide, charred channel.

Material safety, which is not optional

Never cut PVC, vinyl, or anything containing chlorine. Lasering PVC releases **hydrogen chloride gas**, which corrodes the machine's optics and rails and is seriously harmful to breathe. Sign vinyl is PVC. This is the single most important fact in this lesson, and reputable shops refuse the material outright.

Also refuse or check carefully:

- **Polycarbonate** — burns rather than cutting, yellows the edge, smokes badly.
- **ABS** — melts and produces cyanide compounds.
- **Fibreglass and carbon fibre** — resin fumes and abrasive dust.
- **Anything you cannot identify.** If the client cannot say what a sheet is made of, do not put it in a laser.

Safe and common: **plywood and MDF** (both give a charred edge, which is a look), **acrylic / PMMA** (cuts beautifully with a polished edge; cast acrylic engraves frosted white, extruded does not), **paper and card**, **leather** (vegetable tanned, not chrome), **fabric**, **cork**, **anodised aluminium** for marking.

Living hinges and the things a laser makes possible

A **living hinge** is a pattern of interlaced slots cut into rigid sheet that makes it bend. The spacing, slot length and gap determine the bend radius, and it is worth cutting a test strip in the actual material rather than copying numbers.

Similarly, **inlay** work, **stacked layers**, **engrave-then-cut** sequences — all are just ordering operations correctly in the file and on the machine.

The software

- **LightBurn** is what most small shops run. Paid, inexpensive, one-off licence, and genuinely good: layer-based, per-layer power and speed, kerf offset built in, and it drives most common controllers.
- **Inkscape plus a CAM extension** — Gcodetools, or the several laser plugins — is the free route and works, with more friction.
- **The machine's own driver**, often a printer driver where you "print" from any application, using colours for operations.

Send SVG, DXF or PDF at 1:1 in millimetres, with layers named `CUT`, `SCORE` and `ENGRAVE`, and put the material and thickness in the filename.

BEFORE YOU MOVE ON

A designer draws two plywood parts to slot together, with a 6 mm tab and a 6 mm slot, and sends the file for laser cutting. The joint comes out loose. What is the arithmetic?

A. The tab cuts 6 mm minus one kerf and the slot 6 mm plus one, so the gap is two kerfs.
B. Plywood expands when heated by the beam, so both features come out oversized and the fit is lost.
C. The kerf applies only to the outer cut, so the slot is correct and the tab is undersized by one kerf.
D. The file was drawn in inches and then converted, introducing a rounding error of about 0.4 mm on a 6 mm feature.

70 · 8 min

CNC routing, tool radius, and the letters on a shop board

THE ONE THING TO KEEP

A router bit of finite diameter cannot cut an internal corner sharper than its own radius, so ask the bit diameter before drawing and add dog-bone fillets where a square corner is needed — and built-up or channel letters impose a minimum stroke width and make every counter a separate piece.

A router is a spinning cylinder

Where a laser has a beam of near-zero width, a CNC router has a **bit of finite diameter** — commonly 3 mm or 6 mm for sign work. Everything that follows comes from that one fact.

The internal corner problem

A router bit cannot cut an internal corner sharper than its own radius. Draw a square hole and cut it with a 6 mm bit and you get a square hole with 3 mm radiused corners.

Three responses:

1. **Design the radius in.** If the corner is going to be round, draw it round so the result is what you intended.
2. **Use a smaller bit** for the detail, in a second pass. Slower, and a small bit in hard material breaks.
3. **Dog-bone or T-bone fillets.** Add a small circular overcut at each internal corner, so a square tab still fits. This is standard practice in CNC joinery and looks deliberate once you know what it is.

The same limit governs letterforms: a serif's inner angle, the junction of a stem and a bowl, the point of a `V`. **Ask the fabricator the bit diameter**

before you draw, and check every internal angle in the mark against it.

Tabs, and the part that flies across the workshop

A part cut fully free in the middle of a sheet will move once it is loose, and the bit will hit it. So the toolpath leaves small **tabs** — uncut bridges of 3–6 mm — holding the part in place, which are cut through by hand afterwards and sanded.

Tab placement is a decision: put them on edges that will not be seen, and never on a visible curve where the witness mark will show.

Climb and conventional, briefly

The bit can rotate into the material (**conventional**) or with the direction of travel (**climb**). Climb milling gives a better surface finish on most sign materials and pulls the workpiece, so it needs good hold-down. This is the operator's choice; you need it only to understand why they may ask which face is the visible one.

Materials for signage

- **ACP / ACM (aluminium composite panel)** — two thin aluminium skins on a plastic core. The standard shop-board material: light, flat, weatherproof, cuts and folds well. Brand names like Alucobond and Aludecor.
- **Acrylic** — for illuminated letters and panels. Cast acrylic routs better than extruded.
- **MDF** — interior only; it swells with any moisture.
- **Foam PVC (Forex, Sintra)** — light, cheap, interior and sheltered exterior.
- **Aluminium and brass** — for engraved plates and premium letters.

Built-up and channel letters

A shop-board name in three-dimensional letters is built rather than cut flat.

- **Flat-cut letters** — routed from sheet, mounted on studs standing a few millimetres off the wall so they cast a shadow.
- **Built-up letters** — a routed face, a returned side (the "return"), and a back. Effectively a shallow box in the letter's shape.
- **Channel letters** — built-up letters with LEDs inside and a translucent acrylic face. **Face-lit** glows from the front; **halo-lit** or reverse-lit has an opaque face and throws light onto the wall behind, which is the expensive, good-looking one.

What each demands of your artwork:

- **Counters need connections.** The middle of an `O` or an `A` is a separate piece. For a flat-cut letter it is mounted separately; for a channel letter it must either be its own small channel or be bridged.
- **Stroke width sets the return depth's feasibility.** A stroke thinner than about 25 mm cannot hold LEDs and a return.
- **Every letter is a separate piece.** A script with connected letters must be split at sensible points, and the split lines are a design decision you should make rather than the fabricator.
- **Mounting.** Studs, a hidden rail, or a raceway. Ask which, because a race-way is a visible box behind the letters and it changes the design.

What to send

DXF is the lingua franca of CNC, and most shops also accept AI, EPS or PDF. Whatever the format:

- 1:1, in millimetres, one closed path per cut.
- Layers named for the operation: `CUT`, `POCKET`, `ENGRAVE`, `DRILL`.
- Strokes outlined, text curved, overlaps welded, duplicates removed.
- A separate reference PDF showing the assembly, the colours and the mounting.
- The material and thickness in the filename and in the note.

The conversation

> *What bit diameter, what material and thickness, and will these be flat-cut, built-up or channel letters?*

Three answers, and the design constraints follow from them. Asking after the design is finished is how a mark gets redrawn for free.

BEFORE YOU MOVE ON

A logotype with sharp internal angles is routed from ACP with a 6 mm bit. The fabricator returns a sample where every inside corner is visibly rounded. Whose problem is it, and what fixes it?

A. The fabricator's: a finishing pass with a smaller bit is entirely standard practice and should have been included in the quotation.
B. The fabricator's: the toolpath should have been offset inwards to compensate for the bit radius.
C. Neither: it is inherent to the process, and the artwork should carry the radius or dog-bone fillets.
D. The designer's: the file should have been supplied as DXF rather than PDF, since PDF does not preserve corner geometry.

71 · 9 min

Screen printing, separations and the underbase

THE ONE THING TO KEEP

Screen printing is one screen per colour, so build in flat colours on named layers, and on dark fabric add a white underbase choked inwards by 0.3–0.5 mm — with traps around 0.5 mm because manual carousel registration is measured in millimetres, not tenths.

One screen per colour

A screen-printing frame holds a mesh with a stencil. Ink is pushed through the open areas with a squeegee. One screen prints one colour, so a four-colour design means four screens, four setups, four registration alignments.

That constraint is why so much good T-shirt and poster art looks the way it does: flat colours, bold shapes, deliberate overprints. It is a design language that came out of a process.

Preparing spot separations

For a design in flat spot colours, which is most garment work:

1. **Build the artwork in flat colours only** — no gradients, no transparency, no soft edges.
2. **Put each colour on its own layer**, named for the ink: `RED`, `NAVY`, `WHITE`, `UNDERBASE`.
3. **Colour each layer's artwork with a named spot swatch.**
4. **Export one film per layer**, as pure black on transparent, at **1200 dpi** — films are output as solid black positives, and the stencil is made by exposing emulsion through them.

5. **Include registration marks on every film**, in the same position on each, so the screens can be aligned.

In Inkscape: build on layers, then export each layer separately (Layer ▸ Show/Hide, then export PNG at 1200 dpi, or use the *Export layers* extension). In Scribus or Illustrator, exporting separations from a spot-coloured file does the same job more directly.

The underbase, and the choke

On a dark garment, a colour printed directly disappears — the fabric shows through and changes the hue. So a layer of **white ink is printed first** as a base, flashed dry, and the colours printed on top of it.

That underbase is a copy of the whole design's coverage area, and it must be **choked**: shrunk inwards by **0.3–0.5 mm**, so that if the colour layers land slightly off, no white peeks out around the edges.

Make it by duplicating the combined artwork, applying an inward offset, and putting it on the `UNDERBASE` layer. Check the thin parts: from the offsets lesson, an inward offset eats thin features, so a fine line may vanish or self-intersect.

The underbase also costs: an extra screen, an extra pass, and a flash-dry cycle between. It is why printing on white garments is cheaper than on black.

Halftones on a garment

For tonal areas you need halftones, and the mesh sets the limit.

- Mesh counts of **43–61 threads/cm** (110–156 per inch) are common for garment work.
- A screen can hold roughly a **35–65 lpi** halftone. Finer than that and the dots fall through the mesh or clog it.
- **Angles** matter as much as in litho, to avoid moiré between the halftone and the mesh itself. Many printers use **22.5°** on every colour for garment work rather than the litho angle set, precisely because the mesh is a regular grid too.

- **Dot gain is large** — over 30% is normal on textile — so a 50% tint prints as a solid-looking mass unless you compensate.

Ask the printer for their mesh count and preferred line screen. These are not numbers you can choose.

Simulated process

For a photographic image on a dark shirt, **simulated process** separates the image into a set of flat spot inks — often six to eight — chosen for that image, rather than CMYK. It gives far better results on fabric than true process, and it is done with channels in Photoshop.

Free equivalent: **Photopea** in a browser has channels, curves and the selective colour tools needed, and runs on a laptop that could never load Photoshop. GIMP can do it with more effort. Either way, this is a specialist skill and it is worth knowing it exists and what it is called when a client asks for a photograph on a black shirt.

Trapping on a press with millimetre registration

From the colour module: trap width is a property of the process. On a manual carousel, registration can be out by a whole millimetre, so traps of **0.5 mm or more** are normal — vastly more than litho's 0.1 mm.

Ask the shop for their trap value and their underbase choke. They have numbers and they will give them to you.

The checklist before films

1. Flat colours only, one layer per ink, named for the ink.
2. Underbase built and choked, thin features checked.
3. Registration marks on every separation, identical position.
4. All text curved, all strokes outlined, all overlaps welded.
5. Films at 1200 dpi, solid black.
6. Colour count confirmed with the shop, because it is the price.

BEFORE YOU MOVE ON

A two-colour design on black T-shirts is printed with a white underbase built as an exact copy of the artwork's coverage. The shirts show a thin white halo around the coloured areas. What was omitted?

A. A flash-dry cycle between the underbase and the colours, which let the white bleed outwards under the wet colour layers.
B. Trapping between the two colours, which should have overlapped them to hide registration drift.
C. The underbase should have been printed last, on top, so any misregistration would be hidden beneath the colours.
D. The underbase was not choked: shrinking it inwards by 0.3–0.5 mm keeps it hidden.

72 · 8 min

DTF, DTG, sublimation and heat transfer

THE ONE THING TO KEEP

DTG is cotton-only with a machine-printed underbase, DTF works on almost anything but sits on the surface and needs clean hard-edged transparency, sublimation bonds into polyester and cannot print white or work on dark, and HTV is cut mirrored in flat colours.

Four newer processes, four different files

Screen printing is not the only way to get a design onto fabric, and the alternatives have different economics and different file requirements. Knowing which is which lets you advise a client rather than repeat what the last shop told them.

DTG — direct to garment

An inkjet printer that prints water-based ink directly into the fabric.

- **Cotton only**, or high-cotton blends. Ink does not bond to polyester.
- On dark garments, a **white underbase is printed by the machine** after a pre-treatment liquid is applied and heat-pressed. The pre-treatment is essential and shows as a faint square if applied unevenly.
- **No setup cost**, so a run of one is viable. Per-unit cost is high, so a run of 500 is not.
- **Full colour, photographic**, no colour limit.
- **File**: a high-resolution raster — 300 ppi at final size — with a genuine transparent background. Vector is fine as a source; the RIP rasterises it.

Soft hand-feel, excellent detail, limited to cotton, fades faster than screen print.

DTF — direct to film

Currently the process spreading fastest in small shops. The design is printed onto a PET film with white ink behind it, powdered with an adhesive, cured, and heat-pressed onto the garment.

- **Works on almost anything** — cotton, polyester, blends, nylon, leather, dark or light.
- **No pre-treatment**, no cutting, no weeding.
- **Full colour**, and the white is printed automatically as a backing layer.
- **File**: raster at 300 ppi with true transparency, or vector. The RIP generates the white layer from the alpha channel, so **the transparency must be clean** — a soft feathered edge produces a visible white fringe. Hard edges, or a deliberate 1 px feather at most.
- Sits **on top** of the fabric as a film, so the hand-feel is plasticky compared with screen or DTG, and a large solid area is noticeably stiff.

For short runs of mixed garments, it has largely displaced everything else.

Sublimation

Ink is printed onto transfer paper, then heat and pressure turn it into a gas that bonds into the polymer fibres.

Two hard constraints, and they are absolute:

- **Polyester only**, or a polymer-coated surface (mugs, phone cases, metal plates, coated MDF). It will not work on cotton at all.
- **White or very light substrate only.** There is no white sublimation ink, so the substrate provides the white and you cannot print light on dark.

Within those limits it is excellent: the ink is *in* the fibres, so there is no hand-feel at all, and it does not crack or peel. All-over printed sportswear is sublimation.

File: full-colour raster, and the image must be **mirrored** before printing, because it is applied face down.

Heat transfer vinyl (HTV)

Cut vinyl, weeded, heat-pressed. Covered in the vinyl lesson, and the two things to remember here are that it is **cut mirrored** and that it is limited to flat spot colours — which makes it ideal for names and numbers on sports kit and poor for anything detailed.

Choosing between them

- **Run of 1–20, full colour, mixed garment types** — DTF.
- **Run of 1–20, cotton, soft feel matters** — DTG.
- **Run of 100+, few colours** — screen print, comfortably cheapest per unit.
- **Polyester sportswear, all-over or full colour** — sublimation.
- **Names and numbers, any quantity** — HTV.
- **Premium feel, few colours, cotton** — screen print with a soft-hand or discharge ink.

Two things to tell every client

Ask for a sample before the run. The same artwork looks different in each of these processes, and the difference in hand-feel is something nobody can judge from a photograph.

Wash test it. A sample washed five times tells you more about the process than any specification sheet. DTF peeling at the corners, DTG fading, sublimation ghosting on a poly-cotton blend — all show up in the wash and not before.

What to supply, in one line each

- **DTG** — 300 ppi PNG at final size, transparent background, sRGB, cotton confirmed.
- **DTF** — 300 ppi PNG at final size, hard-edged transparency, no feathering beyond a pixel.
- **Sublimation** — 300 ppi at final size, mirrored, and confirm the substrate is polyester or polymer-coated.
- **HTV** — vector, mirrored, strokes outlined, overlaps welded, nothing under 2 mm.
- **Screen** — separations on named layers, 1200 dpi films, underbase choked.

Put the process in the filename. `acme-tee-DTF-260x200mm.png` tells the shop everything before they open it, and it stops a sublimation file being pressed onto a cotton shirt, which is a mistake somebody makes every week.

BEFORE YOU MOVE ON

A client wants twenty black polyester sports shirts with a full-colour photographic design. Which process fits, and why do two of the obvious candidates fail?

A. Sublimation, because polyester is exactly what it bonds into and twenty units needs no setup cost.
B. Screen printing with a choked white underbase, since it handles dark garments and any fabric type at all.
C. DTG, because it prints full colour with a white underbase on dark garments.
D. DTF, because it prints its own white backing, adheres to polyester and has no setup cost.

73 · 8 min

Signage, viewing distance and the seams

THE ONE THING TO KEEP

Large-format resolution is set by viewing distance — 20 to 40 ppi at full size for a hoarding seen from 30 m — and the constraints that actually decide the design are where the panel seams fall, the hem and eyelet allowance, and cap height of about 25 mm per 3 m of viewing distance.

The resolution question answered properly

"What resolution for a hoarding?" is the wrong question, and the answer people give — 300 dpi — is wrong by an order of magnitude.

What matters is the **angular** size of a detail at the viewing distance. The working figures used in the trade:

- Viewed at **1 m** (an exhibition panel, a poster): 150 ppi at final size.
- Viewed at **3–5 m** (a shop board, a pull-up banner): 100 ppi.

- Viewed at **10 m** (a building sign): 50 ppi.
- Viewed at **30 m or more** (a hoarding, a motorway sign): **20–40 ppi** at full size.

A 6 × 3 m hoarding at 30 ppi is 7087 × 3543 pixels. That is an ordinary camera file. The image everybody panics about is usually already sufficient.

The other way to say it: build the artwork at **one-tenth scale at 300 ppi**, which is the same number of pixels and a document your machine can handle.

Vector artwork has no resolution at all and needs none of this. Only placed photographs do.

Legibility, and the rule with its caveats

The commonly quoted rule is **25 mm of capital height for every 3 m of viewing distance** — roughly one inch per ten feet. It comes from readability research and it is a floor for *legibility*, not a target for *impact*.

The caveats matter:

- It assumes good contrast, a clear typeface and a stationary reader.
- A **moving** reader needs considerably more. Traffic signage standards use far larger letters for exactly this reason.
- A **condensed** face at the same cap height is harder to read.
- **Reversed** type on a dark ground needs a slightly heavier weight at distance.

So: 25 mm per 3 m as a minimum, doubled for anything read from a moving vehicle, and checked by printing a sample letter and walking away from it. That last test takes five minutes and settles every argument.

How big a capital has to be

Twenty-five millimetres of cap height for every three metres, which is a floor for legibility rather than a target for impact. It assumes good contrast, a clear face and a reader standing still; double it for anything read from a moving vehicle.

Panels and seams

A hoarding is almost never one piece. Flex or vinyl comes on rolls of a limited width — commonly 1.6 m, 2.2 m or 3.2 m — so a 6 m banner is printed in panels and welded or overlapped.

Find out where the seams fall before you design. Then keep off them:

- Faces, especially eyes and mouths.
- The logo.
- Any text.
- A precise geometric alignment that will visibly step at the join.

Two or three questions to the fabricator — roll width, seam position, overlap or weld — and you can place your elements in the panels rather than across them.

Substrates and finishing

- **Flex / PVC banner** — the standard hoarding material. Needs **eyelets** every 500 mm or so and a **hem** (a folded, welded edge) for strength. Add **100 mm of extra material** all round for the hem, and keep artwork out of it.

- **Mesh** — perforated, for windy sites and building wraps. Passes wind, so the image is seen through holes and loses about 30% of its density; design for it.
- **Self-adhesive vinyl on ACP** — shop boards.
- **Foam board / Sunboard** — interior only, light, cheap.
- **Backlit flex** — translucent, for lightboxes. Colours behave differently with light passing through them; ask for a sample lit from behind.

The things that are not your job but are your problem

- **Wind loading** on a large sign is an engineering question and a legal one in many places. Not your decision, and worth asking about, because a sign that comes down is everybody's problem.
- **Planning permission and local rules** on sign size, illumination and position exist in most jurisdictions and vary enormously. Ask the client whether they have it; do not advise on it.
- **Installation access** — a sign 8 m up needs a ladder, a scaffold or a cherry picker, and the cost of that can exceed the sign.

What to send

1:1 if the RIP takes it, otherwise 1:10 with the scale in the filename. Vector for everything except photographs. CMYK, with a profile if the shop names one. Bleed of **25–50 mm** rather than 3 mm, because large-format trimming is coarser. Hem and eyelet allowance marked on a separate layer. And a low-resolution reference JPEG so the operator can see what it should look like.

BEFORE YOU MOVE ON

A 6 × 3 m hoarding is to be printed from a photograph that is 7000 × 3500 pixels. The designer refuses the image as too low resolution and asks the client for a larger file. Is this correct?

A. No — 7000 pixels across 6 m is about 30 ppi, which is normal for a hoarding read from 30 m.

B. Yes — any printed image below 300 ppi at final size will show visible pixellation regardless of distance.

C. Yes, but only because large-format RIPs upsample to 150 ppi internally and the source must exceed that figure.

D. No — the file should instead be resampled up to 150 ppi at full size before printing, which is what the RIP expects.

CASE STUDY · MODULE 8

Eighteen hundred shirts and a sew-out nobody wanted to pay for

An embroidery unit in Ludhiana, a hotel group's new uniform, and three days of float in the schedule.

A hotel group opening a property in Amritsar ordered uniforms: eighteen hundred garments, split between woven cotton shirts for the front office and knit polos for housekeeping and the restaurant. Both carry the same chest logo.

The mark is a stylised lotus, drawn well, with a tagline beneath it. The group's marketing manager approved it on a screen and on a printed presentation, where it is 90 mm wide and looks excellent.

At the chest size the supplier works to, 55 mm, the lotus stem measures 0.9 mm and the tagline sits at 4 mm cap height. The mark carries five colours, one of which is a pale gold.

The digitiser quoted 11,400 stitches and five colour stops. Embroidery is priced per thousand stitches, and every colour change is a machine stop on every single garment, so both numbers are in the price on all eighteen hundred pieces.

The unit's manager asked for a sew-out before the run: one stitched sample on the actual fabric, seven hundred rupees, and a day and a half, because his machines were booked and he would have to slot it between orders.

The hotel's project manager said no. The artwork had been approved. The property handover date was fixed. There were three days of float in the schedule and he did not want to spend half of them on a sample of something already agreed.

The disagreement is a real one and both men have a reasonable case.

Against the sew-out: a day and a half out of three days of float, a small cost, and a conversation with a client who believes the decision was already made in a meeting they attended.

For the sew-out: satin stitch works down to about 1.2 mm and the stem is 0.9. Embroidered text below about 5 mm cap height smudges and the tagline is at 4. The knit polo will pull far more than the woven shirt, so the same digitised file will come out narrower on one than the other. And the failure mode is not a slightly disappointing garment: it is eighteen hundred garments, the thread, the machine time, and a property opening with no uniforms.

The digitiser can compensate for some of this. Pull compensation means widening shapes by two or three tenths of a millimetre on each side so the finished stitch lands where the drawing says. But compensation is per fabric, and nobody in the room knew how much this knit pulled, because nobody had stitched on it.

The split order made that worse rather than better. Nine hundred woven shirts and nine hundred knit polos is not one job with two fabrics; it is two digitising jobs, because the file that comes out right on poplin comes out narrow on a knit. Somebody was going to have to decide that, and the only way to decide it with a number rather than a guess was to stitch one of each.

The economics are worth stating plainly, because they are what made the argument possible at all. Seven hundred rupees is roughly the cost of four garments. Eleven thousand four hundred stitches at five colour stops, on eighteen hundred pieces, is the largest single line in the embroidery quotation, and every one of those numbers was set by decisions in the artwork rather than by anything the unit could control. A conversation about the artwork was therefore also a conversation about the price, and neither man had framed it that way.

So the decision at that table was not really about seven hundred rupees. It was about whether the unit would run eighteen hundred garments on an assumption nobody had tested, on fabric nobody had stitched, to hold three days of float.

STOP HERE

Before turning the page: what would you have done, and what would it have cost if you were wrong? Write it down. The point of the next section is to compare.

WHAT HAPPENED

The unit's manager refused to run without a sew-out. It is a standing rule he adopted after a 900-piece reject in 2021, and he would rather lose an order than repeat it.

The sew-out came back in two days and showed three things nobody had predicted exactly.

The stem closed into a dark line on the knit polo and held, just, on the woven shirt, which confirmed that one digitised file could not serve both. The tagline at 4 mm was not merely tight; it was unreadable at arm's length. And the pale gold disappeared entirely against the fabric colour, so one of the five colour stops was buying nothing at all.

The designer made three changes in an afternoon. The stem was widened by 0.4 mm on each side for the knit version only, which is ordinary pull compensation and not a change to the mark. The tagline came out of the embroidered version altogether and moved to a woven neck label, where small text belongs. Two colours were merged and the gold was dropped, taking the mark from five thread colours to three.

Stitch count fell from 11,400 to 9,200 and the colour stops from five to three, which reduced the per-garment price enough to pay for the sew-out roughly two hundred times over.

The second sew-out was approved from a photograph on the same day it came off the machine. The shirts shipped four days later, inside the original schedule.

Why did one digitised file behave differently on the polo and on the shirt?

Pull. As the needle draws thread through the cloth, the fabric is pulled inward across the stitch direction, so a shape comes out narrower than drawn and slightly longer along the stitch direction. Knitted fabric pulls considerably more than woven, so the compensation a digitiser builds in has to be set per fabric. The same file on two fabrics is two different finished widths, which is why a sew-out on the actual cloth is the only honest test.

The tagline moved to a neck label instead of being enlarged. Why is that usually the better answer?

Embroidered text has a hard floor of about 5 mm cap height, and lowercase wants 6. Enlarging the tagline to clear it would force the whole lockup wider than the chest position allows, so the mark would shrink relative to it or the lockup would stop fitting the pocket. Moving small text to a surface that can carry small text keeps both the mark and the words legible, and it is also cheaper, because it takes stitches out of every garment.

Going from five thread colours to three changed the price. What exactly did it buy, and what did it cost?

Each colour is a machine stop and a thread change on every single garment, so two fewer colours removed handling time from eighteen hundred pieces, and the lower stitch count reduced the per-thousand charge as well. What it cost was two of the brand's colours in this application, which is a real loss and has to be a decision rather than an accident. It belongs on the usage sheet: the embroidered version uses three threads, named, and that is the approved version for garments.

MODULE 8 · TEST

Check what stuck

6 questions, one right answer each. This one is for you, not for a record: answers and the reason behind each are at the back. If two questions from the same module go wrong, re-read that module before the exam rather than after.

1 Why does a stroked letter come out cut as a single line down its middle on a plotter?

A. The plotter cuts the centreline, because a stroke is not geometry
B. The plotter ignores any stroke width above its blade diameter
C. The export converted the stroke into a hairline on the way out
D. The blade offset was set too small for the blade and material loaded

2 What sets the minimum feature size in cut vinyl, at around 2 mm?

A. The blade's own diameter, which cannot turn inside a radius of 2 mm
B. The plotter's step resolution, which is about 2 mm per pulse
C. The application tape, which will not hold anything smaller
D. Weeding: a person pulls the waste out by hand and thin parts tear

3 How does the cut line travel in a print-and-cut file?

A. As a filled shape on a layer called Cut, printed in solid black
B. As an outline on a named spot swatch, set to overprint
C. As a separate DXF file sent alongside the printed PDF
D. As four registration marks that you draw at the corners

4 A circle digitised for embroidery without compensation comes out as an oval. Why?

A. The hoop stretches the fabric, which then relaxes afterwards
B. The fabric is pulled in across the stitch direction and pushed along it
C. Satin stitch cannot follow a curve without being segmented first
D. The machine rounds every coordinate to the nearest tenth of a millimetre

5 Two laser-cut parts drawn to slot together exactly come out loose. By how much, and why?

A. By one kerf, because the beam only removes material from the hole
B. By half a kerf, because the beam is centred on the drawn path
C. By two kerfs, because the hole grows and the part shrinks
D. By nothing; looseness comes from the sheet shifting in the bed

6 Why must a laser never be used on sign vinyl or any other chlorine-bearing plastic?

A. The chlorine in it reflects the beam back into the lens assembly
B. It releases hydrogen chloride, corroding the machine and harming people
C. It melts instead of cutting, so the cut edge fuses itself back together
D. It burns at a very uneven rate, which makes the kerf impossible to predict

EXAMINATION

Vectors, Logos and Things That Get Printed: final examination

This paper covers the whole course: what is actually inside a vector file, drawing and correcting curves, type and its licences, colour that survives a press, designing and delivering a mark, building a document that survives a late change, the handover to a printer, and preparing files for machines that cut, stitch and burn. Twenty-five questions are drawn for you from a larger pool, so no two attempts are the same paper. The pass mark is 70 per cent. There is no timer: read each question as slowly as you like. If you do not pass, you may retake after thirty minutes and the questions will be drawn fresh. Passing opens the next course in the track, and a teacher can open it for you regardless of this result.

On the website a sitting is 25 questions drawn from this pool and the pass mark is 70%. There is no time limit, there and here. Passing opens the next course in the track.

1 `1. The thing that is actually in the file`

An SVG path reads M 20,80 C 20,20 80,20 80,80 Z. What are the first four numbers after the C?

A. The positions of the two handles, before the end point is given
B. A control point and the chord length from the previous anchor
C. Two points on the curve that it is guaranteed to pass through
D. A centre point and a radius, which the renderer turns into an arc

2 `1. The thing that is actually in the file`

You open a 1.9 MB file somebody has sent you as a vector logo and find the string base64 inside it. What have you got?

A. A font embedded in the file, which is normal in an SVG
B. An editor's metadata, which svgo or scour will strip out for you
C. A raster image embedded inside a file that is called a vector
D. A compressed set of path data that any editor can expand again

3 `1. The thing that is actually in the file`

A client asks you for a 300 dpi version of their logo. What is the useful reply?

A. Export the SVG at 300 dpi from the document properties dialog
B. Ask what the finished size is, because a vector file has no dpi
C. Send a 300 dpi PNG at 1000 pixels, which covers most uses
D. Explain that 300 dpi applies only to CMYK files, not to RGB ones

4 `1. The thing that is actually in the file`

A mark drawn at 200 mm with a 2 mm outline is scaled to a 15 mm badge and the outline looks like a blob. What happened?

A. The miter limit bevelled the corners once the shape got smaller than the limit
B. The stroke alignment changed from centre to outside during scaling
C. Dot gain thickened the line, which is normal at that size in print
D. The stroke was set not to scale with the object, so it stayed 2 mm

5 `1. The thing that is actually in the file`

A line specified as 10 mm long, with a 4 mm round cap at each end, measures 14 mm on the artwork. Why?

A. The measurement includes the bounding box, not the geometry
B. The stroke width was added to the length by the export
C. Round caps extend half the stroke width past each endpoint
D. The path was drawn 14 mm and the dimension recorded wrongly

6 `1. The thing that is actually in the file`

Why is a black rectangle drawn over a name in a PDF not redaction?

A. It is a drawing on top, and the text underneath is still extractable
B. It flattens into a raster region, which some viewers then render as grey
C. It only works if the PDF is exported as PDF/X-1a with no layers
D. It hides the name on screen but prints the text on a press

7 1. The thing that is actually in the file

Why can a multi-colour auto-trace produce a doubled edge on a cutting plotter even though it looks correct on screen?

A. The plotter cannot read fill colours, so it cuts every path it finds
B. The tracer writes each colour as a separate file that the plotter merges
C. Each region is stroked as well as filled, so the blade follows both
D. The output is a stack of opaque shapes that overlap by a pixel or two

8 2. Drawing curves people will pay for

A beginner drags the pen tool towards the place they want the curve to bend, and gets a loop or an S instead of an arc. Why?

A. The pen tool measures the drag from the previous anchor, not from this one
B. The handle sets the direction of travel as the curve leaves the anchor
C. Dragging towards the bend creates a cusp node rather than a smooth one
D. The first handle is mirrored, so dragging inwards reverses both of them

9 2. Drawing curves people will pay for

Why do experienced people build marks from primitives and booleans rather than drawing the outline with the pen?

A. A circle from the ellipse tool is exact; one drawn by hand is four near misses
B. Booleans are non-destructive, so the whole construction stays editable afterwards
C. The pen tool cannot close a path accurately without snapping enabled
D. Primitives use fewer bytes, so the file opens faster on a shop machine

10 2. Drawing curves people will pay for

After a boolean, the result has taken a fill you did not choose. What is the habit that deals with it?

A. Convert the result to a compound path, which resets the appearance
B. Undo, set both of the inputs to the target colour, and repeat the operation
C. Set the colour after the boolean, because the inherited rule differs
D. Always select the object you want the colour of last before welding

11 2. Drawing curves people will pay for

Why must thin features be checked after an inward offset, such as an underbase choke or a laser kerf compensation?

A. Inward offsets are calculated at a lower precision than outward ones
B. A feature narrower than twice the offset vanishes or self-intersects
C. The offset is applied per subpath, so compound paths lose their holes
D. An inward offset rounds every corner, which closes narrow gaps

12 2. Drawing curves people will pay for

Twelve circles are selected and scaled to 50 per cent. What is the difference between doing that and using Transform Each?

A. Transform Each applies the scale about the page origin instead
B. A plain scale converts the selection into a group, which is harder to undo
C. Transform Each keeps the strokes at their original weight
D. A plain scale shrinks the pattern; Transform Each shrinks only the dots

13 2. Drawing curves people will pay for

Why should pixel grid alignment be switched off for print artwork?

A. It disables snapping to guides, which print layouts depend on
B. It rounds stroke weights to whole pixels, so the hairlines disappear entirely
C. It snaps edges to a 96-per-inch grid, adding up to 0.13 mm of error
D. It converts millimetres to points at 72 rather than 96 to the inch

14 `2. Drawing curves people will pay for`

When rebuilding a logo from a photograph, why is locking the dimmed template layer described as not optional?

A. An unlocked template gets nudged, and you then draw to a moved reference
B. Locking forces the application to redraw it at full resolution
C. An unlocked template exports along with the artwork and shows up on the proof
D. Locking is what allows the opacity to be set below 100 per cent

15 `3. Type, which is also vector`

You have the same typeface as both a TTF and an OTF, and you are going to edit the wordmark by hand. Which do you take, and why?

A. Either, because outlining converts both into the same cubic representation
B. The TTF, because its hinting survives the conversion to outlines
C. The TTF, because quadratic curves give you more control points to move
D. The OTF, because its cubic outlines come across as the designer drew them

16 `3. Type, which is also vector`

A pdffonts listing shows a font named ABCDEF+Mukta-Regular with yes under emb. What does the prefix tell you?

A. Six variants of the family were merged into one embedded font
B. The embedded copy is a subset containing only the glyphs used
C. The font was instanced from a variable file at those axis values
D. The font was substituted, and ABCDEF identifies the replacement

17 3. Type, which is also vector

A client sends you an OTF with the brief and asks you to use it. What do you ask, in one email?

A. Whether they hold a desktop licence, and whether it covers you as supplier
B. Whether the font is OFL, because only OFL fonts may be used commercially
C. Whether they can send the web version as well, in case it is needed later on
D. Whether the foundry permits outlining, since that is how it will be output

18 3. Type, which is also vector

A paragraph justified at a 45-character measure has developed vertical channels of white running down it. What are they, and what fixes them?

A. A ladder, and the fix is to limit consecutive hyphens to two
B. Optical margin drift, and the fix is to hang the punctuation outside
C. Rivers, and the fix is a longer measure or setting it ragged right
D. Widows, and the fix is the Keep Options in the paragraph style

19 3. Type, which is also vector

A price column will not line up however you set the tabs. What is wrong?

A. The cells are centred rather than right-aligned in the table style
B. The figures are proportional, so a 1 is narrower than an 8
C. The tab is a left tab and needs to be a decimal tab instead
D. The table has no repeating header row, so the columns drift

20 3. Type, which is also vector

A price is set in your display face and one character, the rupee sign, is visibly in a different typeface. What has happened?

A. The face carries the rupee only in its tabular figure set
B. The PDF subset dropped it, so the viewer is showing you a fallback glyph
C. The character was pasted from another document with its own style
D. The glyph is missing and the application silently substituted a font

21 3. Type, which is also vector

Why is a variable font's width axis an acceptable way to condense a headline when horizontal scaling is not?

A. A width axis changes the advance widths without changing the outlines
B. A width axis is applied by the RIP itself, so no distortion ever reaches the plate
C. A width axis narrows the letter as designed; scaling thins the verticals only
D. Scaling is forbidden by PDF/X, while an axis value is permitted

22 4. Colour that survives the press

A logo has a white element in the middle of a four-colour process job on white stock. How is that white produced?

A. By leaving the paper unprinted; there is no white ink
B. By knocking out all four plates and overprinting a spot white
C. As a fifth station carrying opaque white ink
D. As 0/0/0/0, which the RIP renders as a very light tint

23 4. Colour that survives the press

A supplier quotes a colour match of Delta E 1.8. What must you ask before that number means anything?

A. Whether the figure applies to the solid or to the fifty per cent tint
B. Whether the press was running coated or uncoated stock that day
C. How many sheets were measured across the length of the run
D. Which formula was used, and under what measurement condition

24 4. Colour that survives the press

Which destination CMYK profile should a job be converted to?

A. Whichever your application offers as the working CMYK space
B. Whichever the printer's RIP is set to, which is a question for them
C. Always ISO Coated v2, which is the international default for all work
D. The profile embedded in the client's supplied photographs

25 4. Colour that survives the press

Pure RGB blue comes back from a CMYK conversion as a flat purple navy, while a cyan-leaning blue survives almost untouched. What does that tell you about choosing a brand colour?

A. That colours near a process primary reproduce and corner colours do not
B. That the monitor was unprofiled, so the original was never that blue
C. That blues should always be specified as a spot ink from the start
D. That the conversion used the wrong rendering intent for this kind of artwork

26 4. Colour that survives the press

Why is each process ink screened at a different angle?

A. To compensate for the direction the paper stretches through the press
B. To spread the ink load evenly across the four printing units
C. So the four dot patterns form a rosette rather than interfering
D. So the RIP can identify which plate a dot belongs to

27 4. Colour that survives the press

How many usable values can a single plate give you in a one-colour job?

A. Three: the solid, a single mid tint, and the paper showing through it
B. About five: the solid, three usable tints, and the reversed paper
C. One: a plate carries a single ink at full strength
D. Two: the solid and the paper showing through a reversed area

28 4. Colour that survives the press

A client photographs a printed sheet on their phone and says the colour is wrong. What is the useful reply?

A. Send them a screenshot of the file with the correct CMYK values marked
B. Ask the printer to remeasure the sheet and send the Delta E figure
C. Ask them to resend the photograph taken in daylight with the flash turned off
D. Ask them to hold the sheet beside the agreed printed reference and compare

29 5. Designing a mark

Which single question in a logo brief decides most of the design?

A. What three adjectives describe how the organisation wants to feel
B. Which competitors' marks the client admires and which they dislike
C. Where will this appear, specifically, listing every place you can think of
D. What the budget is, since that sets how many rounds of revision are available

30 5. Designing a mark

Why present exactly three concepts, all of which you would be happy to see used?

A. Because a client may choose the weak one, and then you have to make it
B. Because three fits on one page at a size where each can be judged
C. Because showing more than three invites a request to combine two of them
D. Three is what clients expect, so fewer looks like a lack of effort

31 5. Designing a mark

What is the honest status of the golden-ratio circle diagrams laid over famous logos?

A. A rough guide, accurate to within a percentage point on most marks
B. A construction method used by most large agencies since the 1970s
C. A proportional system that only works for marks built on a square grid module
D. Almost always retrofitted decoration, placed afterwards to fit the mark

32 5. Designing a mark

A triangle centred in a circle by its bounding box looks off to the left. What do you do, and why?

A. Recentre it on the triangle's centroid, which is the true centre of its area
B. Nudge it right, because its visual mass sits left of its geometric centre
C. Scale it down by one or two per cent so the imbalance is less visible
D. Leave it, because bounding-box centring is geometrically correct

33 5. Designing a mark

Why deliver a separate stacked lockup rather than letting somebody scale the horizontal one?

A. A narrow space needs a different arrangement, drawn and spaced for it
B. Most applications cannot scale a grouped lockup without distortion
C. A scaled lockup loses its clear space, which is defined in millimetres
D. Scaling changes the stroke weights unless stroke scaling is enabled

34 5. Designing a mark

Why is the single-thread embroidery version usually a separate drawing rather than a recolouring?

A. Thread colours must be picked from one brand's chart, which changes the hues
B. Embroidery software cannot import a file with more than one colour in it
C. Fine detail below about 1.5 mm disappears, so it must be redrawn thicker
D. The machine mirrors the design, so the artwork has to be drawn reversed

35 5. Designing a mark

A client asks whether they will own the logo. What is the accurate shape of the answer?

A. Copyright always vests in the client, since they commissioned the work
B. Copyright in the drawing and a trademark for the sign are different things
C. Registering the trademark automatically assigns the copyright as well
D. Ownership is decided by whoever paid the invoice, in every jurisdiction there is

36 6. Documents that survive a late change

Which combination should be avoided outright when choosing a layout tool?

A. CorelDRAW for an eight-page brochure with a contents list
B. Scribus for a press job, because its colour management is limited
C. LaTeX for a thesis, because visual control of placement is indirect
D. A word processor producing a press PDF for a job with bleed

37 6. Documents that survive a late change

Page numbers on four pages of an eighty-page document sit 3 mm out of place. What is the likeliest cause?

A. Those four pages use a section with a different numbering style
B. The folio is a typed number rather than a page number marker
C. The master item on those pages was overridden and is now local
D. The master page was edited after those pages were created

38 6. Documents that survive a late change

Copy overruns its frame by three lines on a designed spread. Which fix is never acceptable?

A. Scaling the type horizontally to 96 per cent to gain the space
B. Changing the body style globally from 10 on 14 to 9.5 on 13
C. Tracking the offending paragraph by minus five thousandths of an em
D. Cutting forty words from the copy, which changes the author's text

39 6. Documents that survive a late change

What does a real table object give you that tabbed text cannot?

A. Tabular figures, which are a property of the table rather than the font
B. Automatic sorting of rows, which keeps the data current
C. Right-aligned numbers, which tabs cannot produce in any application
D. Repeating header rows, cell styles and central restyling

40 6. Documents that survive a late change

Why does resampling a 150 ppi image up to 300 ppi not help?

A. Most RIPs will downsample it again to suit the line screen they are running
B. The detail is not there to recover, so you get a bigger, softer file
C. The stored dpi tag stays at 150, so the RIP ignores the new pixels
D. The interpolation runs in RGB, so it clips on conversion to CMYK

41 `6. Documents that survive a late change`

On a 64-page saddle-stitched booklet, the folios on the centre pages sit closer to the trimmed fore-edge than on the outer pages. What is this?

A. Creep: inner sheets project further and are trimmed more at the fore-edge
B. Gutter loss, which is the reason the inner margin is set larger than the outer
C. A master page override applied to the centre spread during layout
D. Imposition error, which the press corrects by reprinting the section

42 `6. Documents that survive a late change`

Why should a reference to a figure number and its page be inserted as a cross-reference rather than typed?

A. A cross-reference forces the figure caption onto the same page as its text body
B. Typed references are not picked up by the generated contents list
C. A cross-reference updates when the figure moves, and flags a deleted target
D. Typed references cannot be made into PDF hyperlinks on export

43 `7. The handover`

Why is a decorative border sitting exactly on the trim line described as the worst decision in print?

A. The RIP cannot resolve a line that lies exactly on the boundary of the TrimBox
B. Guillotine drift of a millimetre or two makes it visibly uneven on one side
C. A border on the trim cannot be given bleed, so it will show white
D. Crop marks are placed on the trim, so they will print over the border

44 7. The handover

Why must small black text never be set in rich black?

A. Rich black is defined per press, so the build changes between suppliers
B. The RIP converts small text to 100 per cent K anyway, so it is wasted
C. Rich black exceeds the total ink limit at any size of type
D. Four plates never register perfectly, so every letter gets a fringe

45 7. The handover

What two things does a normal view of a PDF in Acrobat hide, unless you ask for them?

A. Transparency flattening, and the output intent named in the file
B. The bleed and trim boxes, and the total ink coverage per page
C. Overprint, and the separations that show what is on each plate
D. Embedded fonts, and the effective resolution of placed images

46 7. The handover

Separations show a fifth plate on a job you believed was four-colour. Which of these is a classic cause?

A. A spot swatch defined and used once on a hidden or off-page object
B. Crop marks, which are always written to a fifth separation
C. An image carrying an embedded profile that the export did not convert
D. A photograph placed as CMYK TIFF rather than as a PSD

47 7. The handover

What can a preflight run never tell you?

A. Whether the total ink coverage exceeds the press's stated limit
B. Whether the TrimBox matches the size you agreed with the client
C. Whether every font in the file has actually been embedded
D. Whether the client's name is spelled correctly on the page

48 7. The handover

A carton needs a spot varnish over the logo. How is that supplied?

A. As a separate PDF showing the varnish areas in position
B. As solid shapes on their own layer, in a named spot, set to overprint
C. As a transparency group above the artwork with a blend mode applied
D. As a 20 per cent tint of black on the artwork layer, marked in the note

49 7. The handover

Why does the printer need to know where the folds are before choosing paper?

A. Folding across the grain cracks the fold on every copy
B. The fold position determines the total ink limit for the sheet
C. Grain direction decides the sheet size the press can run
D. A fold across the grain increases dot gain along the crease

50 8. Machines that cut, stitch and burn

What single property must a file have before it goes to a plotter, a laser or an embroidery machine?

A. It must contain no colour information, which machines cannot interpret
B. It must be supplied as a DXF, which is the only universal machine format
C. It must describe a physical action: closed paths and outlined strokes
D. It must be built at one tenth scale, so the machine's software can size it

51 8. Machines that cut, stitch and burn

Why does a letter carrying four thousand nodes take minutes to cut and come out rougher?

A. The blade rotates to a new angle at every node, which dulls the edge
B. The plotter decelerates at every node, so the path is cut in tiny steps
C. The software converts each node to a separate cut, then reassembles them
D. The controller's buffer overflows and the machine restarts the path

52 8. Machines that cut, stitch and burn

Corners on a print-and-cut job come out slightly rounded. Whose problem is it, and what is the cause?

A. Yours: the cut path was drawn with rounded corners in the artwork
B. The RIP's: the registration marks were placed too close to the artwork edge
C. Yours: the cut path was offset inwards, which rounds every corner
D. The operator's: the blade offset is set too small for the blade fitted

53 8. Machines that cut, stitch and burn

Satin stitch is described as working between about 1.2 and 12 mm. What happens outside that range?

A. Below it the stitch becomes a run; above it it becomes a fill automatically
B. Below it the thread breaks; above it the machine refuses the file
C. Below it the needle perforates a line; above it the threads snag and sag
D. Below it the colour reads wrongly; above it the fabric puckers

54 8. Machines that cut, stitch and burn

A shop board's letters have square internal corners in the artwork and rounded ones after routing. What is the fix?

A. Draw the corner radius in, or add dog-bone fillets where square is needed
B. Route the letters face down so the rounding falls on the hidden side
C. Increase the feed rate, which reduces the radius the bit leaves
D. Ask for a second finishing pass with a smaller bit to square the corners up

55 8. Machines that cut, stitch and burn

Why is an underbase on a dark garment choked inwards by 0.3 to 0.5 mm?

A. To keep the halftone dots from clogging the mesh during the run
B. To let the underbase dry fully before it reaches the flash-cure stage itself
C. To reduce the amount of white ink used across a long run
D. To stop the white showing around the edges when the colours land off

56 8. Machines that cut, stitch and burn

A client wants a full-colour design on dark polyester sports shirts, twenty pieces. Which process is ruled out immediately, and why?

A. Direct to film, because it does not bond to polyester fibres
B. Sublimation, because there is no white ink and the shirt is dark
C. Direct to garment, because the run is too short to pre-treat
D. Screen printing, because twenty pieces will not justify the screens

Answers

Each entry gives the correct option, then what each wrong one reveals about how somebody was thinking. The second part is worth more than the first: a wrong answer here is a named misreading, not a mistake.

Lesson checks

1. Why your Photoshop logo cannot go on a hoarding — C

A — Treats a routine colour conversion as the obstacle instead of the missing geometry.

B — Reads dpi as a quality dial that must rise with size, when the requirement actually falls as viewing distance grows.

D — Blames lossy compression for a resolution limit, when PNG is lossless and the pixel count is identical either way.

2. What the `d` attribute is saying — C

A — Assumes a vector file has a resolution. Nothing in an SVG stores one; the size difference is entirely node count and number precision.

B — Confuses transport size with geometry. Compression shrinks the bytes but leaves 412 paths for the renderer and the cutter to process.

D — SVG has no CMYK. Colour notation is a handful of bytes either way and never accounts for a megabyte.

3. Points, millimetres, and two different inches — B

A — Mixes up the two conversions. Points and millimetres would give a far larger discrepancy than 6.25%, and the file's units were never points.

C — Invents a default. Document scale is 1 unless someone changes it; the discrepancy comes from the units-per-inch constant, not a scale factor.

D — SVG stores real unit suffixes perfectly well. The problem is files that omit them, leaving the renderer to apply its own constant.

4. A path has no thickness — B

A — Registration drift moves artwork by tenths of a millimetre; it does not remove a line from the whole run, and it would not explain the proof looking right.

C — Treats hairline as a fixed measurement. It is device-relative, which is exactly why the same file behaved differently on two devices.

D — Flattening affects transparency, not strokes. A stroke survives to the RIP perfectly well — the question is what width it asks for.

5. Why the hole will not appear — C

A — A group would show two independent objects with their own fills; the hole appears with no extra object present, because it is a fill-rule decision on one path.

B — Open paths are filled as if closed by a straight line back to the start; openness changes the stroke, not the winding calculation.

D — There is only one subpath here, so there are no opposing directions to detect. Direction matters for non-zero, but the difference between the two renderings is the rule itself.

6. Clipping hides; it does not remove — B

A — Compression never reconstructs removed data. The pixels were recoverable because they were never removed.

C — Invents a mechanism. Some PDFs carry page thumbnails, but they are low-resolution renderings of the composed page, not the uncropped source.

D — Both routes are still clipping. No clipping operation in a layout application deletes source pixels; only editing the image file or cropping on export does.

7. Where vector quietly becomes pixels — C

A — A low raster-effects resolution produces a visibly stepped, chunky shadow. It does not create a rectangular boundary or a seam, which are geometry, not sampling.

B — Colour conversion changes hue and density uniformly; it cannot produce an edge at a rectangle boundary that has nothing to do with the artwork's shape.

D — Coarser screening shows as dot pattern across the whole tonal area, not as a rectangular boundary and a single straight seam.

8. What auto-trace is doing, and when to redraw instead — A

B — Speckles appear as separate small shapes, not as a smooth wave along an otherwise continuous edge.

C — Blade offset shows as rounded or overshooting corners, not as waviness along a straight run — and the wobble is present in the file before it reaches the machine.

D — Tracers convert to greyscale internally before thresholding; the colour mode of the source does not introduce a spatial variation along an edge.

9. Your curves are lumpy because you used too many points — B

A — Treats node type as the cause, when the problem is the sheer number of tangent directions being declared.

C — Mistakes a geometry problem for a rendering one, which resolution independence already rules out.

D — Expects an averaging algorithm to recover intent, when it removes points blindly and rounds off deliberate corners.

10. Most good logos are circles and rectangles cut into each other — A

B — Pictures the hole as a white object stacked on top rather than as a region excluded by the fill rule.

C — Applies a whole-object property to a subpath that no longer has a fill of its own once combined.

D — Assumes the wrong operation was chosen, missing that a correctly combined compound path behaves exactly this way when the winding is wrong.

11. The pen tool is four gestures — B

A — Cusp nodes allow an inflection; they do not cause one. The node type here was smooth, and the shape came from where the handle pointed.

C — An over-long handle produces a bulge or a loop at one end, not a symmetrical S; the direction was wrong before the length mattered.

D — Click-and-drag sets the handles of the anchor being placed. The previous anchor's outgoing handle was already fixed by its own gesture.

12. The kink you can see but cannot find — C

A — If the handles are collinear the join is G1 regardless of the label; the node type is not what is producing the artefact.

B — Scaling multiplies all coordinates uniformly, so a smooth curve stays smooth; coordinate precision at this size is far below a visible flat.

D — Stroke alignment is a property of the whole path, not of individual segments, and it would show as a step in width rather than a flat in the curve.

13. Booleans, and the mess they leave behind — D

A — Excess pressure cuts the backing along the whole path, not as isolated nicks at a few points, and it would not correlate with the boolean history.

B — SVG precision loss moves points by fractions of a micron at normal export settings; it does not create new geometry for the blade to follow.

C — Uncut strokes produce a full extra outline through the middle of every shape, which would be obvious, not a few small nicks.

14. Offsets, and the spike that appears at a sharp corner — C

A — Offset is always perpendicular to the path. Nothing accumulates along the direction of travel, and the effect would be uniform rather than concentrated at points.

B — Reverses cause and effect: the limit is a cap that switches a long mitre to a bevel. Raising it permits the spike, but the length comes from the corner angle.

D — A self-intersecting star changes which regions are filled, not how far a mitre extends; the same spike appears on a cleanly drawn star.

15. Transforms are arithmetic, so type the numbers — B

A — Tighter or looser snapping cannot fix a chain of transforms each applied to the output of the previous one; the error is cumulative, not per-step tolerance.

C — A 360° rotation returns every object to where it already was; it cannot redistribute unevenly placed copies.

D — Compound paths store coordinates the same way as any other path. Precision is not the issue; repeated dragging by eye about a shifting pivot is.

16. Grids and snapping that help instead of fighting — A

B — Alignment would change the gap by one stroke width at most and would show on screen as unequal edge positions, not as a flush-looking join.

C — Outlining a centred stroke expands by half the width on each side of the path and does not move the objects relative to one another.

D — Grid snapping would leave a gap equal to a grid interval, not one exactly matching the combined stroke widths, and both edges would sit on grid lines.

17. Rebuilding a mark that only exists as a photograph — B

A — Tracers run happily at any resolution. The problem is not a size limit in the algorithm but what the algorithm is inferring from a blurred, compressed source.

C — Colour mode is trivially changed and is unrelated to the geometry problem; a redrawn mark needs the same conversion.

D — A trace replaces the bitmap with paths; the source is discarded unless you ask to keep it. The inherited error is in the path shapes themselves.

18. Why the printer asked you to outline the fonts — D

A — Invents extra work for the RIP; hinting stops mattering once glyphs are plain geometry.

B — Confuses file size with processing load, and PDF compresses path data in any case.

C — Attaches an unrelated flattening step to a conversion that changes only how the marks are described.

19. What is actually inside a font file — B

A — Quadratic curves convert to cubic exactly — every quadratic has an exact cubic equivalent. The node count rises but the shape does not move.

C — Losing the character map would produce missing or wrong glyphs, not subtly different rendering of the correct ones.

D — Outlining preserves the positions the text engine already computed; it does not apply any additional spacing.

20. The licence, which is the part that bites — D

A — Desktop licences generally do permit output, whether embedded in a PDF or as outlines. The issue is not output; it is handing over the software.

B — Packaging copies whole font files; subsetting happens on PDF export. Subsetting for output is routinely permitted in any case.

C — Agency does not transfer a software licence by default. Some licences have an explicit output-provider clause, which is exactly why you check rather than assume.

21. Kerning a wordmark by hand — B

A — Kern offsets are geometry and do not change sign when colours are swapped; the artwork is identical, only the tonal relationship differs.

C — Explains a press difference, not the on-screen difference — the looseness is visible in the file before anything is printed.

D — Nothing in the question changes the typeface or its optical size; the same outlines are being viewed with the tones reversed.

22. Setting a paragraph somebody can read — B

A — Tighter leading makes rivers harder to see, not easier; the channels come from horizontal word spaces aligning, which leading does not control.

C — Close, but incomplete: hyphenation helps, and the underlying cause is that 40 characters gives the composer too few spaces to distribute the slack across.

D — Justification overrides the font's word space entirely; the stretched value is computed per line, so the font's default is not what you are seeing.

23. Type on a curve, and why it splays — A

B — Baseline shift moves the whole line radially; it changes the effective radius slightly but does not alter that the glyph is rigid across its height.

C — A many-sided polygon is geometrically almost identical to the circle, so the arc-length difference between baseline and cap line is unchanged.

D — Skew keeps glyphs upright and shears them instead, which distorts the letterforms rather than fixing the spacing.

24. Devanagari, Arabic, and the scripts that break — B

A — Font tables do not behave differently when embedded, and the question states the font has the glyphs; reordering is applied by the engine, not stored per-render.

C — Outlining freezes whatever the engine produced. If the engine had shaped correctly, the outlines would be in the correct visual order.

D — PDF has carried Unicode text for decades and glyph positions are written explicitly, not reconstructed from character codes at display time.

25. Figures, dashes and the characters nobody checks — C

A — A right-aligned tab stop positions the following text at the stop regardless of the tab's own width; the space the tab occupies is not what varies here.

B — Justification stretches spaces between words within the measure; a right tab stop is an absolute position and is not affected.

D — Oldstyle figures vary in height, not in horizontal position, so they would look uneven vertically rather than misaligned across the column.

26. Variable fonts, and what a press does with one — C

A — Horizontal scaling is baked into the glyph positions and outlines on export; the press receives exactly what you saw. The objection is aesthetic and structural, not technical.

B — Both remain live text. Scaled type embeds perfectly well; nothing forces outlining in either case.

D — Both operate at the same unit precision. Rounding is far below anything visible; the difference is in stroke weight relationships, not accuracy.

27. Two different physical processes — B

A — Dot gain would shift the CMYK version under every light equally, not produce a difference that appears only when the illuminant changes.

C — Inks are not formulated to an illuminant. Viewing standards apply to how you look at the result, not to how the ink is mixed.

D — Rendering intent affects the conversion once, producing one fixed set of ink values; it cannot make the printed result respond differently to different lights.

28. Lab, Delta E, and what "close enough" means — C

A — Gamut is about which colours a device can produce. The problem here is that the two existing notations mean different things on different equipment, not that the colour is unreachable.

B — Precision is not the issue; both notations are exact numbers. They are exact instructions to unspecified devices.

D — RIPs work in device space and use profiles connecting through a reference space; specifying Lab does not remove a conversion step.

29. Profiles, and the dialog that decides everything — B

A — Converting from an assumed sRGB to Adobe RGB changes the numbers to preserve the flat appearance you are looking at, locking in the error.

C — Treats a misinterpretation as an aesthetic problem. Manual correction guesses at a transform that a single assignment performs exactly.

D — Applications read Adobe RGB perfectly well; the file simply failed to say what it was. Going back to raw is unnecessary work for a tagging problem.

30. What happens to a colour the press cannot print — A

B — A document conversion applies the same destination profile to everything; an embedded image profile governs its own source, not the artwork beside it.

C — Describes relative colorimetric. Perceptual compresses rather than clips, which is precisely why in-gamut colours also move.

D — Black point compensation affects the mapping of the darkest tones; it does not move a mid-tone saturated orange perceptibly.

31. Why your 20% tint printed at 30% — D

A — FM screening changes dot placement, not total coverage, and it is generally chosen to improve fidelity rather than to darken light tints.

B — Paper brightness contributes a little, but a 5% tint becoming clearly visible is an ink-coverage effect, not a substrate-brightness effect.

C — Screening does not inherently darken. A correctly profiled screened tint matches its continuous-tone equivalent.

32. A spot colour is a plate, and a plate is money — B

A — RIPs read separation names perfectly well. The quote reflects what is actually in the file, not a failure to read names.

C — PDF/X-4 preserves spot colours; it is about live transparency and colour management, not about flattening separations.

D — An RGB colour does get converted, but the underlying problem is that the tool never created a spot separation to begin with — a CMYK swatch in Inkscape would behave the same way.

33. Getting a whole design out of two inks — C

A — Overprint preview does honour tint values. The mismatch is about ink physics, not about the preview ignoring percentages.

B — Sequence does change the result, but the mismatch with a screen preview would exist in either order — the screen model is the thing that does not match.

D — Spot inks overprint readily; that is the basis of two-colour work. A knockout would have shown pure red, not a plum at all.

34. Trapping, and the white line around everything — C

A — Units are interchangeable and 0.12 mm is a normal trap value. The problem is the direction, not the size.

B — Spread and choke describe which colour moves; both are valid techniques. Naming them is not what went wrong.

D — A knocked-out trap object would leave the registration problem unsolved, but the question states registration is now forgiving, so overprint was applied.

35. Proofing when you cannot afford a proofing system — C

A — The photograph's pixel values have been through white balance, tone mapping and compression, so they carry no measurable relationship to the ink on the sheet.

B — Acts before establishing that anything is wrong, and changes the conversion without evidence that the conversion was the problem.

D — The display is irrelevant when comparing two physical objects, and no phone photograph of print is reliable however it is lit.

36. If it does not work in black at 16 pixels, it is not a logo — B

A — Treats a reproduction-process limit as a resolution problem.

C — Assumes halftones work everywhere, when a needle and a rubber die cannot render dots at all.

D — Solves a colour-matching problem when the obstacle is that these processes cannot render tonal transitions at all.

37. The questions to ask before you draw anything — A

B — Treats a constraint as a problem to remove. Single-colour flexo is a legitimate process and many premium brands use it deliberately.

C — Flexo plates are imaged from vector artwork through a RIP like any other process; supplying a raster would reduce quality for no reason.

D — The bag is the surface the customer holds. A constraint on the most-seen application is the governing constraint, not an exception to be averaged away.

38. Five kinds of mark, and what each one costs — B

A — Abstract marks are among the strongest registrable marks precisely because they are distinctive; representationalism is not a registration requirement.

C — Reverses the reality: a simple abstract form is usually the easiest of all to reproduce small and in one colour.

D — Admiring a mark does not make a new, different abstract form derivative; the objection is commercial, not one of infringement.

39. Thumbnails, and why you never show a concept you would not ship — C

A — A weak rationale makes selection arbitrary, but the structural problem is that a concept the designer would refuse to build was put in front of someone with the power to choose it.

B — Showing one looks like no exploration was done and removes the client's agency entirely; the failure here is what was shown, not how many.

D — Presenting a weak option badly is still presenting it. The remedy is not to show it in a less flattering way but not to show it.

40. Real construction, and the golden-ratio story — D

A — The question states the heights are mathematically identical, which is exactly the intended relationship; the scale of the grid is not what viewers are responding to.

B — Anti-aliasing affects both shapes and the effect exists at any size, including in print where there is no anti-aliasing at all.

C — Invents a rule. Any construction system can contain circles; the issue is that no construction system substitutes for optical correction.

41. Lockups, clear space and the smallest it may ever be — B

A — A percentage of the surrounding space makes the zone depend on the layout rather than on the mark, which changes the relationship every time.

C — Clear space applies wherever the mark appears; fabrication constraints are additional, not a replacement.

D — There is no such industry standard. Clear space is defined per mark by its designer, in relative units.

42. Reversed, on a photograph, and in one thread — C

A — A raster cannot be cut at all — the plotter would reject it. The rectangle is from the artwork's opaque background, not from its format.

B — Closed counters would show as filled holes inside the letterforms, not as a rectangle surrounding the whole mark.

D — A gloss mismatch would follow the mark's own outline, not appear as a rectangle, and would not depend on how the artwork was made.

43. Names, trademarks and who owns the file — C

A — Redrawing strengthens the authorship claim, but whether it is sufficient — and what "sufficient" means — is exactly the unsettled question, and it differs between countries.

B — Overstates the position. Positions vary by jurisdiction and turn on the degree of human authorship, not on whether a tool was touched.

D — Registration of a trademark does not create copyright; they are separate rights arising in different ways.

44. The folder you hand over — C

A — Supplies a raster when a press needs vector, and answers the literal wording rather than the actual need. A 1024 px PNG fails at any decent print size.

B — Hands over an editable master and any embedded fonts unnecessarily, and most printers neither want nor should modify the artwork.

D — Accepts a category error and extends it. Asking for a dpi on a logo produces a number that describes nothing about vector artwork.

45. Eighty pages, one font change, six hours of your life — A

B — Assumes divergence must come from a separate style, missing that a style can be overridden in place while still showing its own name.

C — Expects font substitution to happen silently and selectively, when it is flagged and would not follow paragraph boundaries.

D — Believes masters carry a separate style list, when a master supplies repeating items and draws on the same styles.

46. Choosing a layout tool, honestly — B

A — Correctly identifies Scribus's strength, but colour management is irrelevant to a text-only mono interior, and Scribus is heavier work for this than the alternatives.

C — A reasonable pipeline, but it adds a toolchain and a skill the author does not have for a benefit — dual publishing — that was not asked for.

D — CorelDRAW's text flow model makes a 280-page document far harder than it needs to be, and it is not free in any case.

47. Deriving a grid from the type, not from a template — B

A — Drifting margins would produce inconsistency between pages, not a uniform problem on every spread near the spine.

C — Measure affects reading across the whole line uniformly; it does not explain text physically disappearing into the spine.

D — Bleed governs artwork beyond the trim edge. It has no effect on where text sits relative to the spine.

48. Masters, running heads and why folios drift — C

A — A longer string changes width, not vertical position, and a variable does not alter the frame's placement on the page.

B — A section changes the number's style and starting value, not the geometry of the frame the marker sits in.

D — Plausible, but the question states all eleven pages use the same master; a derived master would show as a different letter in the pages panel.

49. Threaded text, overset, and copy that does not fit — A

B — Text placed on a master would appear on every page using it, and body copy is not placed on masters in a threaded document.

C — A non-printing frame would still be visible on screen with a distinctive marker, and non-printing is set deliberately rather than by accident.

D — Smart Text Reflow acts while editing, not during export, and it removes empty pages rather than pages carrying content.

50. Linked images, and the number that actually matters — B

A — The stored tag records an intended size, not a property of the pixels. It does not change when the image is scaled in the layout.

C — Miscounts: 420 mm at 300 ppi needs 4961 pixels, not 3543, which is the figure for A4 width.

D — Reaches the right verdict for the wrong reason. Upsampling adds interpolated pixels and no detail; it makes the file larger, not better.

51. Tables, and why tabs are not one — C

A — Tab stops are absolute positions too. The difference is not how the positions are stored but what happens when content exceeds them.

B — Tabular figures are an OpenType feature you enable; no table object applies them for you, and misalignment here is caused by text width, not figure width.

D — Tab stops are a paragraph property and survive a typeface change intact — which is precisely why the stops are unchanged in the question.

52. Contents lists and cross-references that renumber themselves — B

A — Contents generation reads paragraph styles across the whole document and has no relationship to which master a page uses.

C — Plausible-sounding, but the question describes ordinary headings in the text flow, and most generators do collect from threaded frames regardless.

D — Regeneration would pick up genuinely styled headings added later; the fact that these three remain missing after generation points at the styling, not the timing.

53. Five hundred different cards from one template — B

A — Failed shaping produces misplaced matras or split conjuncts, not question marks and boxes, and the characters would still be Devanagari.

C — A reasonable second guess, but a missing-glyph fallback gives uniform boxes, whereas the mix of question marks and boxes points to data corrupted before rendering.

D — Image paths have nothing to do with text rendering, and a broken link produces a missing image, not corrupted text.

54. Page counts, creep and the spine — C

A — Adhesive adds well under a millimetre and is not a fixed figure; the discrepancy here is far larger and scales with the page count.

B — The spine is built from leaves, so pages divided by two is correct. Signature size governs how sheets are folded, not the thickness calculation.

D — A perfect-bound cover is always supplied as one piece with the spine included, and the question describes exactly that layout being used.

55. The blue you chose on screen cannot be printed — D

A — Reaches for flattening, which changes how objects are rasterised but does not silently delete one.

B — Explains a missing element at the edge of a sheet, not one in the middle of a photograph.

C — Would leave a visible dark shape; here nothing printed at all, which points at ink behaviour rather than a colour value.

56. What a RIP actually does with your file — A

B — Flattening on export moves that work off the RIP rather than onto it; a timeout points at data volume the RIP still has to process.

C — Outlining a whole page of body text can cause this, but the question describes ordinary body text alongside two far more likely culprits.

D — Colour conversion is table-driven and fast regardless of which profile is used; it is not a plausible source of a timeout.

57. PDF/X, and what each flavour forbids — D

A — Every PDF/X flavour requires an output intent, including X-1a; it is one of the defining requirements of the family.

B — Subsetting is an export setting unrelated to the X flavour; X-1a neither requires nor forbids full embedding.

C — X-1a forbids optional content layers; X-4 is the flavour that permits them, and a shop cannot switch off flattened artwork in any case.

58. The five rectangles inside a PDF — A

B — Guides are not exported and carry no meaning in a PDF; nothing in the file records the intended trim.

C — A MediaBox is always present and the TrimBox is written from the document page size; crop marks are drawn marks, not the source of the boxes.

D — Nothing in the file distinguishes bleed from trim here, and no press defaults to cutting an assumed amount inside an undefined edge.

59. Preflighting your own PDF, free — C

A — Base-14 fonts are only listed when the file references them. A referenced font that is not embedded is a real dependency, whatever wrote the file.

B — The base-font mechanism exists, but relying on it means the press's own Helvetica substitute is used, with different metrics from whatever was intended.

D — Subsets report `yes` under `emb` and carry a six-letter prefix in the name; the two indicators are independent.

60. Reading separations, and the fifth plate you did not order — D

A — An output intent describes the printing condition; it does not add separations for colours nothing in the file uses.

B — Trapping modifies existing separations by spreading or choking; it does not create plates for inks the artwork never specified.

C — X-4 preserves spot colours. Nothing in any PDF/X flavour converts a spot into process as a side effect of transparency.

61. Paper, and everything done after printing — B

A — Heavy ink coverage makes cracking more visible, but a correctly grained fold on 170 gsm will not crack; the grain is what decides it.

C — 170 gsm folds without scoring in most stocks. Creasing becomes necessary well above this weight, and the defect described follows the grain rather than the weight.

D — Absorption softens the surface rather than stiffening it, and uncoated stock generally folds better than coated, not worse.

62. Approvals, and who pays for the reprint — C

A — A signature covers what the proof actually shows. A digital proof shows content, position and approximate colour, not the physical trimming of the final sheet.

B — Normal tolerance is around 1–2 mm. Four millimetres is beyond it, which is precisely why it is worth raising.

D — A border near the trim is a risky design decision, but it does not excuse a drift beyond the stated tolerance for the process.

63. CDR is the most common file in Indian print and the least portable — C

A — Imagines the version difference is a labelling issue rather than a change in the stored format.

B — Treats EPS as universal and lossless, when it flattens transparency and drops modern effects.

D — Assumes an open format carries print-specific data, when SVG has no CMYK, spot colours, bleed or overprint.

64. The covering note, and the file a shop can open — C

A — A verbal spec may be binding in principle and is worthless in practice two weeks later; the point of a written note is that neither party relies on memory.

B — No press workflow reads stock from PDF metadata, and no standard defines a field for it. The specification travels with the job, not in the file.

D — The client is not managing the job. Whoever supplies the file is the person who specifies it, and passing that to the client abandons the role.

65. The paid vector work is mostly machines, not posters — B

A — Colour can assign tools on some machines, but it cannot explain a blade cutting straight through the middle of a letter.

C — Applies a raster measure to a path-following machine that has no resolution at all.

D — Moves a file-preparation fault onto the material, and does not account for the blade cutting through the letters.

66. A vinyl job from file to wall — A

B — This is the welding failure, which happens once strokes have been outlined. With a live stroke the machine never sees the letter shapes at all.

C — Cutting software accepts the file and cuts what it finds. Silent wrong output is the normal failure here, not a rejection.

D — That is what happens after stroke-to-path is applied. Without it there is one path and therefore one cut line.

67. Print and cut, and the marks the machine looks for — D

A — Correctly describes the blade-offset symptom, and that symptom is corner shape, not a uniform translation of the whole cut.

B — Stretch produces an error that grows along the length of the sheet; a consistent shift everywhere is not a stretch signature.

C — A stroked cut path is still followed down its centreline, giving a correct position, and the cut would not be displaced uniformly.

68. Embroidery, which does not use vectors at all — D

A — Underlay affects how stitches sit and whether they sink, not whether a curve is smooth; the distortion described has a specific geometric cause.

B — Tension does not shrink a design uniformly. Fabric is drawn in across the stitch direction and pushed out along it, which is not a uniform scale.

C — Pull affects any stitched area, and a circular outline is typically a satin column, which is where pull is most pronounced.

69. Laser cutting, kerf, and the material you must never use — A

B — Charring removes material rather than adding it, and any thermal expansion has reversed by the time the part is handled.

C — Kerf applies to every cut the beam makes, inside and outside. Both features are affected, in opposite directions.

D — A unit conversion error would scale everything, not just the joint, and 0.4 mm on 6 mm is far larger than any rounding in a conversion.

70. CNC routing, tool radius, and the letters on a shop board — C

A — A second pass with a small bit is possible but slow, fragile and chargeable; it is a negotiated solution, not an omission on the fabricator's part.

B — Offsetting moves the cut but cannot make a round tool produce a corner sharper than its own radius; the geometry is not recoverable by offsetting.

D — PDF preserves geometry exactly. The rounding is a property of the cutting tool, not of the interchange format.

71. Screen printing, separations and the underbase — D

A — Missing flash-drying causes muddy colours and ink pick-up, not a clean white outline sitting proud of the coloured shapes.

B — Trapping between the two colours addresses gaps where they meet each other; it does nothing about white showing at the outer edge of both.

C — Printing white over the colours would simply obscure them. The underbase must go down first so the colours have something opaque to sit on.

72. DTF, DTG, sublimation and heat transfer — D

A — Sublimation is right for polyester but cannot print on a dark garment, since there is no white ink and the substrate must supply the white.

B — Screen printing works here technically, but a photographic design needs simulated process across six or more screens, which at twenty units is far too much setup.

C — DTG's water-based ink bonds to cotton, not polyester, so the design would wash out of a polyester shirt.

73. Signage, viewing distance and the seams — A

B — The 300 ppi figure applies at reading distance. At 30 m the eye cannot resolve anything close to that level of detail.

C — RIPs do not impose a source resolution floor, and internal upsampling adds no detail that would justify rejecting the file.

D — Reaches the right verdict on the refusal and then prescribes an upsample that adds interpolated pixels and no detail.

Module test — 1. The thing that is actually in the file

1. C

The file stored 354.33 unitless numbers and never stated what a unit was. Inkscape 0.91 wrote them at 90 units to the inch; Inkscape 1.x reads them at 96, so 354.33 divided by 96 times 25.4 gives 93.75 mm. Nothing is resampled and nothing is corrupt, because there are no pixels involved at all. This is why you measure one known edge after importing anything old.

2. D

Hairline is not a measurement. It resolves to one divided by the device's resolution, so it is 0.042 mm on a 600 dpi laser and 0.011 mm on a 2400 dpi platesetter. The second is thinner than a press can hold ink, so the line breaks into dots or disappears. Quote a real number instead: about 0.1 mm positive on offset litho, 0.15 mm reversed.

3. A

Under non-zero winding, two subpaths drawn the same way give a winding number of 2 in the middle, which is not zero, so it fills. Reversing the inner loop makes the counts cancel to zero and produces a hole. Switching to even-odd also works, but direction is stored in the geometry and travels with the file, while a fill-rule attribute is something some importers drop. Grouping never makes a hole, because a group is separate objects.

4. B

A clip is an instruction to render only part of an object. The object itself is untouched, so every pixel of the original travels in the file, bloats it, and costs the RIP time it decodes and throws away. It is also recoverable by anyone who releases the clip, which is why a clip is never redaction. Turn on crop image data to frames on export, or crop the picture in a raster editor.

5. D

A drop shadow is not geometry; it is a picture generated at a resolution the document sets, and the default on a new Illustrator document was Screen at 72 ppi for years. At 72 ppi it looks right on a screen that is roughly 72 to 110 ppi and prints as visible steps. Set it to 300 ppi on day one, and check it again whenever you open somebody else's file.

6. B

Every tracer reduces to regions, walks contours, decides which staircase steps are genuine corners, then fits curves. Rounded corners come from the third stage deciding your corner was sampling noise, so the corner threshold is the control. Loosening the fit makes it worse, and the speckle filter only discards small features. A wobbly straight edge, by contrast, is the fourth stage smoothing a staircase and wants a better scan.

Module test — 2. Drawing curves people will pay for

1. B

The curve between two anchors is set entirely by the two handles, and the handle direction is the direction the curve travels as it leaves the anchor. Twenty anchors around a circle means twenty tangent constraints, and every one that is slightly wrong shows as a kink. Four anchors at the extremes declare four tangents you can get exactly right, because at an extreme the correct handle is exactly horizontal or exactly vertical.

2. C

Editors enforce matching tangents and never matching curvature. Collinear handles of very different lengths give a join where the curve arrives bending hard and leaves bending gently, so the curvature jumps although no corner exists, and the eye reads the jump as a flat spot. Make the node symmetric rather than merely smooth, or delete the anchor and let one longer segment do the work.

3. D

An offset runs parallel to the original on both sides of the corner, and two parallel lines leaving a 20 degree point do not meet 5 mm out. They meet at 5 divided by the sine of 10 degrees, which is 5 divided by 0.174, or 28.8 mm. It is not a defect; it is what a parallel offset means. Round joins avoid it entirely and are the safe default on anything you cannot inspect.

4. B

A boolean finds every intersection, splits all the input paths there, decides which segments lie on the new boundary and stitches the survivors together. Each old crossing therefore survives as a node, with handles the solver chose rather than handles you chose, and the count compounds across operations. Clean up after each boolean rather than at the end, and do not reach for Simplify, which averages away the corners you meant to keep.

5. A

Every transform multiplies each point by a matrix and stores the result, so twelve rotations put every coordinate through twelve floating-point round trips. On screen that is invisible. On a laser cutting a slot to 0.05 mm, or on parts that have to fit each other, it is not. Type the total angle once, and set the pivot deliberately before you do.

6. D

A 2 mm stroke straddles the path, so it reaches 1 mm beyond the geometry on each side and the bounding box is 2 mm larger than the shape. Snap two of them together by their boxes and the paths themselves are 2 mm apart, which is why things that look aligned on screen print misaligned. Turn bounding-box snapping off and keep node-to-node, path and grid snapping.

Module test — 3. Type, which is also vector

1. C

Outlining replaces a font reference with geometry, so a page of body text goes from a few hundred references to tens of thousands of curves, and node explosion is one of the four things that make a job time out on a RIP. Embed instead: a PDF carries a subset containing only the glyphs used, the press sees your typeface, and no font can go missing. Outline logos, headlines and anything going to a machine.

2. B

fsType lives in the OS/2 table and is the embedding permission bit: 0 is unrestricted, 2 forbids embedding, 4 allows preview and print, 8 allows editable embedding. Your export obeys it. Almost everything commercial and everything under the SIL Open Font Licence is 0 or 4, so a refusal usually means a font of doubtful origin. Read it with ttx before you build a job around the face.

3. A

The shapes belong to whoever drew them whether or not a font file is attached. In practice most desktop licences already permit output as outlines in final artwork, so the question rarely arises. Where the licence genuinely bites is the other direction: sending the printer the font file itself is distributing software, which is a different permission from using it, and most licences have a clause covering exactly that case.

4. C

Spacing is equal white area, not equal distance. Two verticals present a full-height rectangle of white, which is the least area for a given distance, so they need the most room. Two diagonals present a wedge, which is already a lot of white, so they need much less. The ranking from loosest to tightest runs verticals, vertical against round, two rounds, diagonals, then anything against a full stop.

5. B

Type on a path is positioned by arc length along the baseline while the glyph itself is rigid. On a circle of radius r the baseline has circumference 2 pi r and the cap line has 2 pi times r plus the cap height, so the tops are pulled apart and the bottoms crowd. No setting fixes it. A larger radius, more tracking, and for a badge that has to be right, outlining and rotating each letter by hand.

6. D

Shaping happens between the characters and the drawn glyphs, and an application can use one engine to display and another to export. That is the failure that costs money, because it is invisible until a proof arrives. A font missing its conjuncts would be wrong on screen as well, and would show a visible virama. Test with a real string of hard cases and have somebody who reads the script check it before anything is outlined.

Module test — 4. Colour that survives the press

1. C

The eye reduces a whole spectrum to three numbers, so two physically different spectra can produce the same three numbers under one light and different ones under another. A mixed ink and a four-ink build are different spectra by construction. This is why print is judged under a standard illuminant, normally D50 at 5000K, and why a colour approved under an unknown bulb was approved against nothing.

2. B

Assign changes the interpretation and leaves the numbers alone; convert changes the numbers to preserve the appearance. Here the numbers are already correct Adobe RGB values being read as if they were sRGB, so the tag is wrong and assigning the right one fixes it with no loss. Converting would take those misread values as truth and rewrite them, and raising saturation invents colour that was there all along.

3. D

Relative colorimetric leaves every reproducible colour exactly where it is and clips only what will not fit, which is what artwork with named colours needs, and black point compensation should stay on. Perceptual shifts in-gamut colours too, in order to make room, which is right for a photograph and wrong for a logo. Saturation is for business charts. Absolute is for proofing one press on another.

4. A

A press profile already contains the correction for dot gain, which is about 15 per cent at the midtone on coated and 20 to 25 on uncoated. Convert with the coated profile and the plate values are set for a press that spreads less than yours does, so every midtone comes out heavier than intended. Dot gain is only a problem when you use no profile or the wrong one.

5. D

Trapping replaces a possible white gap with a slight overlap, and the eye forgives an overlap. Moving the lighter colour outwards leaves the perceived boundary where it was, because the darker shape's edge is what the eye is reading. Spreading the dark colour instead visibly fattens the shape. Typical widths are 0.08 to 0.15 mm on sheetfed offset, and on most commercial work the printer's RIP does it, not you.

6. A

Overprint means print this on top rather than knock the background out. White in CMYK is 0/0/0/0, so overprinting it lays no ink at all and the photograph behind shows through. A knockout is the correct behaviour for white and is the default. Black text overprints deliberately, so that misregistration leaves no white fringe; white must never overprint. Turn on overprint preview before every export.

Module test — 5. Designing a mark

1. C

The list is not aesthetic. A shirt pocket, a rubber stamp on a delivery challan, an engraved pen, a thermal invoice, a newspaper classified and a vinyl shutter are ordinary requests from ordinary clients, and not one of them can reproduce a gradient, a shadow or a transparency. If the mark works flat and black, colour can be added afterwards. If it does not, colour will not rescue it.

2. D

Everywhere is the union of the worst constraints on the surfaces list: small, one colour, no fine detail, legible at distance and at 16 pixels. That is a design decision derived from an answer rather than from taste, which is what makes it defensible later. A graded set of four marks is for organisations with somebody to police them; a small client needs two at most.

3. C

An abstract form is distinctive, ownable and works at any size, which are all real advantages, and it is usually strong rather than weak as a trademark. What it has on day one is no meaning at all, and meaning is bought with years of consistent exposure. Recommending one to a business with no budget for that is a promise to spend money they do not have.

4. A

A rule expressed as half the icon's height, or as one cap height of the wordmark, is true at 12 mm and at 12 metres. A rule expressed as 6 mm is true at one size and wrong everywhere else, and somebody will apply it at the wrong one. Draw the rule in the usage sheet with the unit itself shown beside it, because people follow a picture and skim a sentence.

5. B

An inversion turns the background into a solid rectangle of the opposite colour. A knockout has a transparent background and white geometry, so the surface it sits on shows through the counters. There is a second job to do as well: light on dark closes counters optically and ink closes them physically, so a reversed mark often needs its counters opened by ten to fifteen per cent, which means two slightly different drawings.

6. D

Hex describes what one screen does and a CMYK build describes what one press does on one paper, so neither travels. Lab is defined by measurement of human vision rather than by any device, so a printer in another city can measure the sheet and tell you how far off it is in Delta E. Almost nobody puts it on a brand sheet, and it is the value a second supplier can actually hit.

Module test — 6. Documents that survive a late change

1. B

Direct formatting beats a style, so a paragraph somebody selected and set by hand no longer follows the named decisions. InDesign shows a plus beside the style name when the selection has overrides, and Alt-clicking the style applies it and clears them at the same time. The discipline that prevents it is formatting only by applying a style, and making a new style whenever you want a variation.

2. B

Size and leading first, then a measure of 45 to 75 characters, which at 10 pt in a typical text face is roughly 80 to 110 mm. That is the column. Columns and gutters follow, and the margins are what remains, adjusted so the outer is larger than the inner and the bottom larger than the top. Doing it in the other order is why so many layouts have uncomfortable line lengths.

3. A

Three thousand pixels divided by 500 mm in inches, which is 19.685, gives about 152 ppi. That is marginal for good commercial print, where the rule of thumb is about twice the line screen. The pixels never change; only the physical size you placed them at does, which is why a stored dpi tag is a note about intention rather than a property of the image.

4. B

Overset copy is still in the story and simply not displayed, so a printed proof shows a document that reads as complete. Nobody notices until a reader does. Every application marks it, with a red plus at the out-port in InDesign or a red cross in Scribus, and it appears in the preflight panel. Check for it before every export, every time, which is what the preflight panel is for.

5. C

A generated list reads the paragraph styles and reads the page each one actually sits on, so it is true by construction once it is regenerated. Two things follow. Headings formatted by hand rather than by style are missing from it, which is the commonest failure, and you still have to regenerate after the pages move. The leader dots are a tab leader, which is a separate and equally worthwhile habit.

6. A

One folded sheet is four pages, two on each side, so every folded job is a multiple of four; larger presses run 8, 16 or 32 to a sheet, so ask what the signature size is before promising a page count. The spine is half the page count multiplied by the caliper of one sheet, and caliper is thickness, not weight, which is why a bulky cream paper makes a thicker book at the same gsm.

Module test — 7. The handover

1. B

A RIP has finite memory and a time budget, and four things break it: node explosion, deeply nested transparency, enormous embedded images clipped down to a small frame, and hundreds of clipping paths. All four are fixed the same way, by simplifying before you export. None of them is visible in a screen view of the PDF, which is why the file can look perfect and still refuse to run.

2. D

Every PDF/X flavour embeds the fonts, defines a TrimBox and names an output intent; those are the point of the family, not the difference between them. X-1a requires everything to be CMYK or spot and flattens all transparency before export. X-4 permits colour-managed content and live transparency, so the RIP flattens once, at device resolution, knowing the device. Ask the shop which they want.

3. C

The press positions and cuts from the TrimBox, and no trim box defined is one of the most common preflight rejections there is. The fix is upstream: set the document page size to trim, set bleed in the document setup, and tick use document bleed settings on export. Then check with pdfinfo -box, where the arithmetic takes ten seconds and either matches what you agreed or does not.

4. D

If black text were 100 per cent K it would be absent from the cyan, magenta and yellow plates entirely. Reading it there means it is built from four inks, usually because it was defined as RGB black and converted, or pasted in from another document. Four plates never register perfectly, so small four-colour text gets a coloured fringe. Fix it at source with a 100 per cent K swatch, not in the export.

5. A

Total area coverage is the sum of the four percentages, and presses cap it: roughly 300 to 330 per cent on coated sheetfed, 260 to 300 on uncoated, 240 on newsprint and web. Above the limit the ink does not dry and sheets set off onto each other in the stack. A 60/40/40/100 build totals 240 and is safe nearly anywhere. Ghostscript's inkcov device reports the figure for nothing.

6. C

An older CorelDRAW generally cannot open a file written by a newer one, and that is the single most common file failure in the trade. A press-ready PDF is the deliverable in nearly every case. If they genuinely need to edit, use the version dropdown in Save As and check the result, because saving down can drop newer effects. EPS carries no transparency and SVG carries no CMYK, spot colours, bleed or overprint.

Module test — 8. Machines that cut, stitch and burn

1. A

A stroke is a rendering instruction attached to a path of zero width, not an object. The plotter follows the path, so a 0.5 pt outline around a letter is cut as one line through the middle of where you expected two. Convert strokes to outlines before anything goes to a machine, along with converting text to curves and welding overlaps, and keep the editable original.

2. D

After cutting, somebody picks the waste vinyl out with a hooked tool, and anything narrower than about 2 mm tears while being weeded. Fine serifs, thin script connectors and small counters are not a stylistic risk here; they are physically impossible. Counters come away as separate pieces held only by the transfer tape, so below about 5 mm across they get lost. Ask what the shop can weed in that material.

3. B

The RIP recognises the swatch name, declines to print it, and passes the geometry to the cutter. CutContour is the usual name in large-format work and Dieline in packaging, but the shop's RIP is configured for one of them, so ask. Overprint matters because a knocked-out cut path punches a white outline through the artwork underneath. Registration marks are placed by the RIP and never drawn by hand.

4. B

As the needle draws thread through, the cloth is pulled inward across the stitch direction, so the shape finishes narrower than drawn, and it grows slightly along the stitch direction. A digitiser compensates by widening shapes, typically 0.2 to 0.3 mm each side. Knitted fabric pulls far more than woven, so the same file on a polo and on a shirt gives two different finished widths, and only a sew-out settles it.

5. C

The beam removes material as it travels, so a cut-out hole finishes one kerf larger than drawn and a cut-out part one kerf smaller. Two parts meant to meet are therefore loose by two kerfs, which at 0.22 mm is about 0.44 mm and a visibly wobbly joint. Measure your own kerf by cutting a 20 mm square from the real material and comparing the square with the hole.

6. B

Lasering PVC produces hydrogen chloride gas, which attacks the machine's optics and rails and is seriously harmful to breathe, and reputable shops refuse the material outright. Sign vinyl is PVC, which is the trap, because it is the material most likely to be lying next to the machine. Also refuse polycarbonate, ABS, fibreglass and anything whose composition the client cannot state.

Vectors, Logos and Things That Get Printed: final examination

1. A *1. The thing that is actually in the file*

A cubic Bezier segment is written as two handle positions followed by the end point, which is why C takes six numbers. The handles never lie on the curve; they set the direction the curve leaves the first anchor and the direction it arrives at the second. Reading a d attribute out loud is the fastest way to stop thinking of a vector file as a better kind of picture.

2. C *1. The thing that is actually in the file*

An image element carrying a data URI means somebody placed a photograph or a screenshot inside what they called vector artwork, and it is the commonest reason a supplied logo fails when the printer enlarges it. Grep a whole folder for base64 to find the fake vectors in one command. Node count, decimal precision and embedded rasters are the three things that inflate a file, and only the third is fatal.

3. B *1. The thing that is actually in the file*

Dots per inch is a property of an output device or of a grid of pixels. A vector file stores unitless numbers plus a rule for how many make an inch, and no dpi at all, so the question has no answer as asked. What the client needs is to state a finished size in millimetres; everything else follows from that, including whether a raster would have done.

4. D *1. The thing that is actually in the file*

Every application has a preference deciding whether a stroke scales with the object, and both behaviours are useful: on is what a logo wants, off is what a die-line or a technical drawing wants, because 0.25 mm there means 0.25 mm at any zoom. With it off, a 2 mm outline on a mark reduced by a factor of thirteen is proportionally thirteen times too heavy, and nobody can say why it looks wrong.

5. C ***1. The thing that is actually in the file***

A cap is part of how the stroke is rendered, not a decoration on the end of it. Butt caps stop exactly at the endpoint; round and square caps extend beyond it by half the stroke width, so a 4 mm stroke adds 2 mm at each end. It is a common reason a dimension you set carefully comes out long, and it matters for anything cut, engraved or routed to a measurement.

6. A ***1. The thing that is actually in the file***

A clip, a mask, a white box and a black box are all just objects drawn over something that is still in the file. Newspapers have published documents where the redacted text came out with copy and paste. If content must be gone, delete it at source or flatten the page to a raster, check the raster, and place that. The same mechanism makes a cropped photograph recoverable from an exported PDF.

7. D ***1. The thing that is actually in the file***

A colour trace does not produce a jigsaw of adjacent regions. It produces overlapping opaque shapes, largest at the back, each partly hidden by the one in front. The overlaps are invisible on screen and are real geometry, so a blade cuts both edges. The same structure is why recolouring the top shape reveals the one beneath and why deleting the background shape puts holes in everything.

8. B ***2. Drawing curves people will pay for***

The handle is a compass and a speedometer: direction says which way the pen is travelling as it leaves, length says how long it keeps going that way before turning towards the next anchor. Dragging towards the inside of the bend points the tangent the wrong way, so the curve has to turn back on itself. Drag along the direction of travel, and hold Shift at an extreme.

9. A ***2. Drawing curves people will pay for***

A primitive is mathematically exact for nothing, and at six metres or in a 16 pixel favicon a slightly wrong circle is what people see without being able to name it. Booleans are in fact destructive in most applications, which is the reason for keeping a locked source layer. The pen tool comes out for the parts no primitive can make.

10. C ***2. Drawing curves people will pay for***

Which input's appearance the result takes differs between applications, differs between operations, and has changed across releases of the same application. Illustrator, Inkscape and CorelDRAW all answer it differently, and testing is more reliable than remembering. Setting the colour afterwards makes the question irrelevant, and it stops costing five minutes a day.

11. B ***2. Drawing curves people will pay for***

A 2 mm bar offset inwards by 1.5 mm leaves a 0.5 mm bar; at 2 mm it has nothing left and most solvers produce a small backwards loop rather than nothing. On an embroidery underlay that loop is stitched as a knot, and in a screen-print underbase a fine line disappears. Measure the narrowest part of the shape before choosing the distance, then look at the result.

12. D ***2. Drawing curves people will pay for***

Scaling a selection scales the whole arrangement, so the circles get smaller and move closer together. Transform Each scales every object about its own centre and moves none of them. It is the difference between make the pattern smaller and make the dots smaller, and choosing the wrong one is how a carefully spaced layout collapses in one drag.

13. C ***2. Drawing curves people will pay for***

Snapping every edge onto a grid of 96 divisions to the inch is exactly what an app icon wants and exactly what a 90 mm card does not, since it introduces error on every edge for no benefit at all. The setting is on by default in some web presets, which is why print files built from a web template quietly acquire it. Check it with the document settings, once, at the start.

14. A ***2. Drawing curves people will pay for***

A template that has moved half a millimetre invalidates everything drawn against it afterwards, and you will not notice until the overlay test at the end. Lock it, set it to about 30 per cent, and verify with the overlay, a horizontal flip, a 16 pixel export and a node count. Under fifty nodes for a typical mark; two hundred means it was traced rather than drawn.

15. D ***3. Type, which is also vector***

TrueType stores quadratic curves with one handle per segment, and converting them to the cubic form your drawing program uses produces more nodes than the designer placed, at positions nobody chose. An OTF with CFF outlines is already cubic, so an o arrives with eight nodes rather than fourteen. Hinting is discarded either way, because there is no text left to hint.

16. B ***3. Type, which is also vector***

A six-letter prefix marks a subset, which is the normal and desirable case: only the glyphs actually used are carried, so a forty-page document in a 400-glyph face adds tens of kilobytes rather than a megabyte. It also means nobody can extract a usable font from your PDF, which is part of why foundries permit embedding at all. An empty emb column is the failure to look for.

17. A ***3. Type, which is also vector***

Desktop, web, app and broadcast are licensed separately, and a licence held by the client does not automatically extend to their suppliers. If the answer is no or unclear, set the job in a licensed alternative and show them both; you are keeping their exposure off your invoice rather than being difficult. Outlining is not the issue, and plenty of commercially licensed fonts are not OFL.

18. C ***3. Type, which is also vector***

Justification stretches word spaces to fill each line, and at a short measure there are few spaces to share the work, so they stretch a long way and line up by chance into rivers. A longer measure, hyphenation and controlled H and J values all help, and ragged right removes the problem entirely. Rivers are a spacing artefact; ladders, widows and a poor rag are three different faults.

19. B ***3. Type, which is also vector***

Most OpenType text faces carry more than one set of numerals, and proportional figures give each digit its natural width, which is right in a sentence and wrong in a column. Tabular figures make every digit the same width, and the column then aligns with no tabs at all. Decimal tabs are worth setting as well, but they cannot rescue digits of different widths.

20. D ***3. Type, which is also vector***

When a font lacks a character, an application either shows the empty box, which is obvious, or silently substitutes another font for that one character, which is not. The rupee sign was adopted in 2010, so fonts released before then do not have it and fonts released since do not all have it. Set the sign in the real face and look, and check the exported PDF with pdffonts for a font you did not choose.

21. C ***3. Type, which is also vector***

Scaling to 92 per cent thins the vertical strokes by eight per cent and leaves the horizontals alone, so the contrast of the face changes and the round letters go oval. A width axis interpolates between masters the designer drew, redrawing the counters and keeping the stroke weights. Same instinct, and only one of them is correct. The font is instanced and embedded as an ordinary static font at export.

22. A ***4. Colour that survives the press***

In a four-colour process job the white is the paper, and the mark is an absence of ink. That is why the knockout version of a logo is transparent geometry rather than white geometry, and why white must never be set to overprint. On coloured stock, a dark garment or transparent film there is no paper white to use, and a real opaque white ink on its own station is needed.

23. D ***4. Colour that survives the press***

CIE76 and Delta E 2000 give different numbers for the same pair, and 2000 is usually smaller, so a figure without a formula is incomplete. The measurement condition matters too: instrument illuminant, backing and whether the paper contains optical brighteners all move the reading, which is why the M1 condition exists. A supplier who can answer both is telling you something useful about how they work.

24. B ***4. Colour that survives the press***

A CMYK file converted with the wrong press profile is simply wrong, and the differences between coated and uncoated profiles are large enough to see. Asking takes one line in an email. If nobody can answer, ISO Coated v2 is a defensible default for coated work and Japan Color 2001 Coated is what a great many Asian shops are set to, but the question beats the default every time.

25. A *4. Colour that survives the press*

Saturated blues and violets, bright greens, vivid oranges and anything fluorescent sit outside what four process inks can make, and the corners of the RGB cube are the worst offenders. A colour chosen near the cyan primary survives a conversion with almost no shift, everywhere, for the life of the brand. A colour chosen in the corner commits the organisation to spot inks or to permanent disappointment.

26. C *4. Colour that survives the press*

Four fine regular patterns laid on top of each other at the same angle interfere and produce moire, a coarse pattern that was in neither. Offsetting the angles, conventionally cyan 15, magenta 75, yellow 0 and black 45, makes them form the small rosette you can see with a loupe. Black takes 45 because a diagonal pattern is least visible, and yellow takes 0 because it is palest.

27. B *4. Colour that survives the press*

One plate means one hue at any percentage. The solid, something around 60 to 70 per cent, something around 30 to 40, a tint panel at 10 to 15, and the paper reversed out of a solid, which is your second colour for nothing. Below about 10 per cent, dot gain and the press minimum dot make it unreliable. Most one-colour work uses two of those five and looks impoverished for no reason.

28. D *4. Colour that survives the press*

That photograph has been through the room's illuminant, the camera's white balance and tone curve, its HDR processing, JPEG compression and their screen. It is evidence of almost nothing. The only comparison that means anything is two physical objects under one known light, which is why the printed reference was agreed at the start and why the Delta E conversation comes after it, not before.

29. C *5. Designing a mark*

The surfaces list converts taste into constraints. A 3 m shop board, an embroidered pocket, a rubber stamp, a 16 pixel app icon and a sponsor board at 40 mm each rule out a class of design, and together they usually decide flat, simple and one-colour-capable. The competitor sheet and the examples matter too; the adjectives are the least reliable input in the room.

30. A ***5. Designing a mark***

Showing one strong concept beside two weak ones to make the strong one look better works until it does not, and then you are producing the weak one. One looks unexplored; five or more makes the client feel responsible for decisions they cannot evaluate, and usually produces a request to combine the top of two with the bottom of three. Three, each shippable, is the number that survives contact.

31. D ***5. Designing a mark***

You can overlay golden-ratio circles on nearly any shape if you may choose where to put them and how large to draw them, and Apple's designer has said publicly that the ratio was not used there. It matters because it teaches beginners that good marks come from a mathematics they have not learned, so they stop trusting their eye, and because optical correction contradicts it directly.

32. B ***5. Designing a mark***

Geometry and perception disagree and perception wins, which is why the play button on every media player is not mathematically centred. The same family of corrections includes overshooting a circle beside a square by one or two per cent and raising a horizontal bar above the exact vertical centre. Do it by eye, then record what you did on the construction sheet so the next supplier does not correct it back.

33. A ***5. Designing a mark***

A mark that exists in one arrangement will be rearranged by somebody else, badly, to fit a tall banner or a narrow column. Three or four lockups, each drawn and spaced deliberately, prevent it, and the internal spacing in each is expressed as a multiple of something inside the mark so it stays true at 12 mm and at 12 metres. Two versions used correctly beat four nobody maintains.

34. C ***5. Designing a mark***

Satin stitch works between about 1.2 and 12 mm, and text below about 5 mm cap height smudges, so a mark with fine barbs or a small tagline is not recoloured into a one-thread version; it is redrawn with details removed and thin strokes thickened. Budget for it if the client has uniforms. Heat transfer vinyl is the process that is cut mirrored, not embroidery.

35. B ***5. Designing a mark***

Copyright arises in the artwork and protects against copying the drawing; a trademark protects a sign used in trade and is limited to its registered classes. Which of them the client gets, and when, is jurisdiction-dependent and has to be settled in writing, which is three sentences in the quotation. If generative tools were used, ownership of the output is unsettled, and the honest answer is that their protection comes from registration and use.

36. D ***6. Documents that survive a late change***

A word processor does not do bleed, and every workaround for that is worse than changing tool. The others are all defensible: Scribus's PDF/X export and colour management are the strongest of the free tools, LaTeX is the right answer for a thesis, and CorelDRAW is perfectly reasonable up to about twenty pages. Choose from the document's requirements, not from a ranking.

37. C ***6. Documents that survive a late change***

Most applications require a modifier, such as Ctrl and Shift and a click in InDesign, to select a master item on a page, precisely so this is hard to do by accident. Once it happens, that item is local and stops following the master. Remove All Local Overrides puts them back. The other causes, in order of frequency, are the wrong master applied and two masters built separately rather than one based on the other.

38. A ***6. Documents that survive a late change***

Horizontal scaling distorts the letterforms and is visible beside any undistorted paragraph on the same spread. Editing the copy is usually the best answer and the one nobody wants; a global style change is legitimate because it keeps the document consistent; a small amount of tracking on one paragraph is acceptable in small doses. Whatever you do, do it through a style rather than by selecting text.

39. D ***6. Documents that survive a late change***

A table spanning four pages should carry its headings onto each, and doing that by hand means they are wrong the moment a row is inserted. Cell styles inside a table style mean one definition for every table in the document, and alternating fills set in the style survive an inserted row. Tabular figures come from the font, and they are what actually makes a price column line up.

40. B ***6. Documents that survive a late change***

Interpolation adds pixels by averaging the ones that exist, so the file grows and the edges soften and no information appears. The one nuance worth stating is that machine-learning upscalers genuinely invent plausible detail, which is acceptable for a decorative background and not for a face, a product or anything where invented detail could be wrong. Say which you used.

41. A ***6. Documents that survive a late change***

Each sheet wraps around the ones inside it, so the inner pages push outwards and lose more at the trim, by three to five millimetres at the centre of a 64-page booklet on normal paper. Many presses compensate during imposition by shifting pages inward progressively, so ask whether yours does. If not, widen the margins, and keep folios well inside the fore-edge either way.

42. C ***6. Documents that survive a late change***

Anything that refers to a position in a document should be generated, because the moment it is typed it becomes something that will quietly become untrue. A live cross-reference follows the figure when pages move, renumbers with the caption's numbered list, and the panel flags any reference whose target has been deleted, which is a real safety net in a technical manual.

43. B ***7. The handover***

A guillotine cutting five hundred sheets drifts by a millimetre or two, which is why bleed exists outwards and a safe area exists inwards. A background photograph pulled to the bleed line survives that drift invisibly. A border on the trim line turns the same drift into a visible asymmetry, wider on one side than the other, on every copy, and nothing at the press can prevent it.

44. D ***7. The handover***

A rich black is four inks, and four impressions never land exactly on top of each other. On a large solid panel that drift is invisible and the extra inks give a genuinely deeper black. On 8 pt text it puts a coloured fringe around every letter. Small text is 100 per cent K only, and it should also be set to overprint, so that misregistration leaves no white gap either.

45. C ***7. The handover***

Overprint simulation is off by default in the free reader on some platforms, which is exactly why a white object set to overprint looks correct on screen and vanishes on the sheet. The composite view also looks right when your black text is built from four plates. Acrobat Pro's Output Preview shows both, and Ghostscript will simulate overprint and write one file per plate for nothing.

46. A ***7. The handover***

The usual sources, in order, are an RGB object nobody converted, a spot swatch used once somewhere invisible, a Registration swatch used as though it were black, and two spot swatches whose names differ only in capitalisation, which a RIP reads as two plates. Delete unused swatches before export and look at the separations afterwards, because a fifth plate is a line on the invoice.

47. D ***7. The handover***

Preflight answers structural questions: boxes, fonts, resolution, ink, separations, overprint. It has nothing to say about whether the phone number is right, the photograph is of the correct building or the price is current, and those are the errors that actually get jobs reprinted. Print the PDF, read every word out loud, and have somebody who did not design it read it too.

48. B ***7. The handover***

Every finishing process follows the same convention: its own layer above the artwork, coloured with a named spot swatch such as SpotUV, Foil, Dieline or Crease, and set to overprint so it does not knock a hole in what is beneath. It is a presence-or-absence mask, so keep it at 100 per cent rather than tinting it. Confirm it appears as its own separation, and name the layers in the covering note.

49. A ***7. The handover***

Paper folds cleanly along the grain and cracks across it, so a folded job with the fibres running the wrong way produces a ragged fold on every single copy. Grain is the printer's choice to make, and they can only make it if they know where the folds fall. It is one line in the covering note, alongside weight, surface, finishing and quantity.

50. C ***8. Machines that cut, stitch and burn***

The through-line for every machine is that the file tells it what to do rather than what to look like. Closed paths, because an open cut leaves the piece attached. Strokes converted to outlines, because a blade follows a centreline. Overlaps welded, because two touching shapes are two closed paths. And a minimum feature size the material can actually hold, which differs by process.

51. B ***8. Machines that cut, stitch and burn***

A plotter slows down for every node it has to hit, so a traced outline with thousands of them turns a ten-second cut into several minutes and produces a visibly rougher edge because the blade is constantly changing speed. It is one of the practical reasons a traced logo is redrawn before it goes to a machine, along with the wobble the trace introduced in the first place.

52. D ***8. Machines that cut, stitch and burn***

The blade trails behind the centre of its holder and swivels to follow the direction of travel, typically by a quarter to half a millimetre. The machine overcuts slightly at corners to compensate, and if the offset value is set too small the blade has not finished turning and corners come out rounded; set too large, they overshoot and leave small tails. It is a calibration, and every machine has a test-cut pattern for it.

53. C ***8. Machines that cut, stitch and burn***

Too narrow and consecutive needle penetrations sit almost on top of each other, cutting the cloth into a line rather than covering it. Too wide and long unsupported threads catch on things and sag away from the fabric, so a wide area is split or filled with tatami instead. Those two limits, plus a floor of about 5 mm cap height on text, decide whether a mark can be embroidered at all.

54. A ***8. Machines that cut, stitch and burn***

A spinning cylinder cannot cut an internal corner sharper than its own radius, so a 6 mm bit leaves 3 mm radii. Drawing the radius in means the result is what you intended; a dog-bone fillet is a small circular overcut at the corner so a square tab still fits, and it is standard practice in CNC joinery. Ask the fabricator the bit diameter before you draw, and check every internal angle against it.

55. D ***8. Machines that cut, stitch and burn***

Registration on a manual carousel is measured in whole millimetres rather than tenths, so the colour layers will not land exactly on the white. Shrinking the white slightly means a small misregistration shows fabric rather than a white halo. Check the thin parts afterwards, because an inward offset eats narrow features, and expect traps of half a millimetre or more on the same job.

56. B ***8. Machines that cut, stitch and burn***

Sublimation bonds into polymer fibres and has no white ink at all, so the substrate supplies the white and it cannot print light on dark. DTG is ruled out separately, because it needs cotton. For twenty mixed dark polyester shirts the answer is usually DTF, which works on almost anything and generates its white backing from the alpha channel, provided the transparency in the file is hard-edged.