

3D - Creality

Notas de trabajo con la Creality K2 Pro y Creality Print: perfiles, diagnostico, materiales y conceptos. Escrito sobre casos reales, con las medidas que los respaldan.

- [Creality Print](#)
 - [Los perfiles de usuario no se aplican solos](#)
 - [Cambiar de filamento: lo que cambia sin avisar](#)
- [La K2 Pro](#)
 - [Interrogar la impresora con Moonraker](#)
- [Conceptos basicos](#)
 - [Que es el brim](#)

Creality Print

El laminador: perfiles, presets y las trampas que cuestan una impresion.

Los perfiles de usuario no se aplican solos

Tengo perfiles personalizados en Creality Print desde hace semanas. El 30 de agosto de 2026 perdí dos impresiones seguidas y, al investigarlo, descubrí que **esos perfiles existían pero no se estaban usando**: el laminador trabajaba con los de fábrica sin decir nada.

Esta página explica cómo detectarlo y cómo evitarlo.

El síntoma

Dos impresiones canceladas con hilos volando, con el ventilador de pieza a tope y la cama notablemente fría al tacto. Mi primera sospecha fue que la aplicación me había «pisado» o corrompido los perfiles.

No era eso. Los perfiles estaban intactos en disco. Simplemente **no eran los seleccionados**.

La comprobación que decide: el gcode, no la interfaz

La interfaz muestra el nombre del preset en un desplegable, pero ese desplegable puede no reflejar lo que acabó dentro del fichero. La única prueba concluyente está en la cabecera del gcode que la máquina ejecuta:

```
; print_settings_id    = 0.20mm Standard @Creality K2 Pro 0.4 nozzle - ABK  
; filament_settings_id = "Hyper PLA @Creality K2 Pro 0.4 nozzle - ABK"
```

Si esas dos líneas **no terminan con el sufijo de tu preset**, estás imprimiendo con los ajustes de fábrica.

El gcode se puede descargar de la propia impresora sin tocar nada, porque la K2 Pro expone Moonraker:

```
curl -s "http://<IP-de-tu-impresora>:7125/server/files/list?root=gcodes"
curl -s "http://<IP-de-tu-impresora>:7125/server/files/gcodes/<fichero>.gcode" \
| grep -aE "^; (print|filament)_settings_id"
```

Qué te llevas si caes a fábrica

Estos son los valores del perfil de fábrica de la K2 Pro que más daño hacen en piezas pequeñas:

- **Cama a 50 °C** — es el **mínimo** del rango del filamento, no un valor neutro.
- `close_fan_the_first_x_layers` = **1** — el ventilador de pieza arranca al 100 % en la capa 2, antes de que la primera capa haya agarrado.
- `fan_min_speed` y `fan_max_speed` = **100** — el ventilador va a tope siempre. No es una avería: es el perfil.
- `outer_wall_speed` = **200 mm/s** — la velocidad del perímetro exterior, que es el que define la cota de encaje de la pieza.

Por qué pasa: cada material necesita SU preset

El agujero que me tumbó las impresiones era este: yo tenía un preset de filamento personalizado, pero de `Hyper PLA-CF` (fibra de carbono), un material que no uso. El día que cambié de bobina a Hyper PLA normal, **no había ningún preset propio al que caer**, y el laminador tomó el de sistema.

La regla que saco de ahí:

- **Un preset de un material que no imprimes no protege nada.** Cada material que uses de verdad necesita su propio preset personalizado.
- Y ojo con los nombres: la lista de perfiles trunca el texto **por el medio**, así que `Hyper PLA @Cr...4 nozzle - ABK` y `Hyper PLA-CF @... nozzle - ABK` se leen casi igual en pantalla. La elipsis se come justo el `-CF`, que es lo único que los distingue.

Los ficheros `.info`: tus presets viajan a la nube

Cada preset creado desde la interfaz genera un fichero hermano `.info` junto al `.json`:

```
setting_id = 6a7c3ed7675aeed975583544
user_id = <tu id de Creality Cloud>
base_id = GP004
updated_time = 1786527762
```

Ese `setting_id` es el vínculo con tu cuenta de Creality Cloud. En el log de la aplicación se ve la sincronización explícitamente:

```
SyncUserPresets downloadUserPreset start...
ProfileList [https://api.crealitycloud.com/api/cxy/v2/slice/profile/user/list]
```

Consecuencia práctica: **tus presets de usuario pueden sobrescribirse desde la nube**. En mi caso no ocurrió, pero el mecanismo existe y conviene saberlo.

Corolario: si borras un preset solo del disco, puede reaparecer en la siguiente sincronización. Para eliminarlo de verdad, bórralo **desde la aplicación**.

Escribir presets a mano

Se puede, y a veces es lo más rápido. Los ficheros viven en:

```
~/Library/Application Support/Creality/Creality Print/<version>/user/<tu-id>/
├─ filament/
└─ process/
```

Precauciones que no son opcionales:

- **Cierra Creality Print antes de escribir.** La aplicación reescribe los ficheros al salir y machacará tus cambios.
- **Haz copia de seguridad** de la carpeta antes de tocar nada.
- **Los valores van como listas de cadenas** (`["60"]`), nunca como números. Un valor numérico rompe el parser, que hace operaciones de texto sobre él.
- Un preset escrito a mano **nace sin** `.info`, es decir, sin registrar en la nube. En mi caso sobrevivió y la aplicación lo cargó sin problema, pero la vía segura para que persista es crearlo desde la interfaz.

Ejemplo de preset de filamento válido, que hereda del de sistema y solo cambia lo necesario:

```
{
  "base_id": "GFSA04",
  "close_fan_the_first_x_layers": ["3"],
```

```
"during_print_exhaust_fan_speed": ["60"],
"filament_settings_id": ["Hyper PLA @Creality K2 Pro 0.4 nozzle - ABK"],
"from": "User",
"hot_plate_temp": ["60"],
"hot_plate_temp_initial_layer": ["60"],
"inherits": "Hyper PLA @Creality K2 Pro 0.4 nozzle",
"name": "Hyper PLA @Creality K2 Pro 0.4 nozzle - ABK",
"slow_down_layer_time": ["14"],
"textured_plate_temp": ["60"],
"textured_plate_temp_initial_layer": ["60"],
"version": "26.8.4.20"
}
```

Higiene: los presets duplicados son una trampa

Al duplicar un perfil sin renombrarlo quedan nombres como `- Copiar(1)` y `- Copiar(2)`. Yo tenía dos, y uno de ellos traía `enable_support = 1` con umbral de voladizo a **5°** — seleccionarlo por error llena la pieza de soportes.

Con el truncado de nombres de la lista, ese duplicado se lee casi igual que el bueno. Bórralos en cuanto los detectes.

Y sobre las actualizaciones de la aplicación

El mismo día, la aplicación se actualizó sola de la versión 7.2.1 a la 7.2.2 y **reescribió 1.896 de 1.957 ficheros de perfil de sistema**, añadiendo seis perfiles nuevos.

Lo comprobé contra una copia de Time Machine anterior: de los 56 perfiles de mi impresora, **53 eran idénticos en valores** y solo cambió el formato del fichero. Los tres que cambiaron de verdad no afectaban a mi material.

Dos cosas que aprendí de esa comprobación:

- **La aplicación sí reescribe los perfiles de sistema** al actualizarse. Que cambie la fecha de todos los ficheros no significa que hayan cambiado los valores.

- **El diff útil es semántico sobre el JSON, no byte a byte.** Un `cmp` marcaba los 56 como distintos solo por el reformato, lo que habría llevado a una conclusión falsa.

Resumen operativo

1. Crea un preset personalizado **para cada material que uses de verdad.**
2. Antes de lanzar una impresión larga, verifica en el gcode que los dos `settings_id` son los tuyos.
3. Borra los duplicados `Copiar(n)` en cuanto aparezcan.
4. Si escribes presets a mano: aplicación cerrada, copia de seguridad, y valores como cadenas.

Cambiar de filamento: lo que cambia sin avisar

Cuando se acaba una bobina y pones otra, el laminador cambia de perfil de filamento. Eso no solo cambia la temperatura: **cambia parámetros de comportamiento que no aparecen en ninguna ficha comercial** y que pueden tumbar una impresión que llevaba semanas saliendo bien.

Esta página documenta el caso concreto que me pasó y, sobre todo, el método para no volver a comerlo.

El caso: Soleyin Ultra PLA ? Hyper PLA

Las dos bobinas son PLA. Misma boquilla, misma máquina, misma pieza. Estos son los perfiles de fábrica de la K2 Pro, resueltos y puestos uno al lado del otro:

parámetro	Soleyin Ultra PLA	Hyper PLA	efecto
slow_down_layer_time	14 s	8 s	las capas cortas dejan de frenarse
filament_max_volumetric_speed	16	23	el perímetro deja de recortarse por caudal
during_print_exhaust_fan_speed	60	0	extracción apagada
filament_flow_ratio	0,97	0,95	—
chamber_temperature	35	30	—
additional_cooling_fan_speed	95	80	—

Y estos son **idénticos** en los dos, lo cual también es información: temperatura de boquilla (220), cama (50), ventilador de pieza (100/100) y close_fan_the_first_x_layers (1).

El parámetro que más pesa:

slow_down_layer_time

Es un mínimo de tiempo por capa. Si una capa se va a imprimir en menos tiempo que ese valor, el laminador la **ralentiza** para dar margen a que el material solidifique antes de que la boquilla vuelva a pasar por encima.

En mi pieza medí capas de **6, 9 y 12 segundos** en la zona alta, donde la sección se reduce a un poste fino. Con el Soleyin esas capas se estiraban a 14 s. Con Hyper PLA pasaban tal cual: sin tiempo de enfriar, sobre una pieza pequeña y con el ventilador a tope.

Bajar de 14 a 8 parece un detalle. En piezas grandes lo es, porque ninguna capa baja de ese umbral. En piezas pequeñas es la diferencia entre imprimir y no imprimir.

Cómo se ve el efecto

Tras subir `slow_down_layer_time` a 14 en mi preset, comparé dos laminados de la misma pieza:

	antes	después
piezas en la placa	4	2
material	4,95 g	3,84 g (−22 %)
tiempo estimado	17m 2s	17m 3s

Mitad de piezas y un 22 % menos de material, **exactamente el mismo tiempo**. Ese es el parámetro trabajando: paga tiempo a cambio de que cada capa solidifique.

El caudal cambia la velocidad real, no solo el máximo

El tope de caudal (`filament_max_volumetric_speed`) recorta automáticamente cualquier velocidad que lo supere. Con capa de 0,2 mm y perímetro exterior de 0,42 mm de ancho:

- Con **16 mm³/s** (Soleyin), un perímetro a 200 mm/s pide 16,8 mm³/s → **se recorta**.
- Con **23 mm³/s** (Hyper PLA), ese mismo perímetro **pasa entero** y va a 200 mm/s reales.

Es decir: el mismo número en la casilla de velocidad produce velocidades distintas según el filamento. Al cambiar de bobina, una pieza pequeña empieza a recibir más fuerza lateral del cabezal sin que hayas tocado nada.

El método: leer los JSON, no las fichas comerciales

Las fichas de fabricante dicen genéricos («optimizado para alta velocidad», «resistente a impactos») y no permiten decidir nada. Los perfiles reales están en disco:

```
~/Library/Application Support/Creality/Creality
Print/<version>/system/Creality/filament/*.json
```

Ojo: heredan. Cada fichero tiene una clave `inherits` que apunta a su padre, y los valores efectivos son la cadena resuelta. Comparar los ficheros en bruto no sirve: hay que resolver la herencia primero.

```
import json, os

def resuelve(nombre, base, prof=0):
    ruta = os.path.join(base, nombre + '.json')
    if not os.path.exists(ruta):
        return {}
    d = json.load(open(ruta))
    padre = {}
    if 'inherits' in d and prof < 8:
        padre = resuelve(d['inherits'], base, prof + 1)
    padre.update(d)
    return padre
```

Con eso, comparar dos materiales es poner las claves que importan en una tabla. Las que miro siempre:

- `slow_down_layer_time`
- `filament_max_volumetric_speed`
- `fan_min_speed`, `fan_max_speed`, `close_fan_the_first_x_layers`
- `during_print_exhaust_fan_speed`
- `activate_chamber_temp_control`, `chamber_temperature`
- las temperaturas de cama según el tipo de placa

Otros hallazgos del mismo método

Comparando perfiles de PETG con esta técnica salieron cosas que ninguna ficha decía:

	CR-PETG	Hyper PETG	Hyper PETG-CF
ventilador de extracción	0 %	60	60
caudal máximo	18	23	12
temperatura de transición	60 °C	75	80

Que CR-PETG sea el único que apaga la extracción explicó, con dato en vez de sospecha, un atasco por acumulación de calor en una impresión larga.

Y en PLA: **CR-Silk admite 10 mm³/s frente a los 23 de Hyper PLA**. El silk imprime a menos de la mitad de velocidad, cosa que conviene saber antes de planificar una serie.

Dos avisos sobre lo que NO te dice el perfil

- `required_nozzle_HRC` **no sirve para detectar abrasivos**. En toda la biblioteca solo 28 perfiles de 1.379 lo rellenan. Que ponga 0 no significa que el material no desgaste: el PLA marble lleva ~30 % de carbonato cálcico y pide boquilla endurecida para producción, con un 0 en esa casilla.
- **El tipo de placa puede ser decorativo**. En el perfil de fábrica de mi máquina, «High Temp Plate» y «Textured PEI Plate» declaran **la misma temperatura** de cama. Elegir bien el tipo de placa no sube la cama; hay que subirla a mano.

Resumen operativo

Cuando cambies de bobina a un material distinto, aunque sea el mismo tipo de plástico:

1. Resuelve y compara los dos perfiles con la tabla de claves de arriba.
2. Mira especialmente `slow_down_layer_time` si vas a imprimir piezas pequeñas.
3. Crea tu preset personalizado para el material nuevo antes de lanzar nada largo.
4. Verifica en el gcode que se aplicó.

La K2 Pro

La maquina: diagnostico, Moonraker, nivelacion y comportamiento real.

La K2 Pro

Interrogar la impresora con Moonraker

La K2 Pro corre **Klipper** por dentro, y expone **Moonraker**, su API HTTP, en el puerto **7125**. Eso significa que se le puede preguntar a la máquina qué ha pasado, en vez de deducirlo de una foto de espaguetis.

Es la herramienta que más tiempo me ha ahorrado diagnosticando fallos.

Antes de nada: seguridad

Moonraker responde **sin autenticación** en la red local. Eso es cómodo dentro de casa y es un problema serio si la impresora queda expuesta a internet: cualquiera podría lanzar trabajos o mover ejes.

- **No abras el puerto 7125 en el router.** Nunca.
- Si necesitas acceso remoto, hazlo por VPN.

Localizar la impresora

```
ping K2Pro-XXXX.local
```

El sufijo son los últimos caracteres del identificador que muestra la propia impresora en su pantalla de red. Si mDNS no resuelve, barre el puerto 7125 en tu rango local.

Comprobación de que responde:

```
curl -s "http://<IP>:7125/printer/info"
```

Devuelve el estado (`ready`, `startup`, `error`), el hostname y las rutas de configuración.

Historial de trabajos

El primer sitio donde mirar tras un fallo:

```
curl -s "http://<IP>:7125/server/history/list?limit=10"
```

De cada trabajo interesan tres campos:

- `status` — `completed`, `cancelled`, `error`, `in_progress`.
- `print_duration` — tiempo **imprimiendo**.
- `end_time` — `start_time` — tiempo de **reloj**.

Que esos dos últimos difieran delata una pausa. En un fallo que investigué, 43 minutos impresos dentro de 5 h 34 min de reloj: la máquina había estado pausada casi cinco horas sin que nadie lo supiera.

Este endpoint también sirve para medir el sesgo del estimador de tiempo. En mi máquina, comparando estimado contra real en decenas de trabajos, sale un factor consistente de **×1,06**: Creality Print siempre se queda corto.

Los ficheros: gcode, logs y configuración

```
curl -s "http://<IP>:7125/server/files/list?root=gcodes"
curl -s "http://<IP>:7125/server/files/list?root=logs"
curl -s "http://<IP>:7125/server/files/roots"
```

Descargar el gcode que la máquina ejecutó **es la única forma fiable** de saber con qué ajustes se laminó de verdad:

```
curl -s "http://<IP>:7125/server/files/gcodes/<fichero>.gcode" \
| grep -aE "^;(print|filament)_settings_id|^;(hot_plate_temp|slow_down_layer_time)"
```

Los `klippy.log` diarios pueden pasar de **100 MB**. Descarga solo el tramo final con HTTP Range:

```
curl -s -r "-900000" -o cola.log "http://<IP>:7125/server/files/logs/klippy.log"
```

La línea que da la capa exacta del fallo

Klipper escribe esto **en cada capa**, para poder recuperar tras un corte de luz:

```
record_power_loss_info to eeprom layer:33 last_position[2]:6.4
```

Da **número de capa y altura Z exactos**. Es el dato de oro de cualquier diagnóstico, por dos motivos:

- **Cruzando esa Z con la altura de cada pieza de la placa** sabes qué estaba imprimiéndose y qué había terminado. En un fallo mío, la impresión murió justo en la capa donde acababan los tapones más bajos, con las bases al 48 % — eso señalaba directamente al problema de mezclar alturas dispares en una misma cama.
- **La cadencia entre esas líneas es la señal de salud**. Si el intervalo se mantiene constante hasta el último segundo, se descarta una degradación progresiva; si se va alargando, hay algo que se está atascando.

La rueda de medición: extrusión real

Si tienes CFS, registra el **avance real del filamento**:

```
[get_measuring_wheel] data_hex: 0xc5bf2700, data:-6116.875
```

Su derivada delata un atasco **antes de que salte ningún error**. En un heat creep que sufrí, la rueda se paró en seco y la máquina no reportó nada hasta cuatro minutos después. Quien mira solo los códigos de error diagnostica el síntoma; la rueda da la causa y el minuto exacto.

Distingue además dos fallos que se confunden:

- **Rueda parada con el estado en `printing`** → atasco o falta de material.
- **Rueda avanzando con normalidad y aun así espaguetis** → es adherencia, la pieza se ha soltado. El filamento salía bien.

Estado del CFS y del material

```
curl -s "http://<IP>:7125/printer/objects/query?box"
```

Da por hueco: material declarado, color, longitud restante, temperatura y **humedad**. Útil para no secar una bobina que no lo necesita — en un caso marcaba 18 % de humedad relativa y me ahorró ocho horas de secado inútil.

Aviso: el nombre del fichero `.gcode` puede mentir sobre el material. He visto un `_PLA_` en el nombre de un trabajo hecho en PETG. Este endpoint dice lo que hay **realmente** cargado.

Otros objetos útiles

```
curl -s
```

```
"http://<IP>:7125/printer/objects/query?print_stats&virtual_sdcard&extruder&heater_bed&bed_mes  
h"
```

- `print_stats` — estado, fichero, duración, mensaje de error.
- `virtual_sdcard` — progreso real (0 a 1).
- `extruder` / `heater_bed` — temperatura actual y objetivo. Sirve para verificar que tu perfil se aplicó: si pediste cama a 60 y aquí ves objetivo 50, no se aplicó.
- `bed_mesh` — malla activa, perfiles guardados y matriz de puntos palpados.

Filtrar el ruido

El `klippy.log` está inundado de líneas que no aportan nada al diagnóstico. Filtra siempre:

- `buf_len`, `buf[0] = 0x...` — volcados del bus RS-485. Aparecen marcados como `ERROR` y **no son errores de impresión**. Me costó un rato entenderlo la primera vez.
- `Serial_485` — tráfico del mismo bus.
- `GET_BOX_STATE`, `cur_temp` — sondeos periódicos.

```
grep -a "ERROR" cola.log | grep -avE "buf\[|buf_len|Serial_485"
```

Una advertencia sobre la detección por IA

En mi máquina, la detección por IA del cabezal solo corre **seis segundos al arrancar** (`execute_toolhead_ai_waste_management`: enciende la cámara, analiza, la apaga). La detección continua durante la impresión la hace un subsistema propietario que **no deja una sola línea en el log**.

No la busques ahí. Si esperas encontrar por qué la máquina paró (o por qué no paró), ese registro no existe.

Resumen: la secuencia de diagnóstico

1. `/server/history/list` — estado, duración impresa vs reloj.
2. **Descargar el gcode** — con qué ajustes se laminó realmente.
3. **Cola del `klippy.log`** con Range — buscar `record_power_loss_info` para la capa exacta.
4. **Cruzar esa Z** con las alturas de las piezas de la placa.
5. **Cadencia entre capas** — ¿degradación progresiva o corte abrupto?

6. **Rueda de medición** — ¿extrusión o adherencia?
7. **Solo entonces**, opinar sobre la causa.

Conceptos basicos

Vocabulario y fundamentos para quien empieza en impresion 3D.

Que es el brim

Brim significa «ala» en inglés, como el ala de un sombrero. Es un anillo plano que el laminador dibuja alrededor de la pieza, pegado a ella y de **una sola capa de alto**. No forma parte del objeto: cuando la impresión termina, se retira con la uña o un cúter.

Sirve para una cosa: **que la pieza no se despegue de la cama**.

Por qué hace falta

Una pieza se sujeta a la cama solo por la superficie que la toca. Si esa superficie es pequeña, cualquier fuerza lateral —el cabezal pasando a 150 mm/s, el ventilador soplando— puede arrancarla. Y una pieza suelta rodando por la cama mientras la boquilla sigue extruyendo produce lo que se conoce como «espaguetis».

Con números reales de una pieza mía, un tapón de 13,5 mm de diámetro:

	superficie pegada a la cama
sin brim	105 mm²
con brim de 5 mm	396 mm²

Casi cuatro veces más agarre, sin cambiar la pieza ni un milímetro.

Qué es el «brim» — el ala que sujeta las piezas pequeñas a la cama

SIN brim — como salió hoy



CON brim 5 mm — el preset nuevo



visto de lado

tapón 5,75 mm de alto



Para comparar: las bases del mismo proyecto apoyan **585 mm² ellas solas** y no necesitan brim ninguno.

Los modos

- `no_brim` — sin brim, nunca.
- `outer_only` — brim alrededor de todas las piezas, siempre. Es el que hay que usar cuando quieres asegurarte.
- `inner_only` / `outer_and_inner` — también dentro de los agujeros de la pieza.
- `auto_brim` — el laminador **decide él**, pieza por pieza, si le hace falta.

La trampa de `auto_brim`

`auto_brim` suena a la opción sensata: ponerlo solo donde hace falta, sin ensuciar el resto. Yo lo tenía puesto y lo recomendaba.

El problema es que su criterio no tiene por qué coincidir con el tuyo. Miró unos tapones de 105 mm² de huella y decidió que no les hacía falta brim. Eran justo las piezas que se despegaban.

Y hay un detalle que cuesta ver: **subir `brim_width` no arregla eso.** El ancho solo actúa *cuando el laminador ya ha decidido poner brim*. Si decide que no, un ancho mayor sigue siendo cero.

O sea: con piezas de huella pequeña, si quieres brim hay que **forzarlo** con `outer_only`. Confiar en el automático es confiar en un criterio que no controlas.

Cuidado con lo que apoya en la cama

Un aviso relacionado, porque es el mismo problema visto al revés: **cualquier resalte en la cara que apoya reduce la huella a ese resalte.**

Si pones un texto en relieve, un logotipo o un pie en la cara que va contra la cama, lo único que toca el cristal son esas letras. En una pieza mía, un nombre en relieve de 0,6 mm dejó la huella en **222 mm²** con 7.491 mm² de pieza colgando en el aire. Embebiendo ese texto a ras del espesor, la huella subió a **6.536 mm²** — casi treinta veces más.

Antes de dar por buena una pieza con texto o resalte en la base, mide la huella real. Si es mucho menor que el área de esa cara, el resalte sobra.

La alternativa: pegamento en spray

El brim no es la única solución, y no siempre es la mejor. Para piezas donde la cara de la cama no se ve ni necesita precisión, un poco de laca o pegamento en spray sobre la placa hace el mismo trabajo con ventajas:

- **No hay que retirar nada.** Un brim de 5 mm alrededor de una pieza pequeña son unos 290 mm² de ala que hay que despegar con cuidado.
- **No toca la geometría.** El brim se une a la pieza por su primera capa y siempre deja algo de rebaba en el perímetro. En una pieza de 13 mm de diámetro, el riesgo de marcar el borde al arrancarla no es despreciable.
- **Lo aplicas cuando toca**, en vez de que un ajuste lo decida en todos tus proyectos.

Yo uso brim cuando la pieza es delicada o el borde importa, y spray cuando son piezas funcionales cuya base va contra una pared.

Resumen

- El brim es un ala de una capa que amplía la superficie de agarre. Se retira al terminar.
- **Piezas con menos de ~150 mm² de huella son candidatas claras.**
- `auto_brim` puede decidir que no hace falta cuando sí hace falta. Si lo necesitas de verdad, fuérzalo con `outer_only`.
- Subir el ancho no cambia la decisión de *si* ponerlo.
- El pegamento en spray es alternativa legítima, sobre todo si la cara de la cama no se ve.