

Demostración de Capacidades de Texforge

Jheison M

13 de agosto de 2026

Resumen

Este documento demuestra las capacidades principales de **texforge**, un compilador LaTeX que utiliza tectonic como motor interno, eliminando la necesidad de instalar TeX Live o MiKTeX. Se presentan diagramas embebidos (Mermaid, Graphviz y D2), citas bibliográficas, referencias cruzadas, matemáticas, listados de código, tablas y más.

Índice

1. Introducción	2
1.1. Antecedentes	2
1.2. Objetivos	2
2. Diagramas Embebidos	2
2.1. Mermaid	2
2.2. Graphviz / DOT	2
2.3. D2	2
2.4. Estilos de Diagrama (style)	5
3. Matemáticas	7
3.1. Ecuaciones en línea	7
3.2. Ecuaciones numeradas	7
3.3. Matrices	7
4. Citas Bibliográficas	7
5. Listados de Código	8
5.1. Python	8
5.2. LaTeX	8
6. Tablas	9
7. Comandos de Texforge	9
8. Conclusión	9

1. Introducción

Texforge simplifica el flujo de trabajo académico al compilar documentos LaTeX directamente a PDF sin distribuciones externas [2]. En esta sección se describen los antecedentes y motivaciones del proyecto.

1.1. Antecedentes

La edición de documentos académicos en $\text{\LaTeX}$ requiere tradicionalmente instalaciones pesadas como TeX Live o MiKTeX. Texforge elimina esa barrera [3], ofreciendo una experiencia de línea de comandos ligera y moderna.

1.2. Objetivos

Los objetivos de este documento son:

- (I) Demostrar diagramas embebidos en tres formatos distintos.
- (II) Mostrar la integración bibliográfica con BibTeX.
- (III) Ejemplificar referencias cruzadas entre secciones.
- (IV) Presentar ecuaciones matemáticas complejas.
- (V) Incluir listados de código fuente.

2. Diagramas Embebidos

Texforge renderiza diagramas directamente desde el código fuente LaTeX sin herramientas externas como Inkscape o Node.js.

2.1. Mermaid

El siguiente diagrama de secuencia muestra la interacción entre los componentes de Texforge:

2.2. Graphviz / DOT

También se pueden crear diagramas directed-Graph con Graphviz:

2.3. D2

El diagrama D2 es otro formato soportado nativamente:

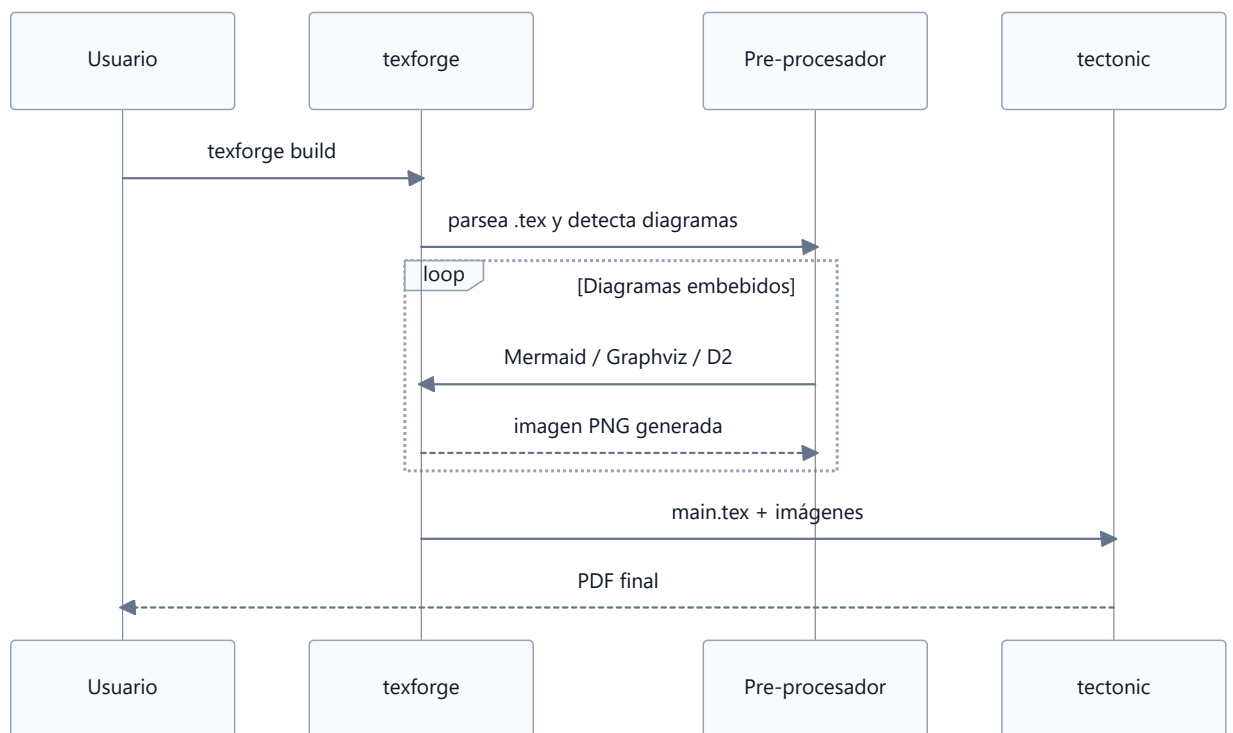


Figura 1: Secuencia de compilación de Texforge

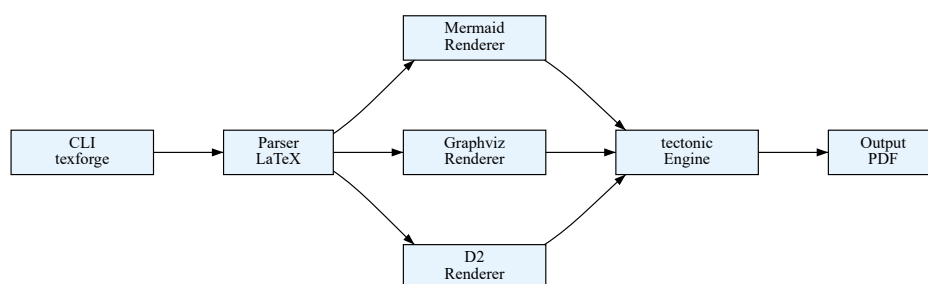


Figura 2: Arquitectura de Texforge

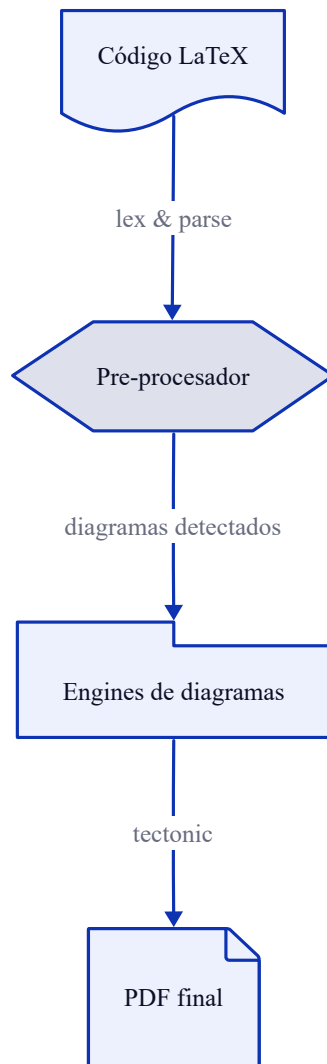


Figura 3: Pipeline de renderizado

2.4. Estilos de Diagrama (style)

Los tres entornos aceptan también un atributo **style**, combinable con **width**, **pos** y **caption**. Un diagrama de tres cajas apenas ejercita un relleno y un color de borde, así que en vez de eso el mismo diagrama de secuencia — con participantes, una nota, un bloque **alt** y una barra de activación — se repite a continuación una vez por preset, para que la diferencia se vea en mucha más superficie:

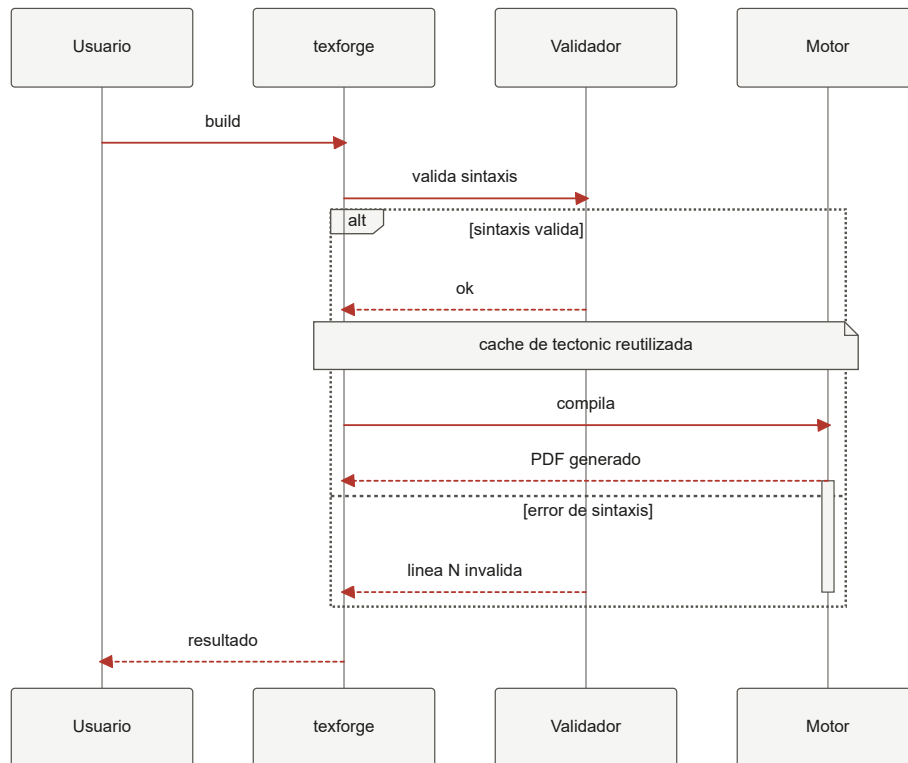


Figura 4: Preset **editorial**: participantes y nota en tono papel con tinta oscura; el acento vermellón vive solo en las flechas de mensaje, nunca en el relleno de una caja

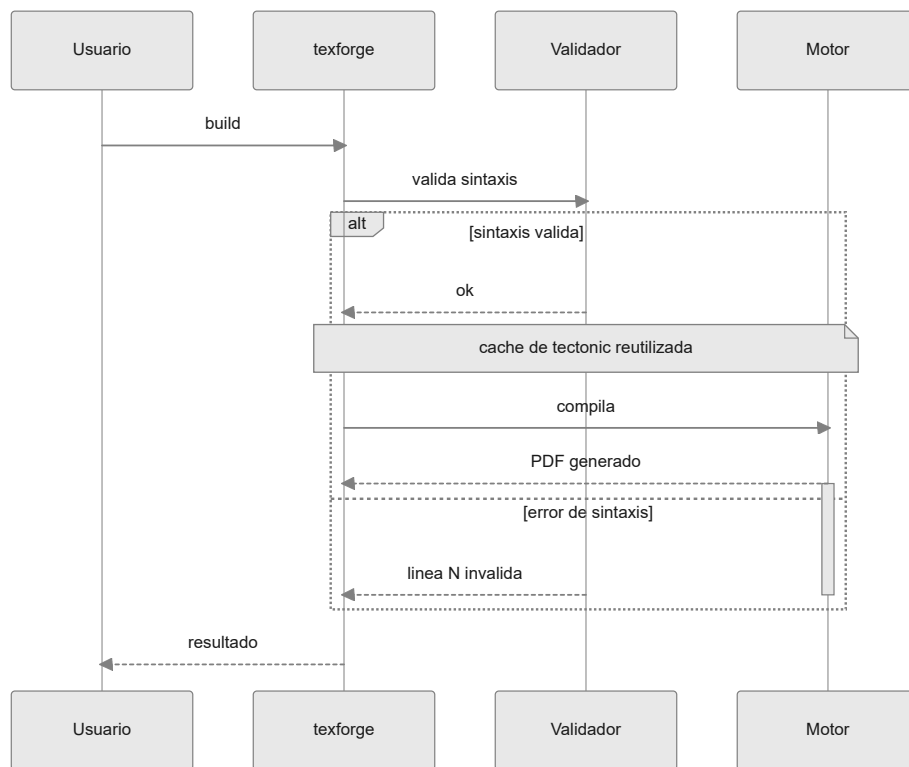


Figura 5: Preset **monochrome**: el mismo diagrama sin un solo color, apto para impresión en blanco y negro

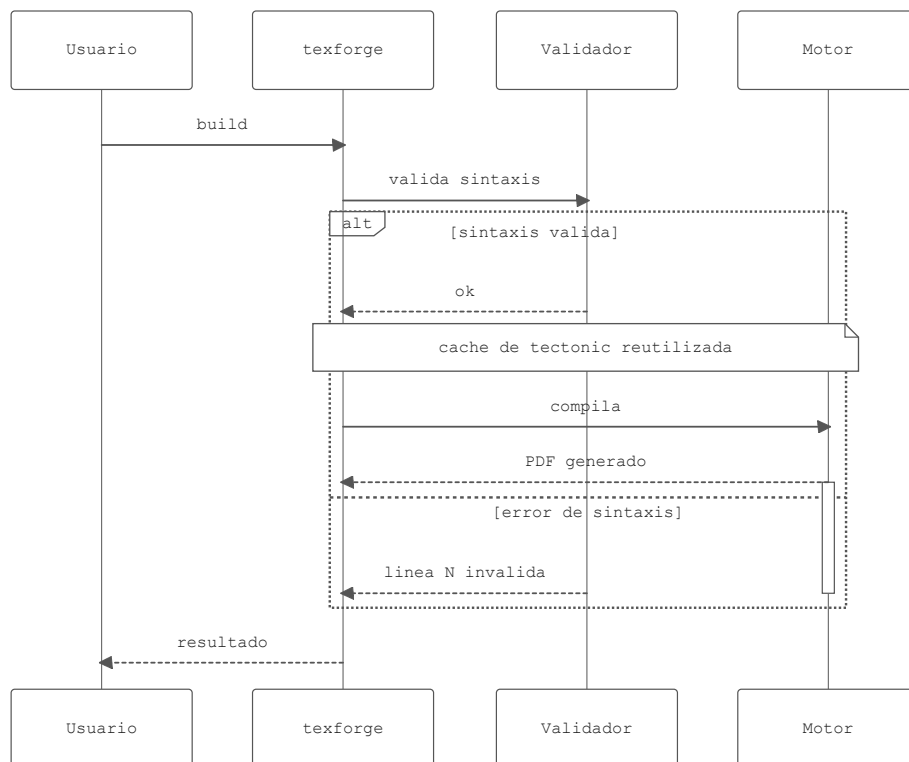


Figura 6: Preset **technical**: cajas sin relleno, trazo uniforme y etiquetas monoespaciadas, incluida la etiqueta del bloque **alt**

El preset `editorial` sirve para documentos que se leen en pantalla o se imprimen a color y buscan un acabado sobrio: las cajas de los participantes, la nota y la barra de activación comparten un neutro cálido, y el acento vermellón se reserva para las flechas de mensaje, sin gastarse nunca en un relleno. El preset `technical` conviene a diagramas que funcionan como planos o esquemas de referencia: sin relleno, con peso de trazo uniforme y etiquetas monoespaciadas hasta en el rótulo del bloque `alt`. El preset `monochrome` tiene una razón menos obvia: existe para documentos que se imprimen en blanco y negro, donde un diagrama a color puede perder significado, porque dos colores que se distinguen en pantalla pueden reducirse al mismo gris al imprimirse sin color; `monochrome` evita ese problema por construcción, ya que nunca depende del color para distinguir nada — ni siquiera una llamada de una respuesta. Omitir `style` equivale a `style=default`, el aspecto propio de cada motor, como en los diagramas de las secciones anteriores.

También puede fijarse un preset por defecto para todo el documento en `project.toml`, bajo `[diagrams]` con `style = "..."`; el atributo `style` de un entorno concreto, cuando está presente, sobrescribe ese valor por defecto.

3. Matemáticas

3.1. Ecuaciones en línea

La fórmula cuadrática se expresa como $x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$.

3.2. Ecuaciones numeradas

$$\int_{-\infty}^{\infty} e^{-x^2} dx = \sqrt{\pi} \quad (1)$$

$$e^{i\pi} + 1 = 0 \quad (2)$$

Como se demostró en la Ecuación 1, la integral de Gauss tiene un valor exacto conocido.

3.3. Matrices

$$\mathbf{A} = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{pmatrix} \quad (3)$$

4. Citas Bibliográficas

Texforge gestiona automáticamente las referencias BibTeX. A continuación se presentan tres tipos de cita:

- Artículos de revista [2]
- Libros [3]

-
- Recursos en línea [1]

La sección 3 muestra el uso de referencias cruzadas, mientras que la sección 2 describe los formatos de diagramas disponibles.

5. Listados de Código

5.1. Python

Listing 1: Ejemplo de función en Python

```
import numpy as np

def integrar_gauss(n_puntos=1000):
    """Aproxima la integral de Gauss por cuadratura."""
    x = np.linspace(-10, 10, n_puntos)
    dx = x[1] - x[0]
    y = np.exp(-x**2)
    return np.sum(y) * dx

if __name__ == "__main__":
    resultado = integrar_gauss()
    print(f"Integral_{\_\_}{resultado:.6f}")
```

5.2. LaTeX

Listing 2: Estructura básica de Texforge

```
\documentclass{article}
\usepackage{graphicx}

\begin{document}
\maketitle

\begin{figure}
  \centering
  \includegraphics[width=\linewidth]{diagrams/mermaid-e8dd648c1cd
    0cdb9.pdf}
\end{figure}

\bibliographystyle{plain}
\bibliography{refs}
\end{document}
```

6. Tablas

Cuadro 1: Comparación de motores de renderizado de diagramas

Formato	Motor	Externo?	Velocidad
Mermaid	Rust puro	No	Rápido
Graphviz	layout-rs	No	Rápido
D2	d2-little	No	Rápido

Como se muestra en la Tabla 1, todos los motores son embebidos y no requieren binarios externos.

7. Comandos de Texforge

Cuadro 2: Resumen de comandos principales

Comando	Descripción
<code>texforge new</code>	Crea un proyecto nuevo con un template.
<code>texforge init</code>	Wizard interactivo para proyectos existentes.
<code>texforge build</code>	Compila <code>main.tex</code> a PDF en <code>build/</code> .
<code>texforge check</code>	Lintor estático: valida archivos, citas y labels.
<code>texforge fmt</code>	Formatea archivos <code>.tex</code> .
<code>texforge clean</code>	Elimina el directorio <code>build/</code> .
<code>texforge template</code>	Gestiona templates instalados o remotos.

8. Conclusión

Texforge demuestra ser una herramienta completa para la edición de documentos académicos en $\text{\LaTeX}$. Sus ventajas principales son:

- **Sin dependencias externas:** no requiere TeX Live ni MiKTeX.
- **Diagramas embebidos:** Mermaid, Graphviz y D2 renderizados en Rust puro.
- **Lintor integrado:** detección temprana de errores con `texforge check`.
- **Templates:** soporte para APA, IEEE, cartas y documentos generales.

Como se demostró en este documento, la combinación de diagramas (Sección 2), ecuaciones (Sección 3), citas (Sección 4) y código (Sección 5) permite crear publicaciones académicas de alta calidad con un mínimo esfuerzo.

Referencias

- [1] Blog Author. Título del artículo de blog. <https://www.example.com/blog/article>, 2023. Accessed: 2023-01-01.
- [2] John Doe. Título del artículo de ejemplo. *Nombre de la Revista*, 1(1):1–10, 2023.
- [3] Jane Smith. *Título del Libro de Ejemplo*. Editorial de Ejemplo, 2022.