

# Única Fuente de Verdad: Resolviendo la Deriva de Slugs entre SSG y CMS

Cuadernillo | Vol. 1

mercedev.es — 2026-05-01 | Fase 8 (Expansión de Contenido)

## El Desafío (Síntoma)

---

En nuestra arquitectura híbrida, el orquestador maestro generaba automáticamente versiones en PDF de los artículos dinámicos para que los usuarios pudieran descargarlos. Sin embargo, nuestro rastreador de enlaces ( `merci-linkcheck.py` ) detectó masivos errores 404 (Not Found). Python estaba generando PDFs con nombres matemáticamente perfectos (ej. `mi-post.pdf` ), pero WordPress, al detectar títulos similares en su papelera, alteraba silenciosamente los enlaces en producción (ej. `mi-post-2` ). Al diferir los motores de nomenclatura, los botones de descarga de la web apuntaban a archivos que no existían.

## La Maniobra (Lógica)

---

En ingeniería de software, cuando integras dos sistemas, solo uno puede ostentar la "Única Fuente de Verdad" (Single Source of Truth - SSOT).

En lugar de intentar programar en Python el complejo algoritmo de duplicidad de WordPress, invertimos el orden lógico de ejecución en nuestro script publicador Headless ( `merci-wp.py` ). En vez de *Fabricar PDF -> Publicar en WP*, pasamos a *Publicar en WP -> Leer respuesta de la API -> Extraer el Slug definitivo -> Fabricar PDF*.

## El Aprendizaje / Deuda Técnica

---

Esta maniobra es un ejemplo de manual de **Tolerancia a Fallos y SSOT**. En un ecosistema acoplado, el sistema que persiste los datos (la base de datos de WordPress) es el que manda sobre las URIs.

Al retrasar la generación estática hasta recibir la confirmación del servidor dinámico, aseguramos una **Paridad Dev/Prod absoluta**. La automatización no

consiste en que un script lo haga todo rápido y a ciegas, sino en que sepa preguntar a las herramientas adecuadas antes de actuar.