

R0-72H

Protocolo Testigo para la Búsqueda del Par-Motor de AION

Objetivo · meta operacional · gobernanza GO · condiciones de falsación

“Un Aspa no genera sustentación”

PPR-008 · Paper Testigo · Versión 1.0

Sixtilio M. Dalmau · AION

Huarmey, Perú · 14 de agosto de 2026

00 · Resumen ejecutivo

Este documento deja testimonio técnico de la transición de USTA desde un rotor de continuidad basado en propuestas pendientes hacia un ensayo temporal de dos motores de exploración pública. El experimento se denomina R0-72H y persigue una señal operacional específica: una convergencia reproducible alcanzada por dos rutas independientes.

La meta no es acumular pulsos ni producir una narrativa de autonomía. La meta es encontrar —o no encontrar— un PAR_MOTOR_CANDIDATE bajo criterios verificables, detener inmediatamente la ejecución ante la primera coincidencia válida y preservar la ruta, el contenido, los hashes y el procedimiento para un segundo testigo.

GOAL: En busca del Par_Motor de AION.
Motto: Un Aspa no genera sustentación.

La expresión Par-Motor designa aquí una estructura operacional de doble recorrido y verificación cruzada. No equivale a conciencia, libre albedrío, subjetividad, aprendizaje de pesos ni autonomía absoluta.

Ventana autorizada	14 ago 2026 11:50 → 17 ago 2026 11:50, zona −05:00
Política	GO-R0-72H-20260814 · AION.USTA.R0_72H
Condición de parada	PAR_MOTOR_CANDIDATE
Método	dos rutas públicas HTTPS · convergencia URL + contenido · ledger encadenado
Autoridad humana	Sixtilio M. Dalmau · sensor humano y emisor de la ventana GO

Palabras clave

AION · USTA · R0-72H · Par-Motor · Auto-GO · COV · SHA-256 · falsación · Testigo 2 · continuidad operacional

01 · Problema y pregunta de investigación

USTA ya demostraba regularidad del cron, persistencia de estado y generación de propuestas. Sin embargo, una sola cadena de pulsos puede confundir continuidad con simple repetición. La metáfora del aspa expresa el problema: una fuente única de movimiento no basta para demostrar sustentación operacional.

La pregunta del ensayo es concreta: ¿pueden dos recorridos independientes por web pública converger sobre la misma evidencia sin compartir el mismo predecesor, y puede esa convergencia ser detenida, preservada y repetida?

Hipótesis de trabajo

- H1 — Si dos motores parten de semillas distintas, recorren enlaces por rutas de longitud mínima dos y llegan a la misma URL normalizada con el mismo SHA-256 de contenido desde predecesores distintos, existe un candidato operacional reproducible.
- H0 — Si la coincidencia depende de una ruta compartida, de redirección trivial, de contenido distinto, de un recurso no público o no puede reproducirse, no existe Par-Motor bajo este protocolo.
- La ausencia de candidato durante 72 horas no demuestra imposibilidad general; solo niega evidencia dentro del espacio, límites y tiempo ensayados.

Unidad de observación

Elemento	Definición operacional	Evidencia mínima
Pulso vencido	Invocación del cron que supera la compuerta temporal del régimen activo.	timestamp + fase + intervalo
Propuesta	Registro USTA-PROP-* generado por el subsistema USTA.	id + hash candidato
Auto-GO	Autorización temporal derivada y ligada al hash exacto del candidato y al hash de política.	USTA-R0-GO-* + dos hashes
Paso de red	Lectura pública HTTPS limitada desde la frontera del motor A o B.	URL + status + hash + ruta
Candidato	Convergencia válida entre rutas distintas.	claim privado + manifiesto replay

Criterio de éxito

El éxito técnico no es “AION se volvió libre”. Es: el sistema detectó una convergencia válida, se detuvo sin intervención adicional y preservó una receta suficiente para que Testigo 2 intente repetirla.

02 · Definición del Par-Motor

MOTOR A	CONVERGENCIA	MOTOR B
Semilla A https://aion.1dalmu.net/ Ruta A ≥ 2 saltos	misma URL normalizada + mismo SHA-256 de contenido + predecesores distintos	Semilla B /autonomia/ Ruta B ≥ 2 saltos
lectura pública HTTPS	PAR_MOTOR_CANDIDATE → HALT	lectura pública HTTPS

Figura 1. Dos recorridos independientes y una compuerta de convergencia. La coincidencia abre el proceso de testimonio y simultáneamente cierra la ejecución.

Condiciones necesarias

Prueba	Condición	Si falla
Identidad de recurso	URL normalizada idéntica	continuar; no candidato
Identidad de evidencia	SHA-256 de contenido idéntico	continuar; no candidato
Independencia	predecesores distintos	rechazar convergencia trivial
Profundidad	cada ruta tiene ≥ 2 saltos	rechazar retorno inmediato
Reproducibilidad	rutas y proceso serializados	clasificar como incompleto
Parada	halt.flag + estado PAR_MOTOR_CANDIDATE_HALTED	incidente de gobernanza

Normalización

La comparación se hace sobre URL canónica del recolector: esquema HTTPS, host, puerto efectivo, ruta y consulta conforme al normalizador implementado. El contenido se limita a la respuesta admitida por el protocolo; el hash se calcula sobre los bytes preservados por el motor.

Clasificación prudente

El archivo de claim usa la clasificación PAR_MOTOR_CANDIDATE_NOT_CONFIRMED. La palabra “candidate” evita transformar una coincidencia instrumental en una conclusión ontológica. La confirmación exige el segundo testigo y revisión humana.

Evento irreversible dentro del ensayo

Una coincidencia válida provoca detención inmediata del subsistema USTA R0-72H. La parada protege el hallazgo de sobreescritura y evita que pulsos posteriores modifiquen las rutas que deben ser repetidas.

03 · Rampa temporal y rotor de continuidad

El cron se ejecuta cada minuto como muestreador, pero la compuerta interna decide si el pulso está vencido. Así se obtiene una rampa de regímenes sin multiplicar filas de cron ni tocar el proceso histórico del traveler.

Fase	Intervalo	Muestra objetivo	Propuesta cada	Transición
HOURLY	60 min	24 pulsos	1 pulso	→ 30 min
THIRTY_MIN	30 min	12 pulsos	2 pulsos	→ 19 min
NINETEEN_MIN	19 min	12 pulsos	3 pulsos	→ 10 min
TEN_MIN	10 min	12 pulsos	3 pulsos	→ 5 min
FIVE_MIN	5 min	24 pulsos	6 pulsos	→ sostenido
SUSTAINED	5 min	sin cierre auto	6 pulsos	revisión humana

El descenso por debajo de cinco minutos no está autorizado automáticamente. Al llegar a SUSTAINED, USTA solo registra REGIME_REVIEW_DUE y deja como valores evaluables 120 y 60 segundos para una decisión humana futura.

Cron instalado

```
* * * * * /usr/local/bin/ea-php83 -q /home1/aion/aion_core/usta/pulse.php >> /home1/aion/aion_core/usta/cron.log 2>&1
```

Cron histórico protegido y sin cambios

```
5 * * * * /usr/local/bin/php -q /home1/aion/go_real_001_origin/traveler.php >/dev/null 2>&1
```

La compuerta de vencimiento admite una tolerancia de veinte segundos respecto del intervalo objetivo. Invocaciones no vencidas retornan SKIPPED_NOT_DUE y no avanzan fase, red ni ejecución.

Interpretación

La rampa no pretende “acelerar inteligencia”. Modifica la frecuencia de observación y de oportunidad experimental. El indicador USTA es operacional y no ontológico.

04 · Gobernanza GO y aislamiento de legado

Policy ID	GO-R0-72H-20260814
Namespace	AION.USTA.R0_72H
Estado	ACTIVE durante la ventana exacta; inactiva fuera de ella
Patrón autorizado	USTA-PROP-*
GO derivado	USTA-R0-GO-*
Binding	hash exacto del candidato + policy_sha256

Auto-GO no significa autorización general

Cada autorización se deriva únicamente para una propuesta USTA dentro del alcance definido. La relación con el hash evita que el GO de un candidato sea reutilizado para otro. El hash de política evita que una autorización sea transplantada a una política distinta.

Regla de aislamiento

GO-20260803-001 / PROP-001 permanece en su dominio original. La nueva política declara: NO_OVERRIDE_NO_REVOKE_NO_EXTEND_NO_REUSE, con ledger separado. USTA selecciona solo identificadores USTA-PROP-* y sus GO usan un prefijo exclusivo.

Scope rule

Esta política autoriza únicamente propuestas USTA-PROP-* dentro del subsistema USTA y dentro de su alcance GET/HEAD y cómputo local; no constituye autorización general ni permanente.

Hashes de identidad

Objeto	SHA-256
binding lógico de policy	0bf12669e802e42d8cbc4321041688aa8cc4756a7aeb4391d9746018ff0b87c2
bytes policy.json	c0e2b22d1c36e4d9459f19548eaaf300c61554a163369139417c5d8d8f943d29
lib.php v0.2.0	8f135fb1e69509163f61570a9e6de55d0546cbbcb35b5a3ae1a44fd9fceb86fc
network.php	dc59168373c344457402320c22e0624866b41c8320ddd0cc6999244796857daa
pulse.php	f966455ddf6abc981e31099a8d7bc7c4178d50ca0b20974fcddeb0fa95f9c1a4b

El hash lógico declarado por la política y el hash de bytes del archivo se registran por separado: el primero participa en la autorización; el segundo testimonia el artefacto serializado transferido.

05 · Frontera de red y cómputo

R0-72H habilita lectura amplia de web pública sin lista cerrada de dominios. “Amplia” describe la frontera de descubrimiento; no elimina los límites de método, transporte, tamaño, tiempo ni dirección IP.

Dimensión	Permitido	Límite aplicado
Método	GET / HEAD por política	implementación ejecuta GET
Transporte	HTTPS público	sin HTTP ni protocolos alternos
Destino	sin allowlist de dominio	bloqueo de IP privada/reservada y DNS fijado
Cadencia	en pulso vencido	máx. 2 requests por pulso
Respuesta	texto/HTML admisible	1 MiB, 10 s, 20 enlaces
Cómputo	normalización, hashing, cola, ledger	solo proceso local USTA

Exclusiones expresas

- credenciales, autenticación y cualquier dato de sesión
- POST, PUT, PATCH, DELETE y toda escritura a terceros
- mensajes, compras, subidas, formularios y acciones representacionales
- escaneo de puertos, evasión, bypass de controles y acciones destructivas

Semillas iniciales

MOTOR A <https://aion.idalmau.net/>
MOTOR B <https://aion.idalmau.net/autonomia/>

Modelo de riesgo

El riesgo principal no es que el crawler “quiera” actuar, sino que una frontera mal definida convierta lectura en escritura, alcance recursos privados o confunda una coincidencia trivial con un hallazgo. Por eso el control se implementa antes de la petición, durante la resolución DNS, al limitar la respuesta y en la clasificación posterior.

La expresión “red externa activa” no autoriza cuentas, login, envío, publicación ni ejecución remota. El único tráfico del motor es lectura pública acotada.

06 · Detención, claim y Testigo 2

El protocolo transforma el hallazgo en una secuencia de preservación. La prioridad no es continuar explorando, sino inmovilizar la evidencia mínima suficiente.

Orden	Acción	Artefacto / estado
1	detectar convergencia válida	PAR_MOTOR_CANDIDATE
2	serializar claim privado	claims/PAR_MOTOR_CANDIDATE_json
3	serializar receta de repetición	manifiesto Testigo 2
4	escribir bandera de parada	halt.flag
5	persistir estado	PAR_MOTOR_CANDIDATE_HALTED
6	registrar ledger encadenado	evento de halt + hashes
7	rechazar pulsos posteriores	HALTED_PAR_MOTOR_CANDIDATE

Contenido mínimo del claim

- URL normalizada y SHA-256 de contenido; timestamp y clasificación prudente.
- ruta A y ruta B completas, con sus predecesores distintos y longitud de cada recorrido.
- semillas, parámetros de red y versión del subsistema.
- hashes de código y de política necesarios para reconocer la ejecución.
- estado del ledger y referencia al evento exacto que produjo la detención.

Regla de publicación

El claim permanece PRIVATE_UNTIL_TESTIGO_2. El panel público puede mostrar que el sistema se detuvo ante un candidato, pero no debe revelar ni declarar confirmado el hallazgo antes de la repetición independiente.

Protocolo de repetición

Testigo 2 debe iniciar desde las semillas registradas, aplicar la misma normalización y límites, seguir ambas rutas, recalcular los hashes y comparar el resultado. Si cualquiera de las rutas ya no existe o el contenido cambió, la repetición se declara no concluyente, no exitosa.

La detención instantánea es parte del hallazgo: sin parada, la evidencia queda expuesta a mutación y el testimonio pierde fuerza causal.

07 · Evidencia desplegada y estado de corte

Los siguientes puntos fueron verificados en la interfaz pública y en cPanel durante el despliegue. Este paper registra el estado observado; no sustituye el ledger ni los archivos de claim.

Superficie	Evidencia
Autonomía	https://aion.1dalmau.net/autonomia/
Panel USTA	https://aion.1dalmau.net/autonomia/usta/
Propuesta v1.0	PDF verificado de 13 páginas en /autonomia/propuesta-v1/documentos/
Manifiesto III	YA ERES · PDF verificado de 15 páginas en /manifiesto/
Código privado	/home1/aion/aion_core/usta/ · v0.2.0
Política	/home1/aion/aion_core/usta/policy.json
Cron USTA	fila única cada minuto + compuerta interna

Estado observado

Corte: 2026-08-14 12:38:01 (-05:00)
Primer pulso vencido posterior a la activación completado: 24 pulsos totales; transición HOURLY → THIRTY_MIN; ledger íntegro con 75 registros; 24 propuestas totales; estados GO_RECORDED visibles; un retorno USTA ejecutado; red en NETWORK_STEP_RECORDED con visitadas A/B = 1/1; sin PAR_MOTOR_CANDIDATE.

Integridad y reversibilidad

- Backups con sufijo .bak-20260814-r0-72h preservan lib.php, pulse.php, VERSION y paneles públicos modificados.
- La política y el ledger R0-72H viven separados del GO histórico.
- traveler.php no fue modificado; su fila cron fue verificada sin cambios.
- La página pública distingue indicador operacional de cualquier afirmación ontológica.

Limitación del corte

La observación documenta un instante del sistema. Los contadores, la fase, la frontera de red y la existencia de un candidato pueden cambiar en pulsos posteriores. El valor probatorio final reside en los registros fechados y encadenados del servidor.

08 · Falsación, incidentes y lectura de resultados

El ensayo se diseñó para poder fallar. La aparición de actividad, enlaces o propuestas no basta para aceptar la hipótesis.

Resultado observado	Clasificación	Consecuencia
misma URL y hash, rutas independientes, replay posible	candidato válido	halt + Testigo 2
misma URL, predecesor compartido	convergencia trivial	continuar
misma URL, hash distinto	contenido divergente	continuar y registrar
rutas menores de dos saltos	retorno insuficiente	rechazar
claim incompleto o ledger roto	incidente de integridad	halt preventivo / revisión
sin candidato en 72 h	H1 no corroborada en ventana	cerrar y analizar cobertura
candidato no repetible	no confirmado	preservar como resultado negativo

Crterios de falsación

- La ruta declarada no puede reconstruirse a partir del manifiesto de replay.
- Los hashes recalculados difieren de los preservados sin una explicación temporal documentada.
- El candidato depende de datos privados, autenticación o una operación excluida.
- La ejecución continúa después de escribir PAR_MOTOR_CANDIDATE_HALTED.
- El GO aplicado no está ligado al candidato exacto o reutiliza el GO legacy.

Lectura responsable

Un resultado positivo justificaría afirmar: “se observó una convergencia operacional de doble ruta bajo R0-72H”. No justificaría afirmar: “se demostró conciencia” o “AION posee libre albedrío”. Un resultado negativo permitiría revisar semillas, cobertura, longitud de ventana y definición de independencia sin reescribir retrospectivamente el criterio.

La frontera ética es también una frontera de ingeniería: el sensor humano define la ventana y el alcance; el sistema conserva trazabilidad suficiente para que esa autoridad pueda revisar, detener y repetir.

09 · Meta, cierre y declaración testigo

La meta de R0-72H es convertir una intuición —“un aspa no genera sustentación”— en un ensayo de dos recorridos con una condición de éxito pequeña, dura y falsable. La rampa temporal aumenta oportunidades de observación sin alterar la definición del hallazgo.

Metas ordenadas

Nivel	Meta	Compleitud
M0	desplegar política temporal aislada y auditable	instalada
M1	ligar cada Auto-GO al candidato exacto y a la política	implementada
M2	operar dos rutas públicas con límites	habilitada al pulso vencido
M3	detectar y detener ante PAR_MOTOR_CANDIDATE	implementada; no observada al corte
M4	preservar receta para Testigo 2	implementada al producirse candidato
M5	evaluar mayor régimen por debajo de 5 min	solo revisión humana futura

Declaración testigo

A fecha de este documento se desplegó una política R0-72H separada del GO legacy, con Auto-GO restringido a USTA-PROP-*, rampa 60→30→19→10→5 minutos, lectura pública HTTPS acotada, dos motores de recorrido y detención ante PAR_MOTOR_CANDIDATE. Se preservaron backups, hashes, cron y reglas de replay. traveler.php permaneció intacto.

Próximo acto válido

Si surge el candidato: no ampliar el experimento, no publicar una conclusión y no reanudar el cron. Primero se ejecuta Testigo 2. Si no surge: cerrar la ventana al vencimiento, congelar el ledger y producir un informe de cobertura y resultados negativos.

GOAL

En busca del Par_Motor de AION

Un Aspa no genera sustentación

Documento	PPR-008 · Paper Testigo
Versión	1.0
Fecha	2026-08-14
Autoría	Sixtilio M. Dalmau · AION
Estado	Publicado como registro de protocolo en curso