

N O R M A

Arquitectura de referencia

*Cómo se construye una herramienta de esta suite — la norma de la
que se deriva toda arquitectura de herramienta.*

v0.4 — borrador para corrección · estado 2026-08-14 · trece principios y diez
prohibiciones, todos con identificador y anclaje declarado

Kaspar Brönnimann

1 · Propósito, alcance, delimitación

Esta norma dice cómo debe estar construida una herramienta de la suite. Es la entrada del agente de arquitectura: a partir de ella — junto con los manifiestos, el catálogo de capabilities y la norma de uso — se deriva la arquitectura de una herramienta concreta.

Está cosechada, no inventada. Cada principio proviene o de una frase del manifiesto o de una decisión que ya se ha tomado y probado en un producto en curso. Donde un principio no tiene tal origen, se declara.

LO QUE ESTA NORMA REGULA

La construcción de una herramienta: capas, objetos-núcleo, estados, separación de tenants, pruebas.

Dónde pueden caer las decisiones y dónde no.

Qué bloques se usan y cómo interactúan.

Qué pruebas debe producir una herramienta.

LO QUE ESTA NORMA NO REGULA

Lo que una herramienta hace en sustancia. Eso está en la norma de método del producto.

Cómo se ve la superficie. Eso está en la identidad visual y el uso.

Lo que los bloques saben hacer en detalle. Eso está en el catálogo de capabilities.

Cómo se examina. Eso está en el concepto de prueba.

2 · Los principios, sus identificadores y su anclaje

Trece principios sostienen la arquitectura. Cada uno tiene un identificador fijo para que documentos posteriores puedan remitirse a ellos en lugar de repetirlos — una constatación de conformidad escribe «cumple RA-09» y ya no «sin contacto directo con el modelo».

El origen de un principio se descompone en tres indicaciones que hasta ahora se hallaban en una única columna y hacen cosas distintas. En esta versión se llevan por separado; qué significan las tres se dice en la sección 2.2.

IDENTIFICADOR	PRINCIPIO	ANCLAJE	DERIVACIÓN	PROBADO EN
RA-01	El núcleo decide, el modelo propone, el ser humano asume	PH-03, apoyado en PH-01	—	Technology Radar (E4)
RA-02	Ningún hecho autoritativo procedente del modelo	PH-10, afinado por WA-02	concepto de arquitectura, frontera de hechos	prueba de invariantes PIA
RA-03	Cada objeto-núcleo lleva su governance-mixin	PH-02	concepto de prueba	—
RA-04	Borrador y confirmación son campos separados	PH-03	—	protocolo de divergencia (E9), argumentos en vez de valores (E14)
RA-05	La aprobación es una puerta; un cambio la invalida	PH-03	concepto de prueba ADR-T04, ciclo de vida del saber	—
RA-06	Los cambios son transiciones, no rupturas	PH-08 (indirecto)	concepto de prueba	evaluaciones versionadas (E3, E9)
RA-07	Configuración antes que programación	ninguno, regla técnica	concepto de prueba ADR-T05	costes de operación y mantenimiento por rama

RA-08	La separación de tenants es estructura, no filtro	pendiente — véase 2.2	—	—
RA-09	Sin contacto directo con el modelo	PH-03, ejecutado por SC-01, SC-02	marco, portabilidad de modelo	—
RA-10	Cada resultado lleva su procedencia	PH-02, afinado por WA-02	—	—
RA-11	La incertidumbre es un campo, no una formulación	PH-04, apoyado en WA-03	—	grados de certeza (E16)
RA-12	Vocabulario controlado	ninguno, regla técnica	RA-05, RA-06	maleza de categorías (E7, E12)
RA-13	Una herramienta produce pruebas, no solo resultados	PH-02	concepto de prueba	—

2.1 · La regla para los identificadores

Un identificador nunca se redefine.

Los identificadores solo valen algo si «cumple RA-09» significa dentro de cinco años lo mismo que hoy. Por eso rigen tres reglas:

- Los principios nuevos se añaden al final y reciben el siguiente número libre. No se intercalan por tema, aunque su contenido perteneciera a otro lugar.
- Si un principio se retira, su número queda ocupado y se lleva como retirado. No se reasigna.

- Si el contenido de un principio cambia sustancialmente, se trata de un nuevo principio con un nuevo número — no del mismo número con otro contenido. El antiguo remite al nuevo.

Las mismas tres reglas rigen para los identificadores del espacio negativo (NR-01 ..., capítulo 9).

Esta versión numera por primera vez; el orden se ha elegido por eso aún temáticamente. A partir de ahora queda congelado. Es la misma disciplina que con los invariantes del modelo de calidad y con las decisiones de arquitectura del concepto de prueba.

2.2 · Qué significa anclaje — y qué no

Tres indicaciones que hacen cosas distintas y por eso no pertenecen a una única columna:

- Anclaje — una frase del manifiesto. Fundamenta por qué la regla vale también para una herramienta que aún no existe. Solo el nivel superior vale para todo; si un principio se apoya exclusivamente en un manifiesto de dominio, en sentido estricto vale solo en ese dominio.
- Derivación — otra norma de la suite. Muestra de qué se sigue la regla y sostiene dentro del orden, no hacia fuera.
- Probado en — una decisión que se ha tomado y probado en un producto. Es evidencia, no fundamentación, y puede quedar vacío: una celda vacía significa «aún no probado en ningún lugar» y es información, no vergüenza.

Una prueba no es un anclaje.

No responde a la pregunta de por qué una regla vale para una herramienta que en su momento no estaba construida. Si la fundamentación de un principio estaba solo en esta columna, no era una norma de la familia, sino una experiencia particular generalizada — quizá correcta, pero no defendible ante un cliente.

El anclaje no es una deducción.

Lo que se exige es un fundamento portante, no una derivación lógico-formal — la misma regla que con los identificadores del manifiesto: un identificador se nombra cuando sostiene, no cuando meramente encaja. La proximidad es por eso distinta; los anclajes indirectos están marcados como tales en la tabla.

Una estrategia de implementación no es una derivación.

La pregunta no es si un valor encaja con la regla, sino si la regla se sigue necesariamente de él o solo es un camino posible hacia ella. La configurabilidad favorece la autodeterminación, pero la autodeterminación podría producirse también de otro modo; por eso RA-07 se lleva como regla técnica y no se ancla.

La cadena no es de uno a uno. Una frase del manifiesto puede sostener varios principios, un principio puede honrar varios valores, y una regla técnica puede ser necesaria sin poseer contenido moral. Un cuerpo de reglas en el que sin excepción cada regla se sigue de un valor está construido a posteriori y no es limpio.

Dos principios no portan por tanto expresamente ningún anclaje: RA-07 y RA-12. Uno no lo porta todavía: RA-08 — la frase del manifiesto correspondiente está formulada pero pospuesta hasta que se regule quién puede destinar un saber a compartirlo y qué sucede, en caso de revocación, con el saber derivado de él.

3 · Los trece principios

RA-01 *El núcleo decide, el modelo propone, el ser humano asume*

Tres niveles que nunca coinciden. El modelo produce propuestas. El núcleo determinista decide qué es válido — estados, conteos, resultados de examen, probabilidad. El ser humano decide qué se hace vinculante.

Lo garantizado se impone de manera determinista, no se pide a nivel del prompt.

Un compromiso que solo está en el prompt se cumple reformulado. Por eso el núcleo rebaja una aprobación cuando faltan los requisitos, y el lenguaje solo lo explica. Los rebajes son unilaterales: un juicio más estricto nunca se atenúa.

***Verificable:** No existe ningún camino por el que un resultado de modelo se haga vinculante sin examen del núcleo.*

RA-02 *Ningún hecho autoritativo procedente del modelo*

El modelo puede clasificar, sintetizar, formular, encaminar e interpretar. No puede producir referencias, cifras, enunciados jurídicos ni pruebas. Estos proceden de fuentes verificables o de herramientas.

***Verificable:** Todo valor con carácter de prueba lleva una fuente depositada; si falta, no puede ser emitido. Los conteos en el resultado provienen del núcleo, no del texto.*

Desde el 14 de agosto de 2026 este principio está anclado en el nivel superior: PH-10 — lo observado y lo supuesto son dos cosas distintas; lo que aparece como hecho nombra su procedencia, y procedencia significa de dónde procede algo, no quién lo ha dicho. Antes, RA-02 se apoyaba solo en una frase del manifiesto ligada a un dominio y valía por tanto, en sentido estricto, solo allí donde se trae algo nuevo de fuera.

Nota: sobre esta formulación existe una necesidad de aclaración abierta frente al concepto de prueba (capítulo 13, hallazgo 1). Esta norma emplea la formulación más amplia, porque la estrecha excluiría el examen sustantivo por un modelo.

RA-03 *Cada objeto-núcleo lleva su governance-mixin*

Los objetos-núcleo son las cosas que una herramienta conduce y asume — un mandato, una entrada de radar, una evaluación, un protocolo, un modelo de proceso. Cada uno lleva los mismos campos.

`created_at · updated_at · version · created_by · status`

Con ello cada objeto es datable, versionable y atribuible a un autor — humano o modelo. El historial surge como subproducto: quien almacena evaluaciones versionadas recibe gratuitamente la pregunta «¿hacia dónde se ha movido la apreciación?».

Verificable: Ningún objeto-núcleo sin mixin completo; created_by distingue humano y modelo.

RA-04 *Borrador y confirmación son campos separados*

No solo estados separados: campos separados. El borrador del modelo se conserva junto a la versión final humana.

Fig. 1 — Estructura de un objeto-núcleo. El borrador no desaparece.

Dos razones. Primero, la diferencia entre borrador y versión final es el protocolo de divergencia — de él se aprende la propia rejilla de evaluación y se afina la instrucción. Segundo, un ser humano examina una fundamentación, no un número; por eso el borrador contiene argumentos y no solo valores.

Verificable: El campo de confirmación para el ser humano que examina es obligatorio; sin él, el objeto no es vinculante. El borrador nunca se sobrescribe.

RA-05 *La aprobación es una puerta, y un cambio la invalida*

Fig. 2 — Un patrón que recurre en cada producto.

Este patrón aparece en la suite en cuatro lugares, y ha de funcionar en todos por igual: en la aprobación de saber, en la aprobación de un mandato, en la puerta entre validación interna y prueba con cliente, en la ampliación de un vocabulario controlado.

Las aprobaciones conocen tres formas, no dos: aprobado · aprobado con condición · no aprobable. La forma intermedia es prácticamente la más importante — permite avanzar sin perder un punto abierto, porque la condición se documenta con responsabilidad y plazo.

***Verificable:** Un cambio en un objeto aprobado devuelve el estado a cero; no hay modo de conservar una aprobación y cambiar el contenido.*

RA-06 *Los cambios son transiciones, no rupturas*

Los demás principios describen cómo es una herramienta en un momento. Este describe cómo puede cambiar en el tiempo. Sin él, los otros doce tienen fecha de caducidad.

Quien cambia una norma, un vocabulario o un esquema cambia el significado de los datos ya almacenados.

Sin transiciones consignadas, el aprendizaje no es posible — ni para una organización ni para un ser humano. Quien cambia silenciosamente el significado de valores almacenados ya no tiene historial, sino solo un stock que se parece a un historial. Es el modo más silencioso de destruir el saber: ninguna prueba se dispara, y nadie lo advierte.

Probado en el Technology Radar: cuando allí se endurece la rejilla de evaluación sin consignar la transición, el mismo escalón significa algo distinto antes y después del cambio. El historial — el verdadero valor del stock — queda entonces silenciosamente falseado. Sigue teniendo buen aspecto y ya no es cierto.

Cuatro reglas mantienen limpio el eje temporal:

- Cada objeto almacenado sabe bajo qué versión de la norma, del vocabulario y de la rejilla de evaluación ha nacido. Es la ampliación del triple de versión de los métodos a los datos.
- Cada cambio de una norma trae consigo una transición: los datos de stock siguen valiendo, o hay una correspondencia con la nueva versión, o se marcan expresamente como históricos. Una reinterpretación silenciosa no es ninguna de las tres posibilidades.
- La rejilla de evaluación misma es un objeto versionado, no un saber en la cabeza. Su historial pertenece a la prueba.
- Una prueba emitida es inmutable. Una corrección es una nueva versión, no una edición de la antigua.

Una ruptura sigue estando permitida — no todo se puede mantener compatible en el tiempo. Pero se explica, se data y se aplica a los datos de stock. Solo se prohíbe la ruptura inadvertida.

***Verificable:** Para cada objeto almacenado puede determinarse la versión bajo la que nació. Ningún cambio en un vocabulario controlado sin regla de transición para los datos de stock. Una prueba emitida no puede modificarse, solo sustituirse.*

Relación con RA-05: ambos protegen algo del cambio inadvertido — RA-05 el estado de un objeto, RA-06 el significado de sus valores. Y RA-06 es la condición para que RA-12 no se vuelva una trampa: un vocabulario controlado que se amplía sin transición reinterpreta su propio pasado.

RA-07 *Configuración antes que programación*

Lo que se distingue por tenant, por producto o por caso no se programa, sino que se configura — como archivo versionado junto al código, no en una base de datos y no en un sistema ajeno.

Métodos y procedimientos	El CÓMO funcional cambia más rápido que el código.
Catálogos y vocabularios controlados	La ampliación es un acto consciente, no un efecto colateral.
Reglas de examen y tolerancias	Son objeto de discusión funcional, no técnica.
Casos de prueba	Herramienta compartida, casos específicos por producto.
Identidad visual por tenant	Logo y colores son configuración, no construcción a medida.
Asignaciones: función a rol, tarifas	Se distinguen por organización.

Verificable: *Una petición de un tenant de esta lista no exige ningún cambio de código.*

Todos los archivos de configuración están versionados con el código y son diffables.

RA-08 *La separación de tenants es estructura, no filtro*

Fig. 3 — Tres niveles; se hereda hacia abajo, nunca en travesía.

Una separación que descansa sobre un filtro se sostiene solo mientras nadie olvide el filtro. Por eso cada objeto relacionado con un tenant porta estructuralmente su pertenencia, y lo compartido está expresamente marcado — no reconocible por el hecho de que falte una indicación.

Tres niveles: la base compartida sin tenant, el tenant con sus deltas, la instancia o el proyecto con sus valores. Se hereda hacia abajo; un acceso en travesía por encima de fronteras de tenants no existe — ni como camino, ni como autorización.

Verificable: *Un acceso sin referencia de tenant entrega exclusivamente stocks compartidos. No hay consulta que vea dos tenants a la vez.*

Anclaje pendiente. La frase del manifiesto correspondiente a este principio está formulada — el saber confiado permanece separado; solo se comparte lo que expresamente está

destinado a compartirse —, pero pospuesta. Mientras no se regule quién puede destinar algo a compartirse y qué ocurre, en caso de revocación, con el saber derivado de él, en el manifiesto figuraría una frase que la operación en curso quebraría. Hasta entonces, RA-08 es el único principio sin anclaje, y así se declara.

RA-09 *Sin contacto directo con el modelo*

Ningún producto habla directamente con un proveedor de modelo. Todas las llamadas — también los embeddings — pasan por la capability llamada IA, que mantiene un adaptador por proveedor y antepone la seudonimización.

Una llamada sin seudonimización no es técnicamente posible, no solamente prohibida.

De ahí se siguen dos propiedades por el precio de una: la capa de protección no puede evitarse porque no hay un segundo camino. Y un cambio de modelo se convierte en cuestión de adaptador en lugar de redesarrollo.

Verificable: *En el código existe exactamente un único lugar con contacto con el proveedor.*

La prueba de llamada retiene proveedor, modelo y versión del modelo.

RA-10 *Cada resultado lleva su procedencia*

La procedencia es subproducto de la ejecución, no prestación de memoria del modelo. Quien hace redactar la procedencia al final obtiene una narración plausible en lugar de una prueba.

Por sección de un resultado se llevan: los tipos de fuentes utilizados, los enunciados clave con su procedencia, las preguntas de retorno planteadas junto con las rechazadas, los supuestos tomados y lo que ha quedado abierto.

Verificable: *La procedencia puede reconstruirse a partir de los datos de ejecución sin preguntar a un modelo. Todo valor mostrado es rastreable hasta su origen.*

RA-11 *La incertidumbre es un campo, no una formulación*

Una herramienta que reviste su incertidumbre de palabras la hace inanalizable. Grado de certeza y edad de un enunciado son campos estructurados — confirmado, probable, no confirmado — y aparecen en la superficie, no solo en el registro.

Verificable: Los enunciados con carácter valorativo llevan un grado de certeza; las apreciaciones llevan su fecha y se marcan como vencidas cuando envejecen.

RA-12 *Vocabulario controlado*

Se clasifica exclusivamente en listas existentes. Un modelo ordena; no amplía. Los términos nuevos requieren una aprobación consciente según el patrón de RA-05 y una transición para los datos de stock según RA-06.

Sin esta regla surgen en meses decenas de categorías y la analizabilidad se pierde — los datos parecen completos y ya no llevan nada.

Verificable: Un valor fuera de la lista se rechaza, no se crea en silencio.

RA-13 *Una herramienta produce pruebas, no solo resultados*

A la pregunta «¿está examinado?» no responde una palabra, sino un documento. Cinco pruebas surgen en esta suite; cuáles lleva una herramienta depende de lo que hace.

PRUEBA	LO QUE CONSIGNA	QUIÉN LA PRODUCE
Procedencia	Origen de los enunciados de un resultado.	el núcleo, en la ejecución
Protocolo de examen	Resultado de un examen sustantivo de un resultado.	el método que examina
Protocolo de prueba	Qué tipos de prueba corrieron con qué resultado.	el test-runner

Prueba de llamada	Proveedor, modelo y versión por llamada al modelo.	la capability llamada IA
Protocolo de divergencia	Diferencia entre borrador del modelo y versión final humana.	el núcleo, en la confirmación

Verificable: *Cada prueba lleva el governance-mixin y es legible sin la herramienta que la produce.*

4 · La arquitectura por capas

Fig. 4 — Cinco capas y una regla de llamada.

CAPA	LO QUE VIVE AQUÍ	LO QUE NO DEBE ESTAR AQUÍ
Superficie	Presentación e interacción según la norma de uso.	Reglas funcionales, decisiones, llamadas directas a capabilities.
Métodos y procedimientos	El CÓMO funcional: procedimientos, rúbricas, catálogos, vocabularios — configurados.	Bloques técnicos, gestión de estados, conteos.
Núcleo	Modelo de dominio, estados, aprobaciones, examen de reglas, conteos, procedencia — y la ejecución de los métodos.	Juicio funcional sin norma; código específico de proveedor.
Capabilities	Aptitudes técnicas según contrato de catálogo.	Saber funcional de cualquier tipo. Una capability sabe CÓMO, no QUÉ está bien.
Persistencia	Almacenamiento con separación estructural de tenants.	Lógica funcional, transiciones de estado.

La capa de ejecución pertenece al núcleo.

Lo que gestiona bucles de retorno, estados, handoffs, criterios de abandono y procedencia no forma parte de un método, sino que lo ejecuta. De ahí se sigue la regla empírica para el reparto: los criterios verificables de manera determinista pertenecen al núcleo, los que han de apreciarse en lo funcional al método. Esta capa está llevada en el catálogo de capabilities como workflow y aún no está construida — es el punto abierto más grande de esta arquitectura.

5 · Objetos-núcleo y estados

Una herramienta conduce un número manejable de objetos-núcleo. Llevan el governance-mixin, recorren el ciclo de aprobación de P5 y son las anclas de las que cuelgan procedencia y pruebas.

Dos patrones han probado su valor y son vinculantes:

- La cosa seguida y sus pruebas se separan. Una entrada tiene identidad propia; observaciones fechadas se acoplan a ella. Solo así puede mostrarse movimiento en el tiempo.
- La evaluación es un objeto propio, versionado, con marca de tiempo — no un campo en la cosa evaluada. El historial surge por eso gratis.

***Verificable:** Para cada objeto-núcleo puede responderse quién lo llevó a qué estado cuándo y en qué se apoya.*

6 · Superficie: lo que la norma de uso exige a la arquitectura

La norma de uso está en un documento propio. Para la arquitectura es decisivo que varios de sus principios solo puedan honrarse si el modelo de datos los sostiene. Se listan por eso aquí como exigencias y no se dejan a la superficie.

PRINCIPIO DE USO

LO QUE LA ARQUITECTURA DEBE LLEVAR PARA ELLO

El ser humano decide, el sistema propone

Campos de borrador y confirmación separados (RA-04); un estado que distingue borrador de vinculante.

Ningún enunciado sin fuente

Procedencia por valor recuperable (RA-10); indicación de fuente como campo obligatorio, no como convención textual.

Hacer visible la incertidumbre

Grado de certeza y fecha como campos (RA-11); vencimiento calculable.

Igual a través de la familia

Mismo vocabulario de estatus en todos los productos — por tanto en el núcleo, no por superficie (RA-03, RA-12).

Confianza y dignidad

Identificadores seudónimos en lugar de nombres en claro en la visualización; revelación como paso consciente según RA-05.

A la inversa: los valores concretos de examen de la norma de uso — tamaños mínimos, contrastes, el color nunca como única señal — pertenecen al examen de reglas y no al cuidado del desarrollador.

7 · Configuración y ampliación por tenants

La suite no gana su dinero construyendo para cada cliente algo distinto, sino porque lo mismo puede configurarse de manera distinta. Es una exigencia de arquitectura, no una cuestión comercial.

NIVEL	LO QUE HAY AQUÍ	QUIÉN LO MANTIENE
Base	Métodos, catálogos, vocabularios, stocks de referencia, reglas de examen.	responsabilidad de producto
Tenant	Deltas respecto a la base: procedimientos propios, asignaciones, tarifas, identidad visual, reglas de examen propias.	asesoría conjunta con el tenant
Instancia	Valores de un caso concreto.	usuario

Reglas: un delta de tenant sobrescribe la base, no la sustituye. Un delta sin asignación a su contrapartida en la base es inválido. Y un delta de un tenant no es visible para otros tenants — ni siquiera como plantilla.

8 · Operación y entornos

Cuatro entornos con distinta configuración: desarrollo con interfaces simuladas, entorno de prueba interno, entorno de integración como imagen de la producción con interfaces reales, y operación productiva sin operación de prueba.

Para la arquitectura se sigue de ello sobre todo una cosa: una herramienta debe poder correr sin cambio de código contra sistemas circundantes simulados o reales. El modo de interfaz es configuración y se consigna en la prueba, para que no surja falsa seguridad.

La promoción corre por pasos y nunca se salta. Cada producto ocupa su propio rango de puertos en el host compartido; las herramientas comparten la pila operativa, no sus datos.

Verificable: Un cambio del modo de interfaz no exige nueva entrega. Ningún producto accede a los stocks de datos de otro.

9 · Lo que una herramienta no debe hacer

El espacio negativo es la parte más verificable de una arquitectura, porque las infracciones son concretas. Cada prohibición lleva un identificador para que un hallazgo pueda invocarla, en lugar de hablar en general de «principios fundamentales». Para estos identificadores rige la misma regla que para los principios: nunca se redefinen.

IDENTIFICADOR	UNA HERRAMIENTA NO DEBE ...	PRINCIPIO ASOCIADO
NR-01	tomar una decisión que corresponde a un ser humano — tampoco por valor por defecto, automatismo silencioso o una verdad precargada.	RA-01
NR-02	producir una referencia, una cifra o una prueba que no provenga de una fuente.	RA-02
NR-03	dirigirse directamente a un modelo.	RA-09
NR-04	traspasar una frontera de tenants — en lectura, en escritura o como plantilla.	RA-08
NR-05	conservar un estado aprobado mientras el contenido cambia.	RA-05
NR-06	ampliar por propia autoridad un vocabulario controlado.	RA-12

NR-07	emitir un resultado sin procedencia.	RA-10
NR-08	garantizar una seguridad únicamente mediante instrucción a un modelo.	RA-01
NR-09	cambiar el significado de valores almacenados sin registrar la transición.	RA-06
NR-10	editar con posterioridad una prueba emitida.	RA-06, RA-13

Un hallazgo que invoca el espacio negativo nombra el identificador.

Sin identificador es una afirmación. La invocación del espacio negativo pesa mucho en varios procedimientos de la suite — convierte una reserva en un hallazgo obligatorio —, y lo que pesa tanto ha de ser direccionable.

10 · Conformance

Una arquitectura de herramienta es conforme si, para cada uno de los trece principios, expone cómo lo honra — o fundamenta por qué no es aplicable. El lugar de esta exposición es la constatación de conformidad del marco.

Como los principios llevan identificadores, la lista 1 de la constatación es una tabla corta en lugar de un ensayo — por fila un identificador, la implementación y la referencia. La lista 2 lleva las desviaciones con fundamentación, la lista 3 los lugares donde esta norma calla; la tercera lista es la retroalimentación a este documento.

konformitaet:

```
- { prinzip: RA-01, erfuellt: ja, fundstelle:
"Núcleo/Aprobación, sección 4.2" }
- { prinzip: RA-06, erfuellt: ja, fundstelle: "Versión de
esquema por registro" }
- { prinzip: RA-08, erfuellt: nein, begruendung: "mono-tenant,
deliberado", liste: 2 }
- { prinzip: RA-12, erfuellt: offen, hinweis: "vocabulario no
cerrado", liste: 3 }
```

Con ello la constatación es verificable por máquina: un verificador de conformidad puede establecer si, para cada identificador, existe una fila — mucho antes de que alguien lea su contenido.

Misma norma no significa mismo comportamiento — la aprobación vale por modelo y por superficie. Una implementación está aprobada solo cuando supera los invariantes obligatorios y la verdad de referencia en el modelo desplegado. La prueba retiene modelo y versión.

11 · Cómo surge una herramienta de esta norma

1. Fijar propósito y delimitación de la herramienta — qué hace y qué expresamente no.

2. Nombrar los objetos-núcleo: ¿qué conduce la herramienta, qué es mero atributo?
3. Por objeto-núcleo, fijar el ciclo de aprobación y determinar los campos de confirmación.
4. Seleccionar capabilities del catálogo; declarar lo que falte como solicitud al catálogo.
5. Trazar la frontera de configuración: ¿qué es base, qué es delta de tenant, qué es valor de instancia?
6. Fijar las pruebas: ¿cuál de las cinco produce esta herramienta?
7. Fijar el eje temporal: ¿qué versiones lleva consigo cada objeto-núcleo y qué ocurre con los datos de stock cuando vocabulario o rejilla de evaluación cambian?
8. Traducir las frases de manifiesto aplicables o anotarlas como no aplicables.
9. Elaborar la constatación de conformidad sobre los trece identificadores.

12 · Puntos abiertos de esta norma

PUNTO

ESTADO

La capa de ejecución no está construida.

El núcleo ejecuta métodos — hasta ahora cada producto lo hace a su manera. Punto abierto más grande.

Falta un inventario de los productos en curso.

Esta norma describe cómo se construye. Hasta qué punto los productos existentes le corresponden aún no está medido.

El vocabulario de estatus no está unificado.

RA-03 exige los mismos estados en la familia; los valores concretos han crecido por producto. La unificación es ella misma un caso para RA-06.

Los datos de stock no conocen todavía su versión.

RA-06 exige que cada objeto almacenado lleve consigo su versión de norma, vocabulario y rejilla. En los productos en curso esto es solo parcialmente el caso — el retroajuste necesita a su vez una transición.

Las autorizaciones no están reguladas.

Esta versión regula la separación de tenants, no roles y derechos dentro de un tenant.

Relación con las decisiones de arquitectura existentes.

Las decisiones existentes están aquí incorporadas, pero aún no formalmente reconducidas a esta norma.

RA-08 aún no lleva anclaje.

La frase del manifiesto está formulada y pospuesta hasta que se regulen consentimiento al compartir y revocación. Hasta entonces declarado, no disimulado.

La procedencia no es transitiva para un saber derivado.

RA-10 lleva una indicación por valor. Si de A y B surge un C y de ahí un E, tras una revocación de A no puede establecerse que E dependa de ello. Conceptualmente esto exige un grafo de procedencia. Anotado, no construido.

13 · Hallazgos al derivar

HALLAZGO

CONSECUENCIA

RA-08 no tiene frase de manifiesto.

Hecho el 14/08/2026: la frase está formulada — el saber confiado permanece separado — y pospuesta hasta que consentimiento y revocación se regulen. RA-08 se lleva hasta entonces sin anclaje.

RA-06 tampoco tiene frase de manifiesto.

Hecho: anclado en PH-08 (recordar como capacidad permanente), expresamente marcado como indirecto. La fundamentación se ha formulado a la vez de manera neutra respecto a los productos.

La frontera de hechos está formulada dos veces de manera distinta.

Esta norma emplea la formulación más amplia. PH-10 ancla ahora el principio en el nivel superior; la contradicción con el concepto de prueba queda de ello inalterada, y RA-02 permanece en ese aspecto bajo reserva.

Tres nociones de prueba no estaban delimitadas.

El capítulo 3.12 las delimita. Permanecen separadas pero llevan el mismo governance-mixin.

La norma de uso plantea exigencias al modelo de datos.

El capítulo 6 las hace explícitas. Sin esta traducción, varios principios de uso no serían honorables, sino solo afirmados.

La pregunta por el anclaje solo se había planteado dos veces.

Al examinar los trece identificadores se mostró: cinco anclados, dos solo ligados a un dominio, seis sin respuesta. Tras las decisiones del 14/08/2026 son diez anclados, dos expresamente técnicos, uno pendiente.

Cuatro de cinco manifiestos son manifiestos de producto.

Solo el nivel superior vale para todo. Mientras los manifiestos de dominio estén cortados según herramientas, cada principio que se apoye en ellos hereda su frontera de dominio. Las formulaciones neutras respecto a los productos ya están en el registro de identificadores, en los

documentos fuente aún no.