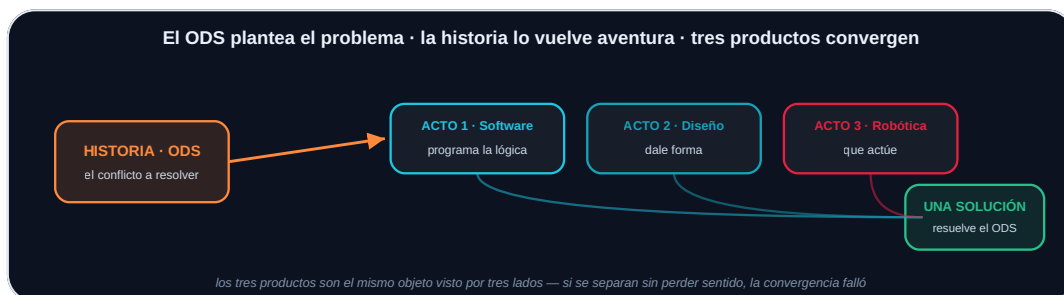


# Línea Ares — El libro como historia: narrativa ODS + tres productos convergentes

ROBOTSchool · Documento maestro (fuente única) Versión 0.1 —  
Junio 2026

El modelo que **amarra todo**: cada libro Ares deja de ser una lista de microproyectos y se vuelve **una historia anclada a su ODS**. Dentro de esa historia, los niños construyen **tres productos que convergen en una sola solución**: un **producto de software** (programación), un **producto de diseño** y un **producto de robótica**. La historia crea la necesidad; los tres productos la resuelven. Es la **convergencia de los tres hilos** (doc 04) ahora con **hilo narrativo**.



## 1. La idea en una frase

El ODS plantea un problema → la historia lo vuelve aventura → para resolverlo, los niños programan, diseñan y robotizan → los tres productos se unen en una solución.

Esto refina —no reemplaza— el modelo editorial (doc 11) y el modelo de convergencia (doc 04): la convergencia ya no se “pide” al final, **emerge de la trama**.

## 2. Por qué este modelo es fuerte

- **El ODS deja de ser etiqueta y se vuelve motivo.** El niño no “trabaja el ODS 6”; *ayuda a un pueblo que se quedó sin agua limpia*. El propósito mueve el aprendizaje.
- **La convergencia se siente natural, no forzada.** Hoy juntar programación + diseño + robótica puede parecer un requisito. En una historia, los tres son **partes obvias de la misma solución**: el cerebro (software), la forma (diseño) y el cuerpo que actúa (robótica).
- **La narrativa sostiene la práctica.** Querer saber *cómo sigue la historia* mantiene al niño resolviendo retos —exactamente lo que pide la práctica deliberada de la ruta (doc 21 §7).
- **Da un arco emocional.** Conflicto → intentos → solución → celebración. El cierre (el robot funciona, el pueblo recupera el agua) es la recompensa.
- **Es memorable.** Un niño olvida “el proyecto del sensor de humedad”; recuerda “cuando salvamos la huerta de doña Rosa”.

## 3. Estructura narrativa de un libro (los tres actos = los tres productos)

Cada libro sigue un **arco de tres actos**, uno por producto, encadenados por la historia:

| Acto          | Hilo | Producto        | Qué construye el niño | Pregunta del acto                       |
|---------------|------|-----------------|-----------------------|---|
| <b>Gancho</b> | —    | (conflicto ODS) | —                     | “¿Qué está pasando y a quién ayudamos?” |

| Acto                      | Hilo         | Producto                    | Qué construye el niño  | Pregunta del acto                      |
|---------------------------|--------------|-----------------------------|--|--|
| <b>Acto 1 · La mente</b>  | Programación | <b>Producto de software</b> | Programa la lógica: una historia programable, un juego, una simulación o el código del personaje | “¿Cómo <i>piensa</i> la solución?”     |
| <b>Acto 2 · La forma</b>  | Diseño       | <b>Producto de diseño</b>   | Diseña y fabrica: la estructura (papel→MDF/CAD), la interfaz o el empaque                        | “¿Qué <i>forma</i> tiene la solución?” |
| <b>Acto 3 · El cuerpo</b> | Robótica     | <b>Producto de robótica</b> | Integra todo en un dispositivo real que actúa sobre el mundo                                     | “¿Cómo <i>actúa</i> en el mundo real?” |
| <b>Cierre</b>             | Convergencia | <b>Una sola solución</b>    | Los tres productos se unen y resuelven el problema de la historia                                | “¿Lo logramos? ¿Qué aprendimos?”       |

**Regla de oro:** los tres productos no son tres tareas sueltas — son **el mismo objeto visto por tres lados**. El software es el cerebro del robot que tiene la forma que se diseñó. Si se pueden entregar por separado sin perder sentido, la convergencia falló.

**Orden flexible:** software→diseño→robótica es el orden natural (“piensa, dale forma, hazlo actuar”), pero la historia puede intercalarlos. Lo no negociable es que **converjan al cierre**.

## 4. Qué son los “tres productos” en cada banda (alcance realista)

Aquí va mi advertencia como mentor: **tres productos es mucho para un niño pequeño**. Si en Transición pedimos un software + un diseño + un robot “completos”, abrumamos y nada queda bien. La solución es **escalar el alcance**: en las primeras bandas los productos son **pequeños, muy guiados y físicos**; crecen en autonomía y abstracción con la edad.

| Banda                           | Producto de software  | Producto de diseño   | Producto de robótica   |
|---------------------------------|---|--|--|
| <b>Exploradores</b><br>(T-2°)   | Una <b>historia programable corta</b> en bloques-ícono / un personaje en Scratch que responde a Makey Makey | Un <b>objeto en papel/cartón</b> (recortable troquelado), decorado | Un <b>circuito simple</b> con Makey Makey o mecanismo (el objeto “cobra vida”) |
| <b>Constructores</b><br>(3°-5°) | Un <b>juego o historia programable</b> (Scratch/mBlock) con bucles y condicionales                          | Una <b>estructura en MDF</b> (mecanismo) diseñada y armada         | Un <b>robot Arduino</b> con sensor→actuador que cumple una misión del cuento   |
| <b>Inventores</b><br>(6°-8°)    | Código <b>en texto</b> (MicroPython) con funciones; una simulación o app del personaje                      | Diseño <b>CAD + fabricación</b> (láser/3D); interfaz básica        | Dispositivo <b>Pico</b> que sensa y actúa según la lógica                      |
| <b>Innovadores</b><br>(9°-11°)  | <b>Sistema</b> con datos (Python + web + SQL + dashboard)   | <b>UI/UX</b> y diseño para fabricación del producto final          | Sistema <b>ESP32/IoT</b> conectado que resuelve el problema a escala           |

*La historia es la misma estructura en las 4 bandas; lo que sube es la escalera de abstracción y la autonomía (doc 21 §2).*

## 5. Cómo enhebra la historia programable (doc 21

### §4.1)

El **Acto 1 (software)** es el lugar natural de la **historia programable**: el niño escribe código para que los personajes del cuento actúen. Ese mismo personaje suele convertirse en el **Acto 3** en el **robot físico** — el personaje de la pantalla *se vuelve real*. Ese salto (“programé al personaje, ahora es mi robot”) es uno de los momentos más potentes del modelo y refuerza que **el código del juego y el del robot son el mismo pensamiento**.

---

## 6. Ejemplo encarnado · Constructores 3° · ODS 6 (Agua limpia)

**Gancho:** La huerta de doña Rosa se está secando; el agua llega solo a veces y nadie sabe cuándo regar.

- **Acto 1 · Software:** los niños programan, en una **historia programable**, a *Gota* —un personaje— para que recorra la huerta y “riegue” solo las plantas secas. Aprenden **bucle** (“recorre todas las plantas”) y **condicional** (“si la planta está seca, riega”). *Puente:* lo reescriben en mBlock.
- **Acto 2 · Diseño:** diseñan y arman en **MDF** el soporte de un mini-sistema de riego (el “cuerpo” de Gota).
- **Acto 3 · Robótica:** montan un **Arduino** con sensor de humedad → bomba/LED: *si el suelo está seco, riega* — la misma lógica del Acto 1, ahora real.
- **Cierre:** el sistema riega solo cuando hace falta. La huerta revive. Los tres productos = **un regador inteligente**. Conecta con el ODS 6: usar el agua justa, sin desperdicio.

*La lógica del condicional aparece tres veces (en el juego, en el bloque, en el robot): eso es el espiral y la convergencia trabajando juntos.*

---

## 7. Impacto en la producción (qué hay que ajustar)

- **Los libros se reescriben con arco narrativo** (no como secuencia de microproyectos sueltos). El motor `build_books.py` necesita un campo de **historia/personaje** y la marcación de los **tres productos** por libro.
  - **Prioridad bandas 1 y 2** (como pediste): ahí el modelo narrativo rinde más. En 3 y 4 la historia se vuelve más “misión/caso” que cuento, pero el arco de tres productos se mantiene.
  - Cada libro necesita **arte de personaje/escenario** (parte del hilo de diseño) y, para el Acto 1, una **historia programable** (Reeborg/Scratch al inicio; motor propio después).
  - Se debe definir el **arco narrativo global de Ares** (un universo/ personajes que crezcan con el niño de Transición a 11°) para dar continuidad — pendiente de §9.
- 

## 8. Advertencias de mentor

1. **No infles los tres productos.** En las bandas pequeñas, dos de los tres pueden ser muy guiados (casi “kit”) y solo uno verdaderamente abierto. Mejor tres cosas pequeñas y conectadas que tres grandes e inconexas.
  2. **La convergencia es el criterio de calidad, no la cantidad.** Un libro está bien hecho si los tres productos **no tienen sentido por separado**. Úsese la “prueba de quita un hilo” (doc 04).
  3. **La historia no puede tapar el concepto.** Igual que el robot no debe tapar la programación (doc 21 §1), la narrativa no debe volverse adorno: el código sigue siendo **la única llave** para avanzar el cuento.
  4. **Cuida la carga del docente.** Tres productos por libro exige guía clara; los comodines (doc 13) y la plantilla de lección (doc 20) son los que lo hacen manejable.
- 

## 9. Pendiente

1. **Definir el arco narrativo global de Ares** (universo, personajes que evolucionan por banda, tono por edad).

2. **Reescribir un libro modelo** con el arco de tres actos (sugiero Constructores 3° · ODS 6, el del ejemplo).
3. **Prototipar el Acto 1** como historia programable jugable (en curso).
4. Ajustar `build_books.py` / `books_data.py` para los campos de historia y los tres productos.
5. Encargar el **arte de personaje** (hilo de diseño) por banda.

*Fin del documento v0.1. Unifica convergencia (04), modelo editorial (11) y la ruta de programación (21) bajo un arco narrativo anclado al ODS.*