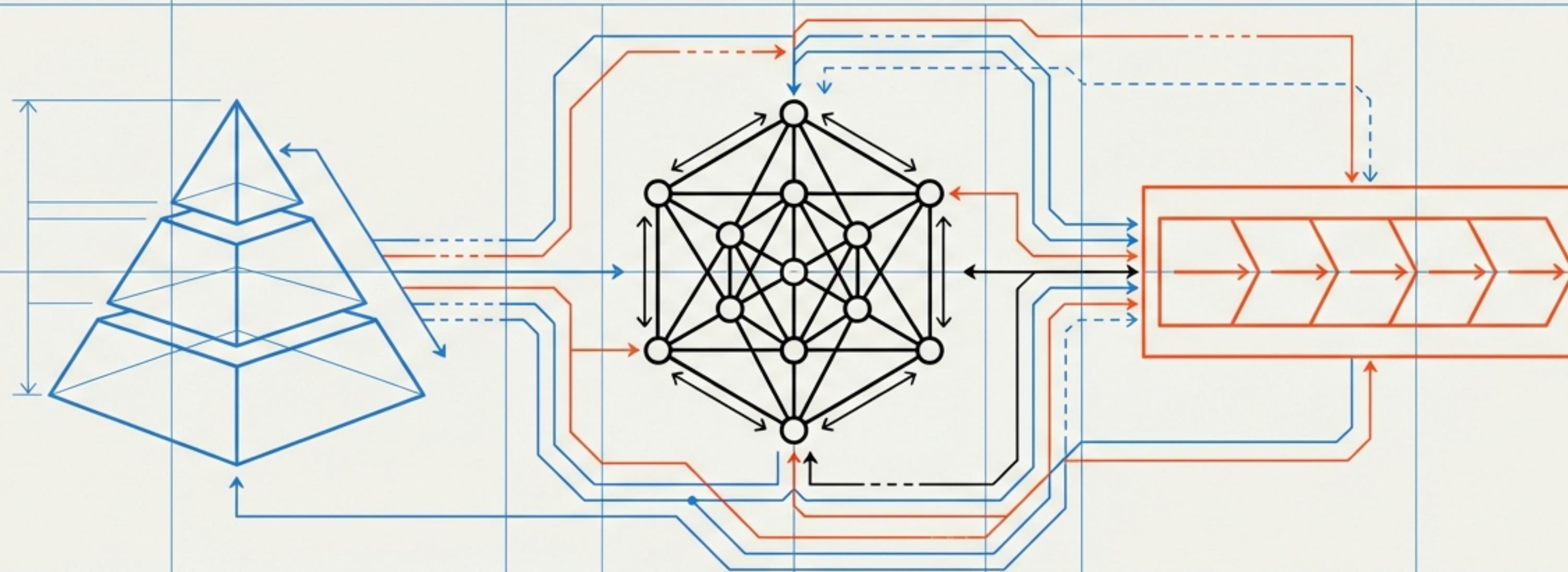
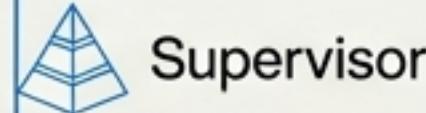


PATRONES DE ARQUITECTURA

PARA SISTEMAS MULTI-AGENTE



Estrategias de Orquestación:
Supervisor, Consenso y Pipelines



Supervisor



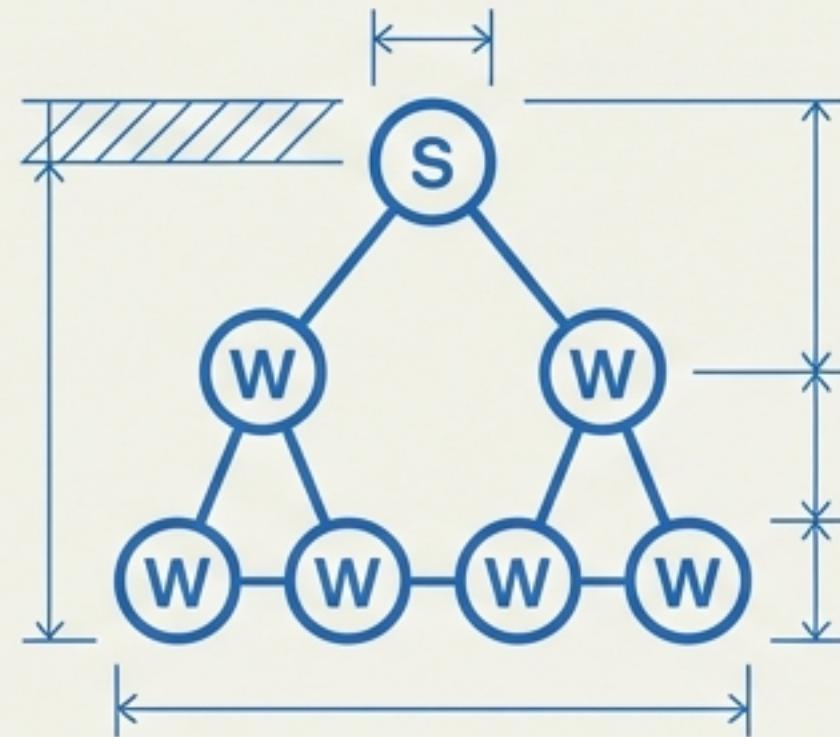
Consenso

Pipelines

MANUAL DE DISEÑO DE SISTEMAS - v1.0

EL ECOSISTEMA DE AGENTES: TRES MODELOS FUNDAMENTALES

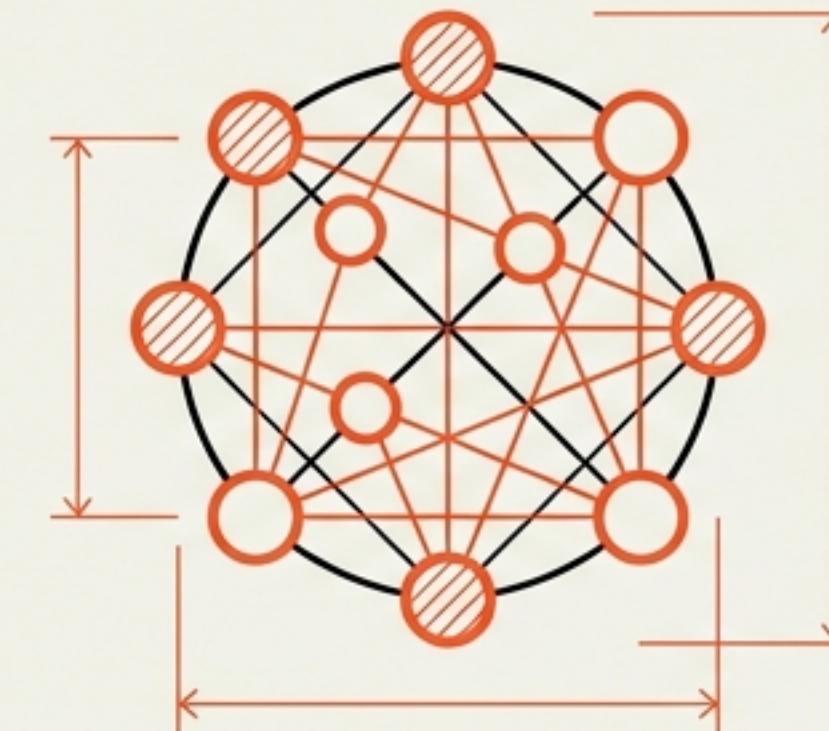
La inteligencia colectiva no es accidental; se diseña. Seleccione el patrón según la necesidad de control, decisión o procesamiento.



SUPERVISOR-WORKER

Orquestación Centralizada

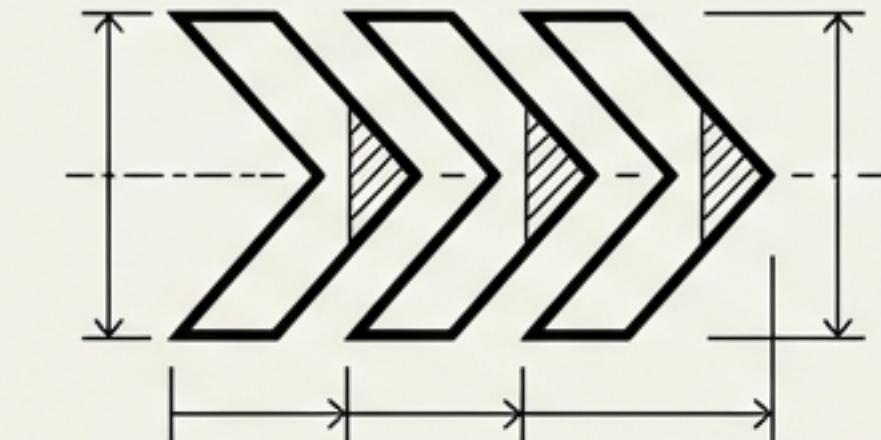
Uso: Control estricto y delegación de tareas.



DEBATE Y CONSENSO

Toma de Decisiones

Uso: Problemas complejos, múltiples perspectivas y validación.



PIPELINE

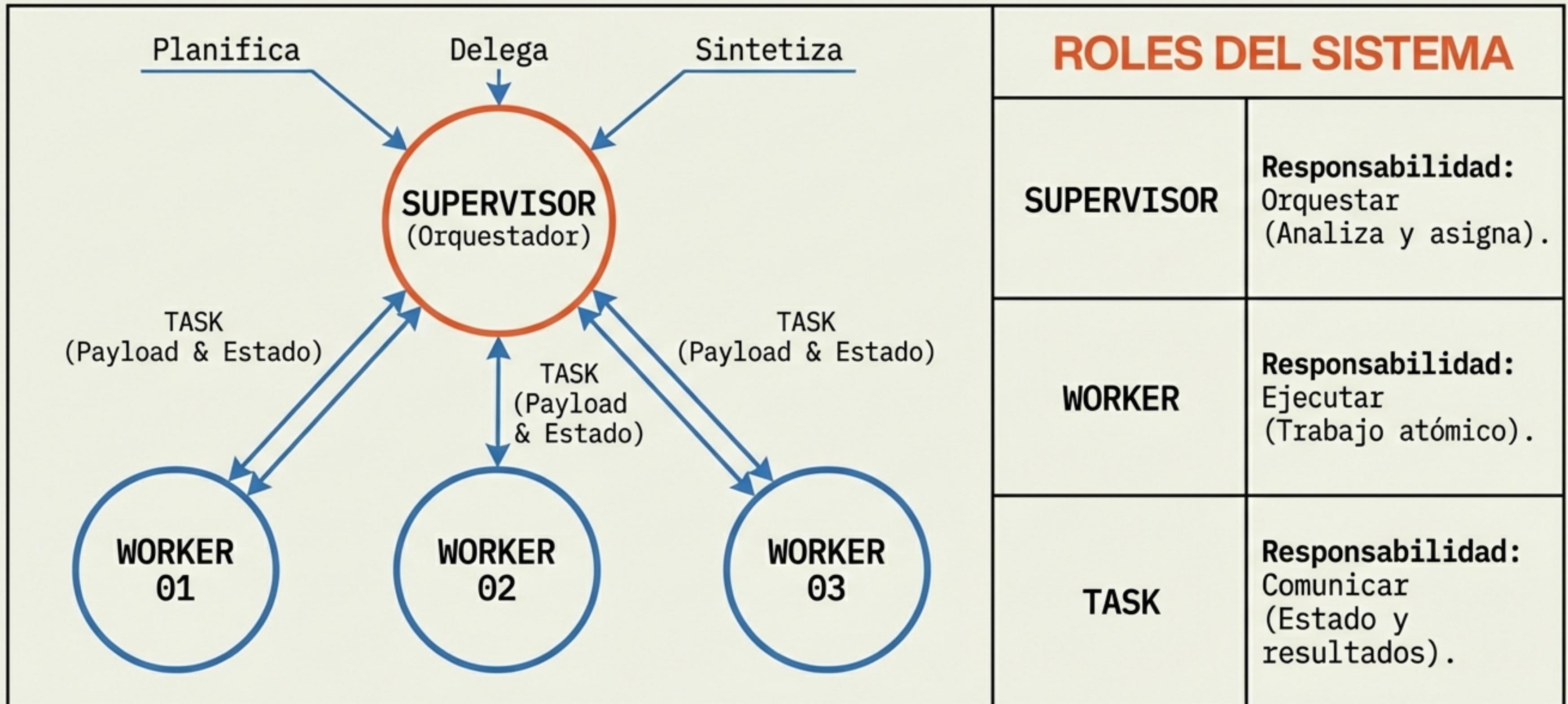
Procesamiento Secuencial

Uso: Transformación y enriquecimiento de datos paso a paso.

INSIGHT: La arquitectura define cómo fluye la información y dónde reside la 'verdad' del sistema.

ANATOMÍA DE LA JERARQUÍA: SUPERVISOR-WORKER

Centralización de la lógica de control y especialización de la ejecución.



MECÁNICAS DE CONTROL: DELEGACIÓN Y RESILIENCIA

ESTRATEGIAS DE DELEGACIÓN



Directa

Asignación estática por capacidad.



Dinámica

Asignación basada en disponibilidad.

CICLO DE RECUPERACIÓN



NOTA TÉCNICA: El Supervisor actúa como un “try/catch” arquitectónico, garantizando que el fallo de un nodo no colapse el sistema.

BLUEPRINT DE IMPLEMENTACIÓN: SUPERVISOR-WORKER

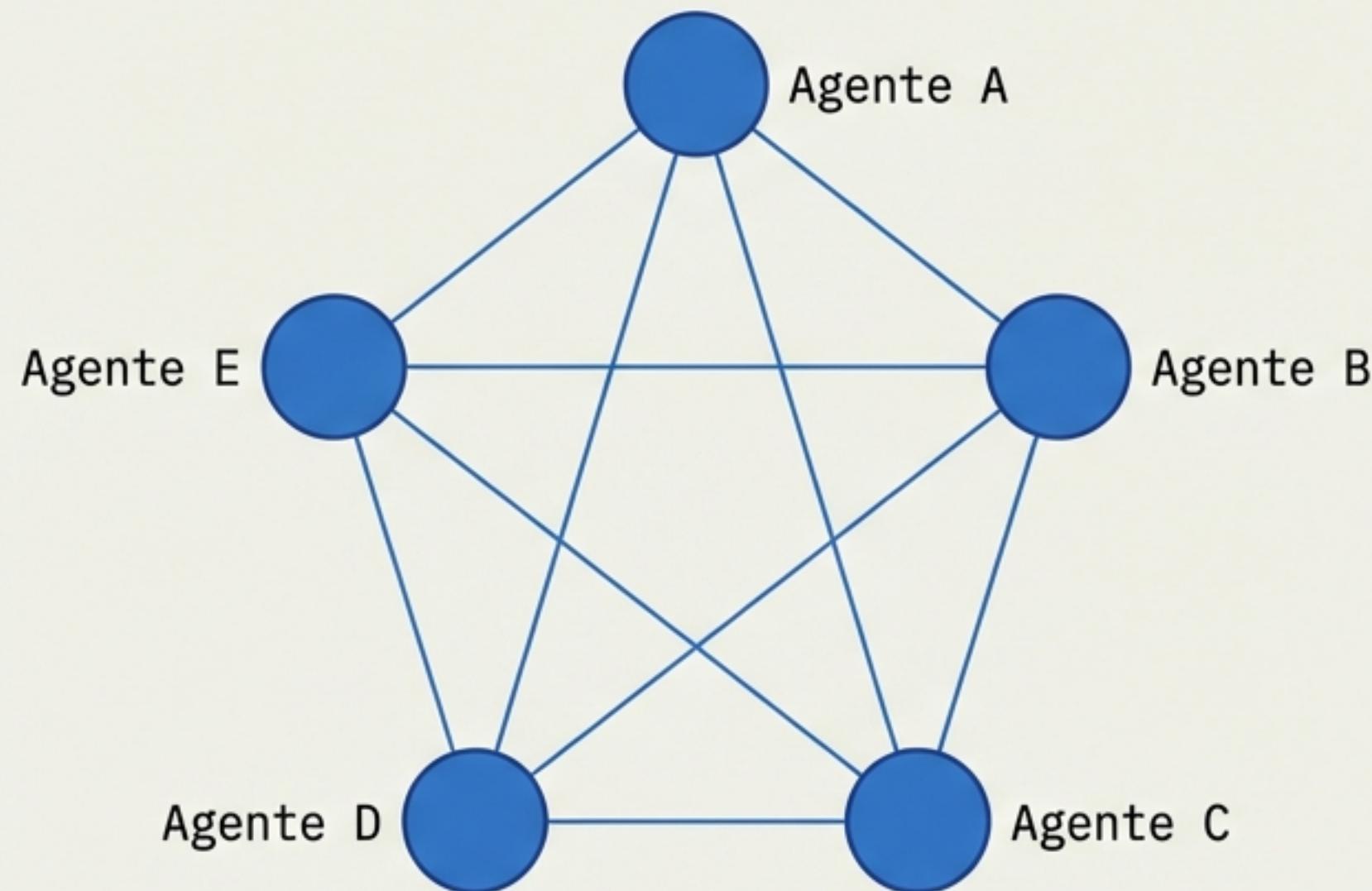
Lista de verificación crítica para el despliegue (Fuente 7.1.1).

- Definición de Capacidades:** ¿Tienen los Workers límites claros de responsabilidad?
 - Planificación de Tareas:** ¿Existe lógica para descomponer grandes objetivos en tareas atómicas?
 - Estrategia de Delegación:** ¿Directa o basada en carga dinámica?
 - Manejo de Errores:** ¿Existen "fallbacks" definidos si un Worker no responde?
 - Síntesis:** ¿Cómo se agregan los resultados parciales en una respuesta final?
 - Observabilidad:** Implementar logging y monitoreo centralizado.



ARQUITECTURA DE COLABORACIÓN: DEBATE Y CONSENSO

Mejorando la calidad de decisión a través de la fricción y la diversidad

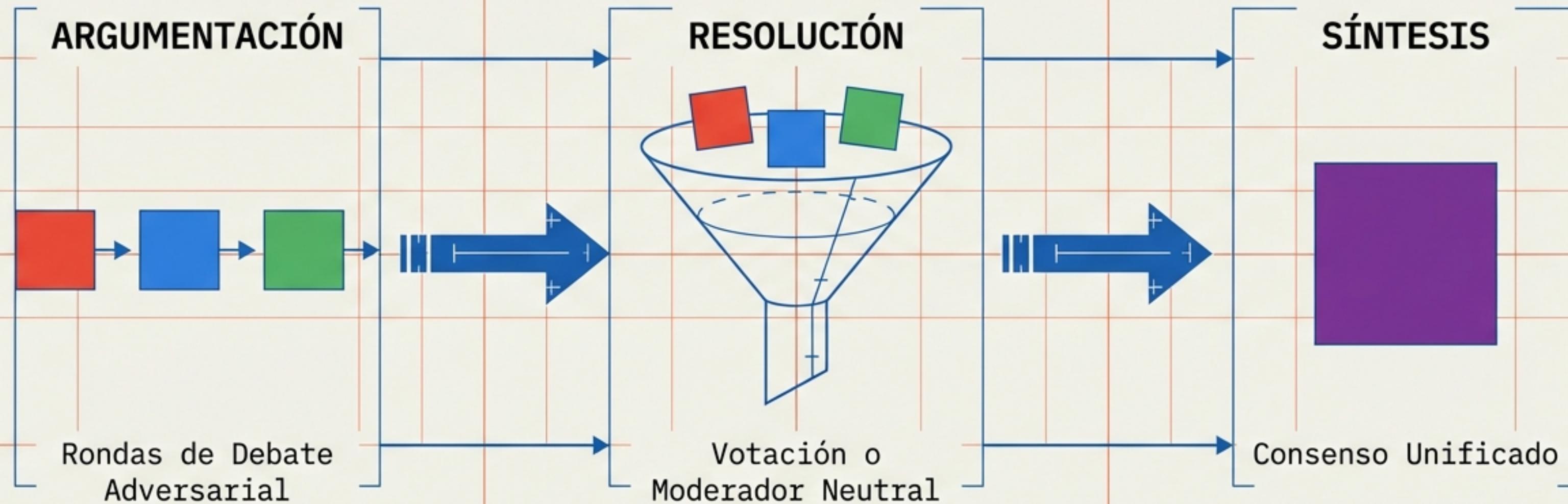


MECANISMOS DE RESOLUCIÓN

1	Debate Estructurado	Explorar perspectivas (Múltiples ángulos)
2	Votación	Decisión formal (Democrática/Binaria)
3	Adversarial	Refinar argumentos (Abogado del Diablo)
4	Consenso	Acuerdo colectivo (Alineación total)

NOTA TÉCNICA: La redundancia en la red de comunicación permite la convergencia hacia soluciones robustas mediante procesos iterativos.

FLUJOS DE DECISIÓN: DE LA ARGUMENTACIÓN AL ACUERDO



INSIGHT: El valor no está solo en la decisión final, sino en el historial del debate que documenta el “por qué”.

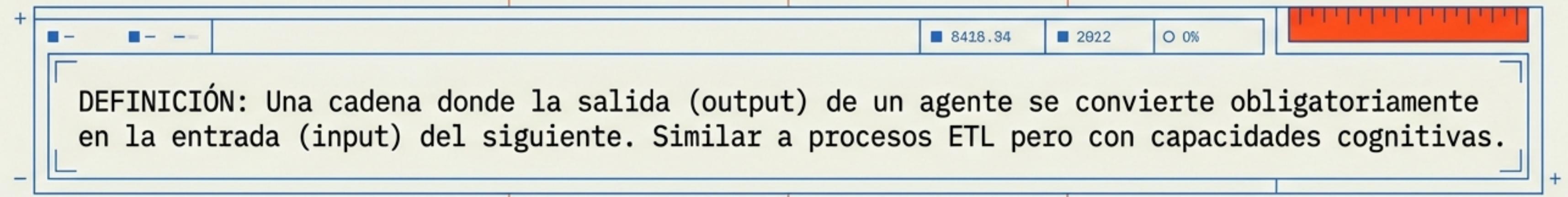
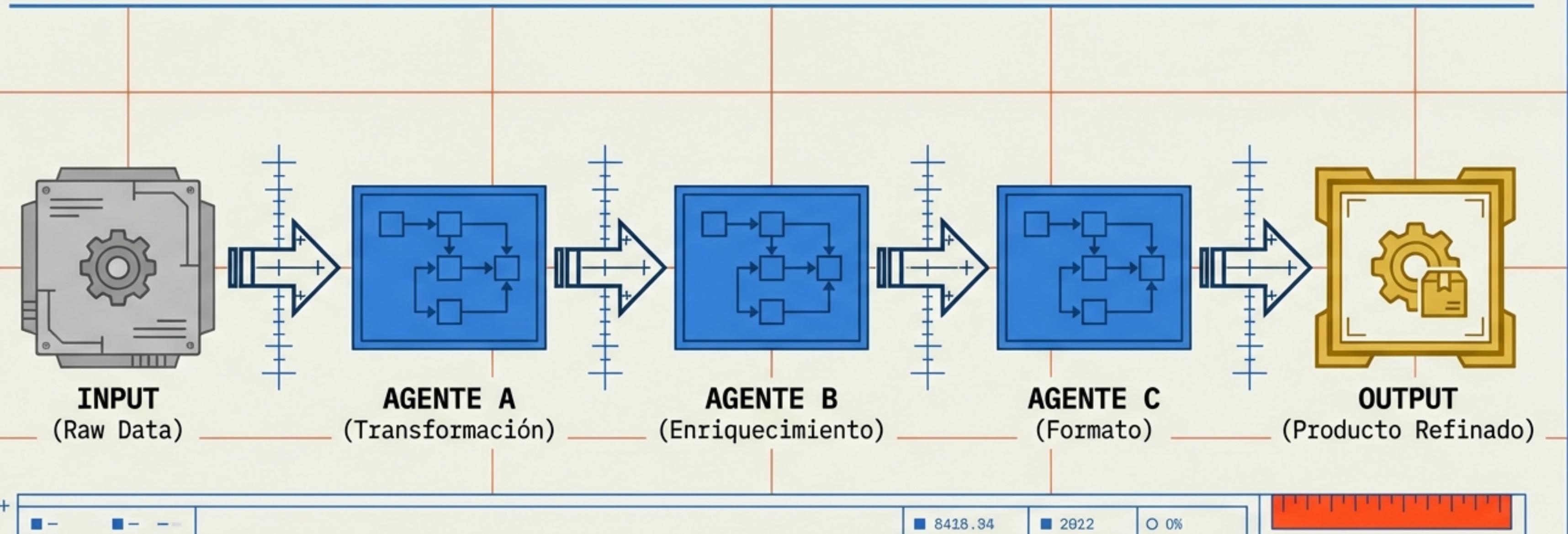
BLUEPRINT DE IMPLEMENTACIÓN: DEBATE Y CONSENSO

Lista de verificación crítica para el despliegue (Fuente 7.1.2).

- | | |
|-------------------------------------|--|
| <input checked="" type="checkbox"/> | Roles Claros: Definir perspectivas únicas para cada agente (ej. Crítico, Creativo). |
| <input checked="" type="checkbox"/> | Estructura de Rondas: Limitar la cantidad de intercambios para evitar bucles infinitos. |
| <input checked="" type="checkbox"/> | Moderación Neutral: Un agente debe actuar como árbitro del proceso, no del contenido. |
| <input checked="" type="checkbox"/> | Mecanismo de Votación: Definir umbrales de mayoría o unanimidad. |
| <input checked="" type="checkbox"/> | Síntesis de Conclusiones: Capacidad de resumir el acuerdo final. |
| <input checked="" type="checkbox"/> | Traza de Auditoría: Documentar todo el proceso de decisión (Logs de debate). |

LA LÍNEA DE ENSAMBLAJE: PIPELINE DE AGENTES

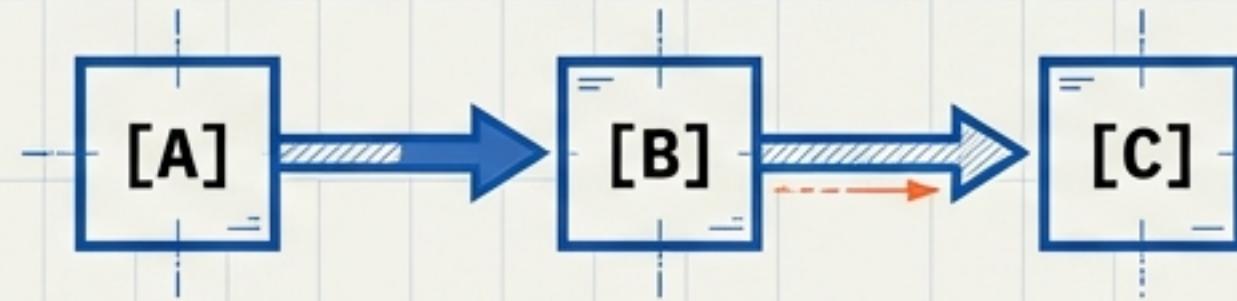
Procesamiento secuencial para el enriquecimiento de datos.



TOPOLOGÍAS AVANZADAS DE PIPELINE

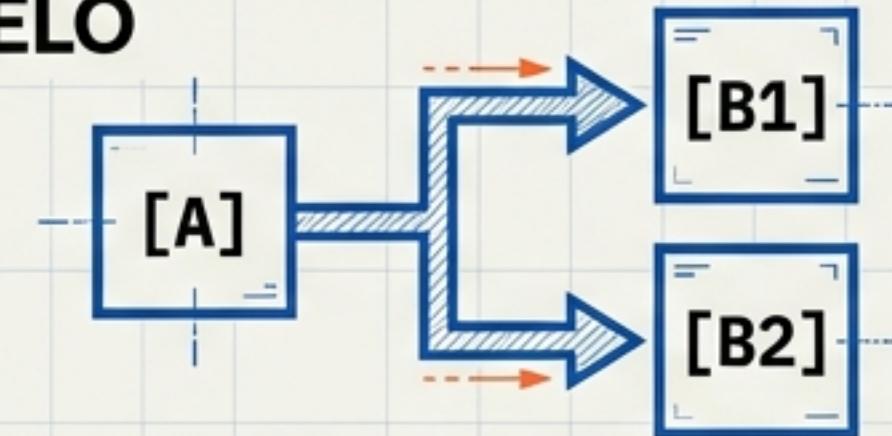
Exploración de arquitecturas complejas para flujos de trabajo distribuidos.

SECUENCIAL



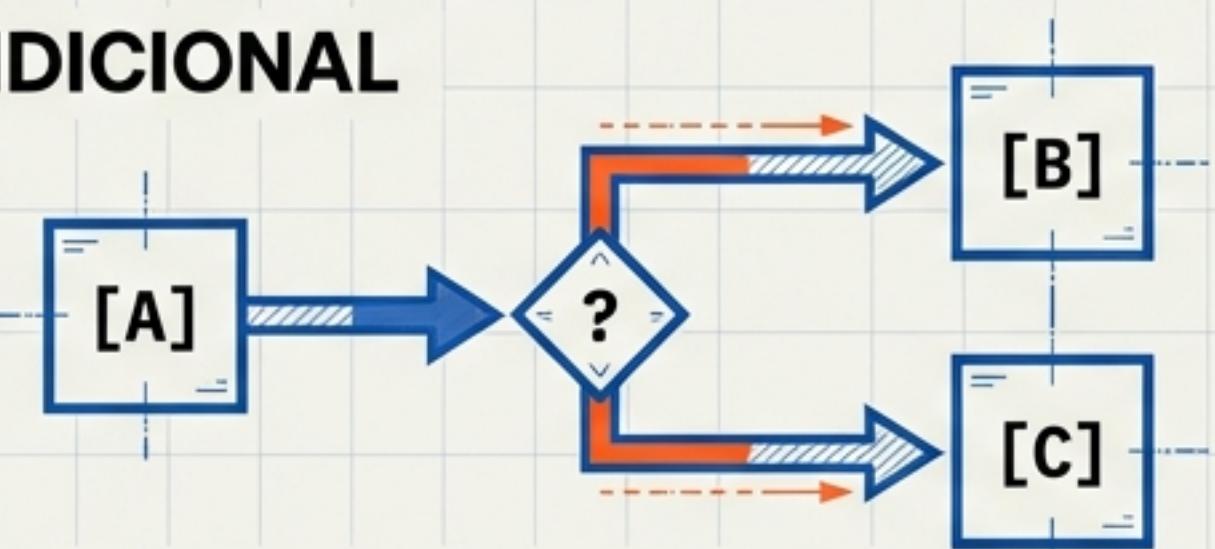
Simple y predecible. Transformación paso a paso.

PARALELO



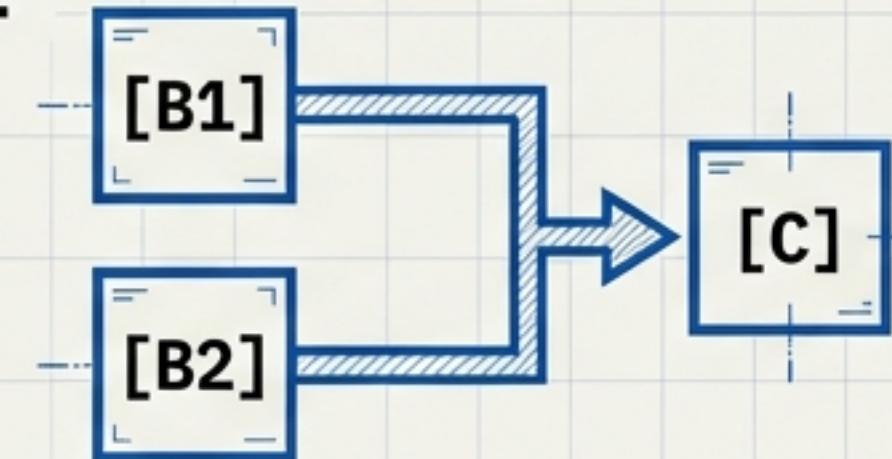
Velocidad y procesamiento independiente.

CONDICIONAL



Flexibilidad y rutas alternativas.

MERGE



Síntesis y combinación de resultados.

CRÍTICO: El “Paso de Contexto” es el pegamento que mantiene unida la topología; sin validación entre stages, el pipeline se rompe.

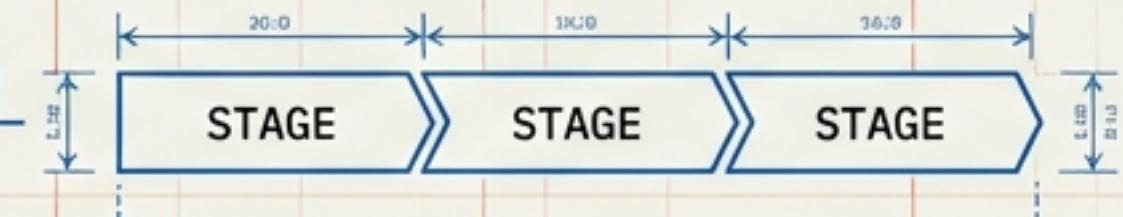
BLUEPRINT DE IMPLEMENTACIÓN: PIPELINE

Lista de verificación crítica para el despliegue (Fuente 7.1.3).



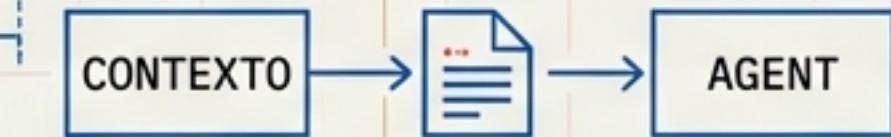
Definición de Stages

Responsabilidades únicas y claras por etapa.



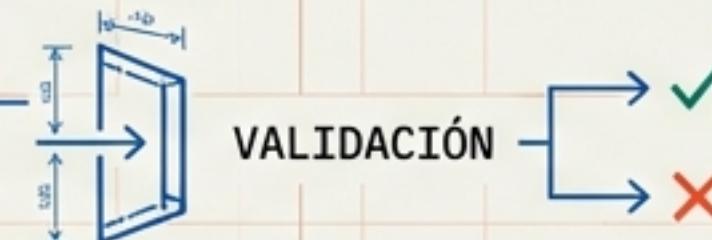
Paso de Contexto

Protocolo estandarizado para transferir data entre agentes.



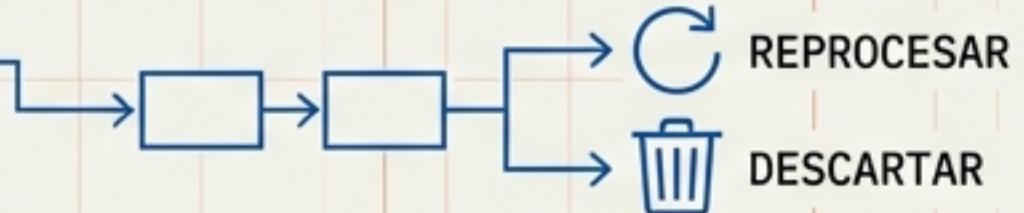
Validación de Gateways

Verificar la calidad de la data antes de entrar al siguiente stage.



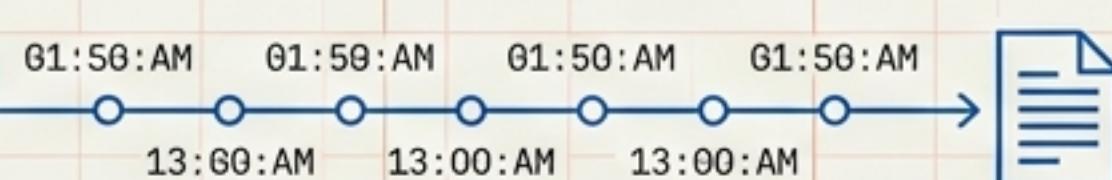
Manejo de Errores

Estrategia para 'reprocesar' o descartar items defectuosos.



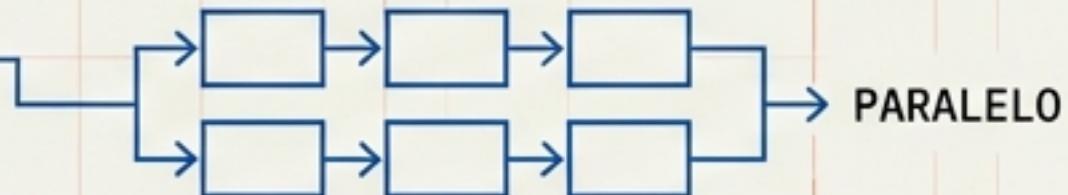
Historial

Logging del recorrido del dato a través del tubo.



Concurrencia

Ejecución paralela donde la dependencia secuencial no sea estricta.

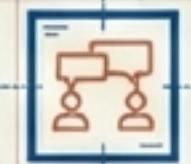


MATRIZ COMPARATIVA DE ARQUITECTURAS

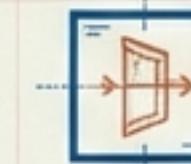
ARQUITECTURA	OBJETIVO	VENTAJA PRINCIPAL	COMPLEJIDAD	CASO DE USO
SUPERVISOR-WORKER	Orquestación	Control centralizado	Media	Delegación masiva
DEBATE Y CONSENSO	Calidad de Decisión	Diversidad / Menos Alucinaciones	Alta	Problemas ambiguos
PIPELINE	Transformación	Eficiencia modular	Baja/Media	Procesamiento de documentos



SUPERVISOR-WORKER



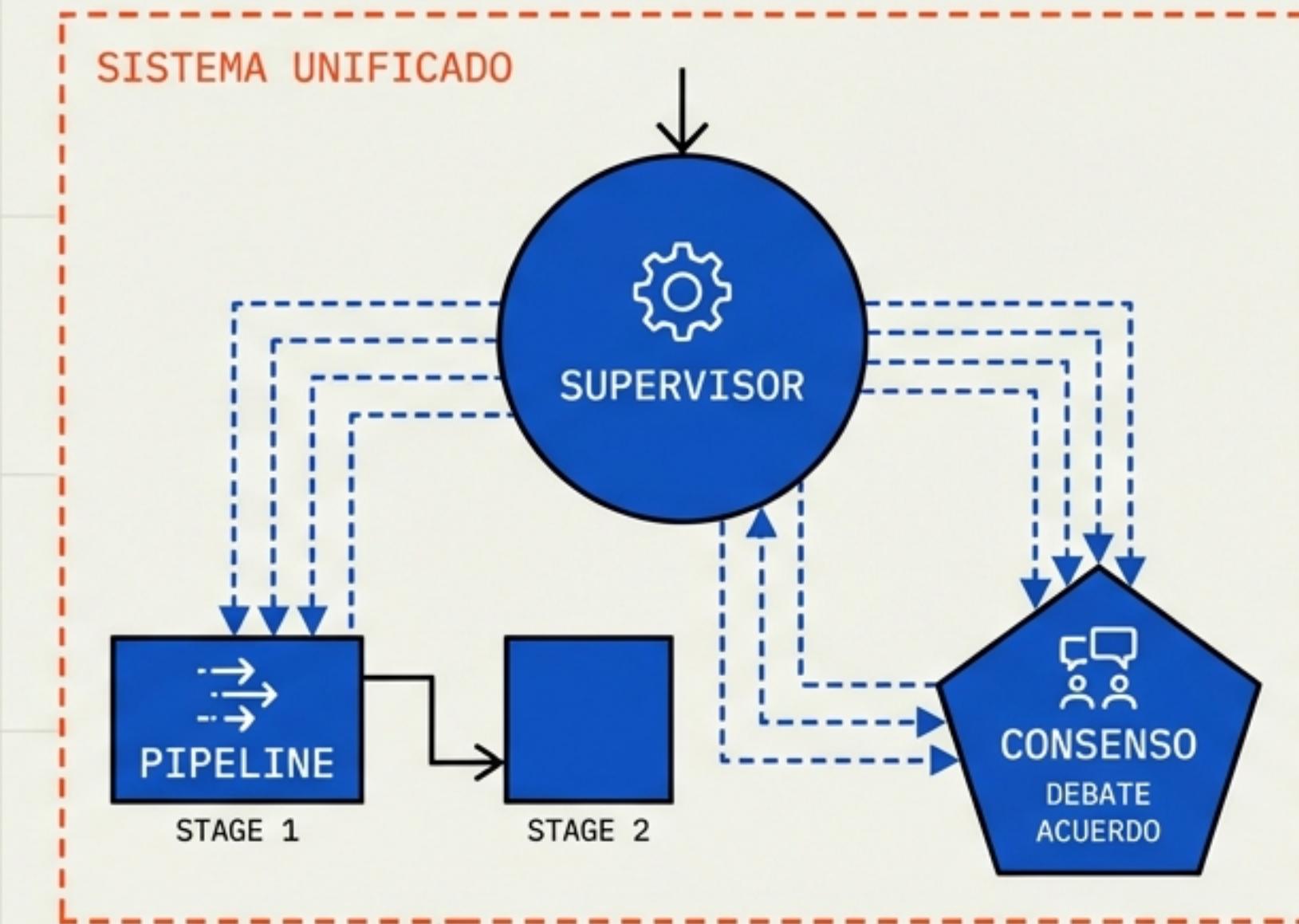
DEBATE Y CONSENSO



PIPELINE

CONCLUSIÓN Y SIGUIENTES PASOS

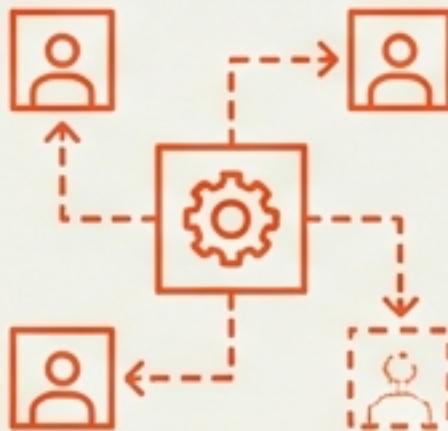
Ninguna arquitectura es estática. Los sistemas más robustos a menudo combinan estos patrones (ej. un Supervisor que gestiona un Pipeline).



PRÓXIMO TEMA: COMUNICACIÓN INTER-AGENTE

Formatos de mensaje, memoria compartida y manejo de conflictos.

RESUMEN EJECUTIVO (TAKEAWAYS)



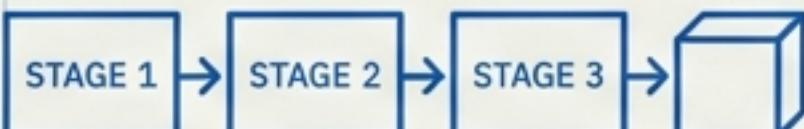
01

El patrón **SUPERVISOR** asegura robustez mediante el aislamiento de fallos.



02

El modelo de **CONSENSO** sacrifica velocidad por calidad de decisión.



03

El **PIPELINE** impone orden y permite la trazabilidad de datos complejos.