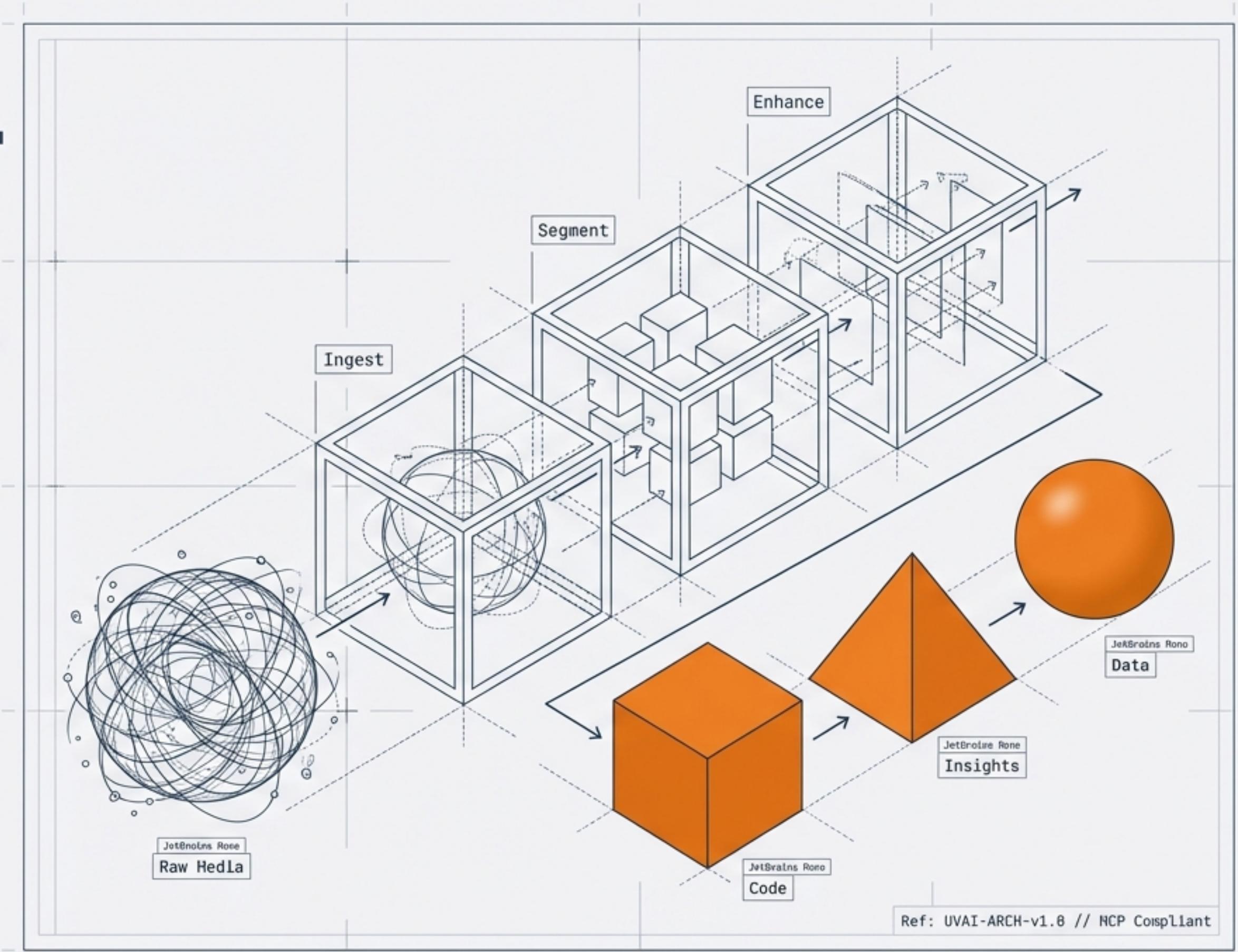


UVAI: THE BLUEPRINT

Transforming Raw Video into Executable Intelligence.

A comprehensive technical architecture for a cloud-native “Digital Refinery.”

Transitioning from local prototyping to Enterprise SaaS on Google Vertex AI and Gemini.

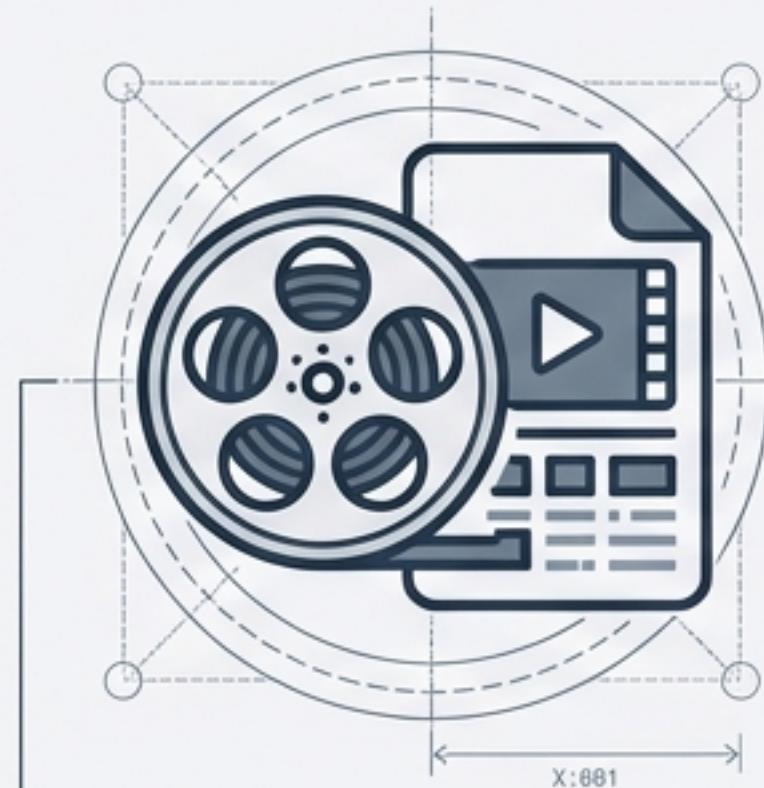


THE MISSION: REFINE OPAQUE MEDIA

Structuring the unstructured through a four-stage workflow.



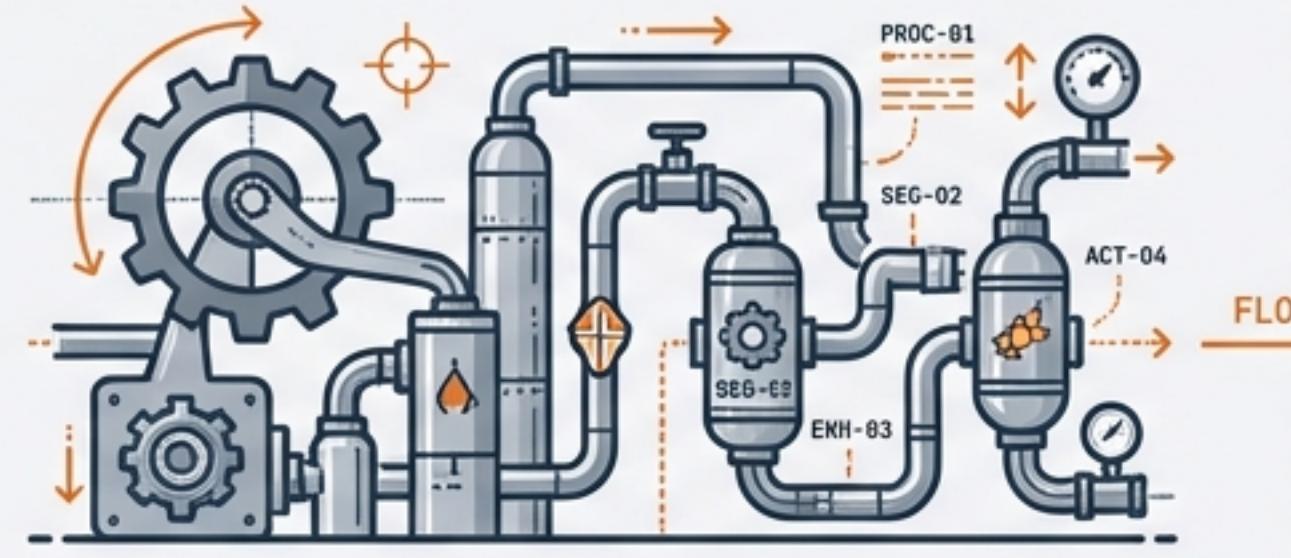
Input: Opaque Data



Video content is heavy, unsearchable, and passive. It traps intelligence inside a linear timeline.

DATA SOURCE: RAW VIDEO

The Digital Refinery



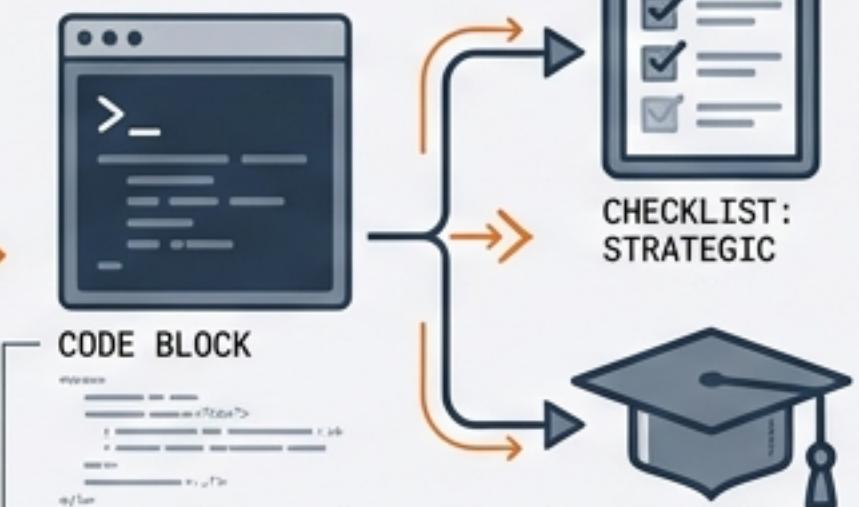
1. Ingest
2. Segment
3. Enhance
4. Action

A specialized pipeline (Ingest, Segment, Enhance, Action) built on Google Cloud AIP standards and Model Context Protocol (MCP).

ARCH: GCP AIP STANDARDS

PROTO: MCP INTEGRATED

Output: Actionable Assets



Distinct, lightweight assets: Executable Code, Strategic Action Items, and Interactive Learning modules.

ASSET TYPES: EXEC, STRAT, INTERACTIVE

USER EXPERIENCE: THE DASHBOARD

A friction-free interface for submission and real-time tracking.

The dashboard interface is designed for effortless submission and real-time tracking. It features a dark header bar with a 'URLInputForm' for pasting video URLs and a 'USER PROFILE' section. Below the header is a 'RECENT SUBMISSIONS' card for 'Quarterly All-Hands.mp4', showing processing status and action buttons.

URLInputForm: Effortless submission entry point.

USER PROFILE: User profile icon and link.

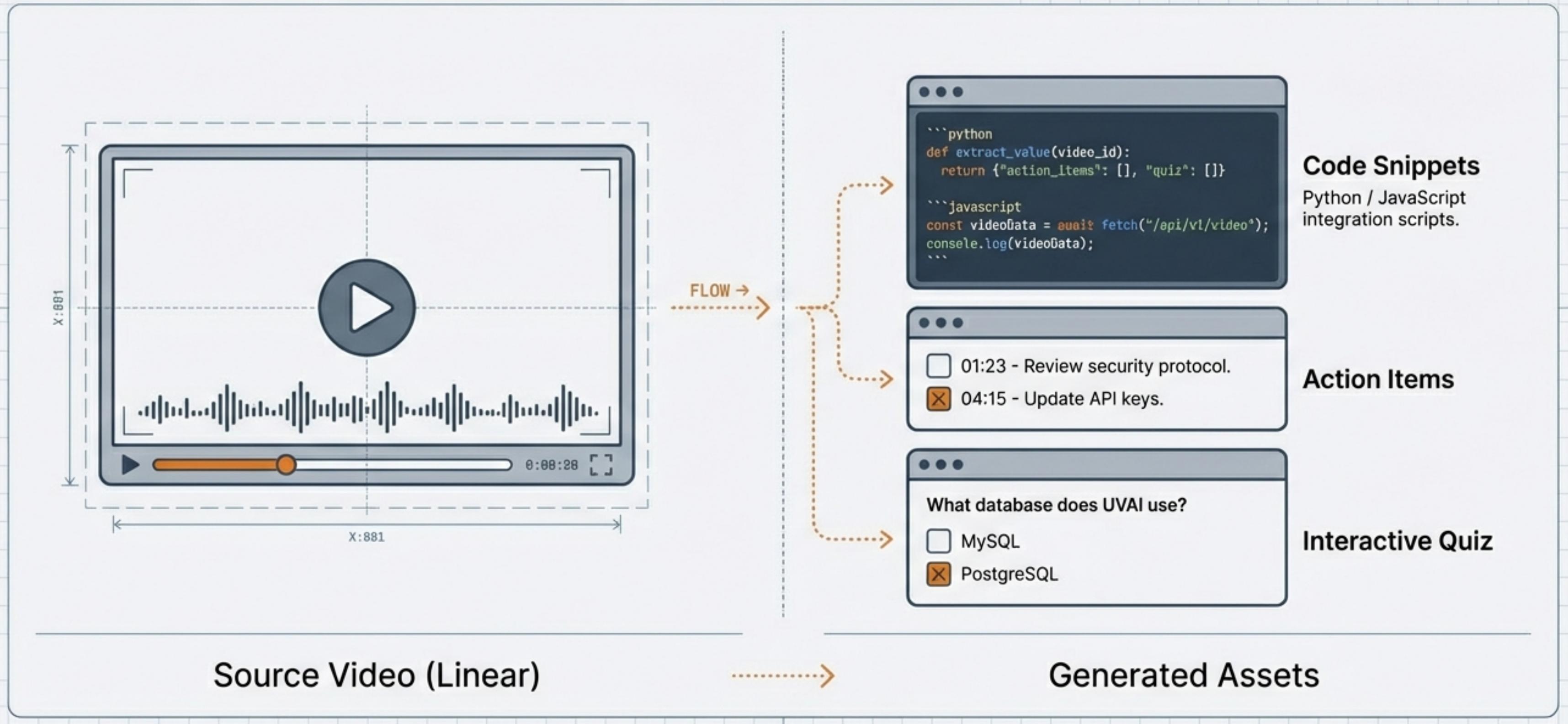
PROCESS: Orange button for processing.

ProcessingTimeline: Real-time state visualization.

VideoProcessingCard: Status tracking container.

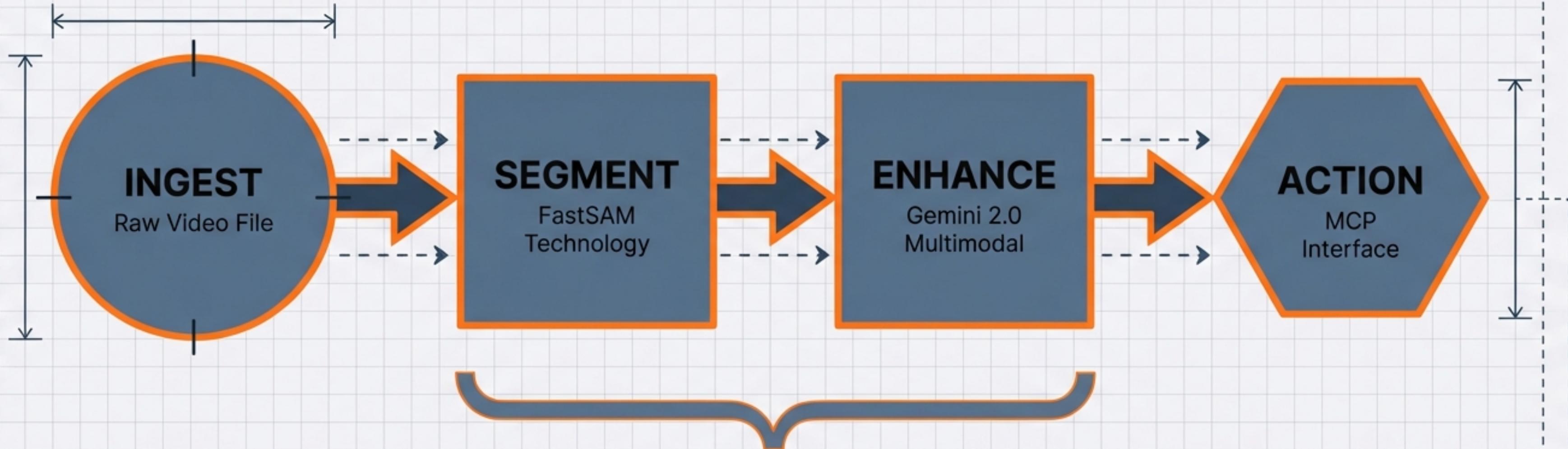
DELIVERING THREE PILLARS OF VALUE

From linear playback to interactive utility.



THE WORKFLOW ENGINE (ADK GRAPH)

Orchestrating the journey from pixel to byte.



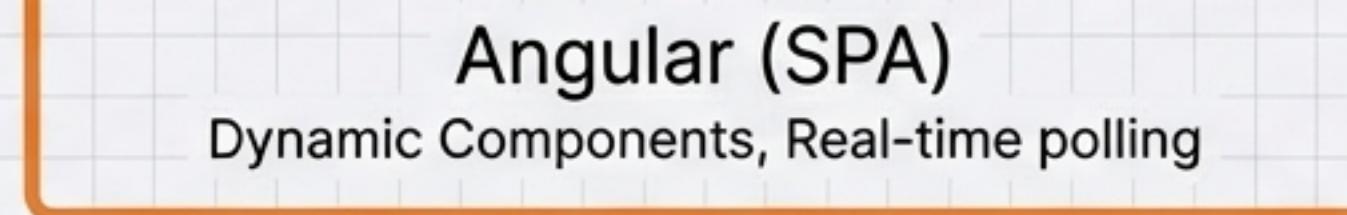
AI Processing Core

The Collaboration Hub
shares this blueprint
across the architecture.

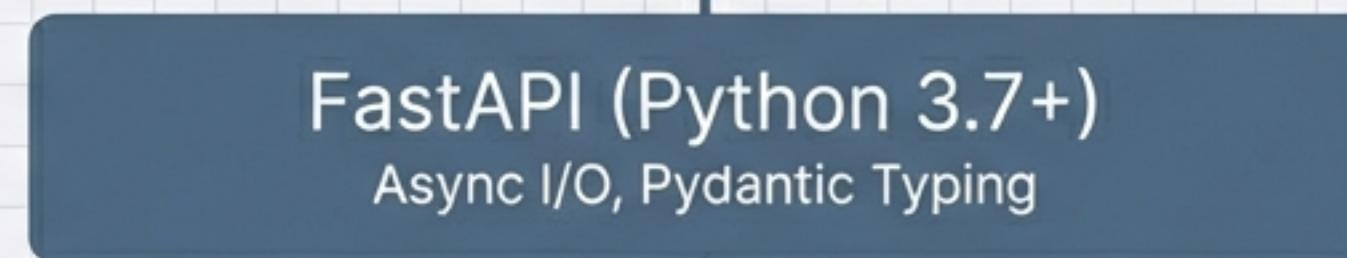
THE TECH STACK

A modern, type-safe build plan designed for scale.

Interface



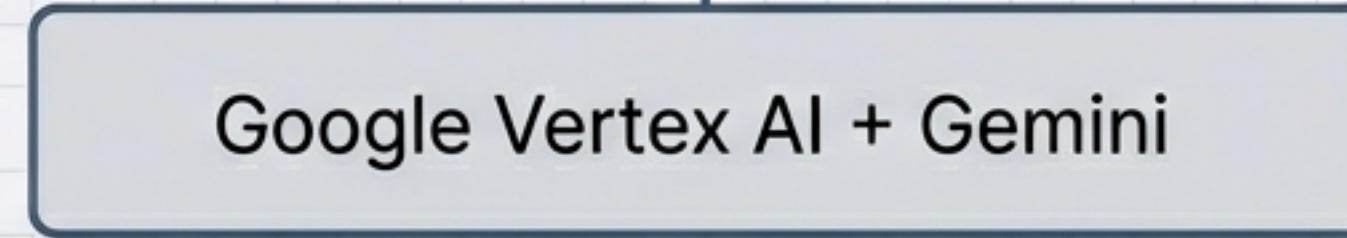
Orchestration



Persistence



Cloud & AI



Selected for
Type Safety &
Performance.

BACKEND ORCHESTRATION

```
# main.py - FastAPI Initialization
app = FastAPI(
    title="UVAI Backend API",
    description="Orchestrating video processing and user interactions.",
    version="0.1.0",
)

# CORS Configuration for Angular Frontend
origins = ["http://localhost:4200"]

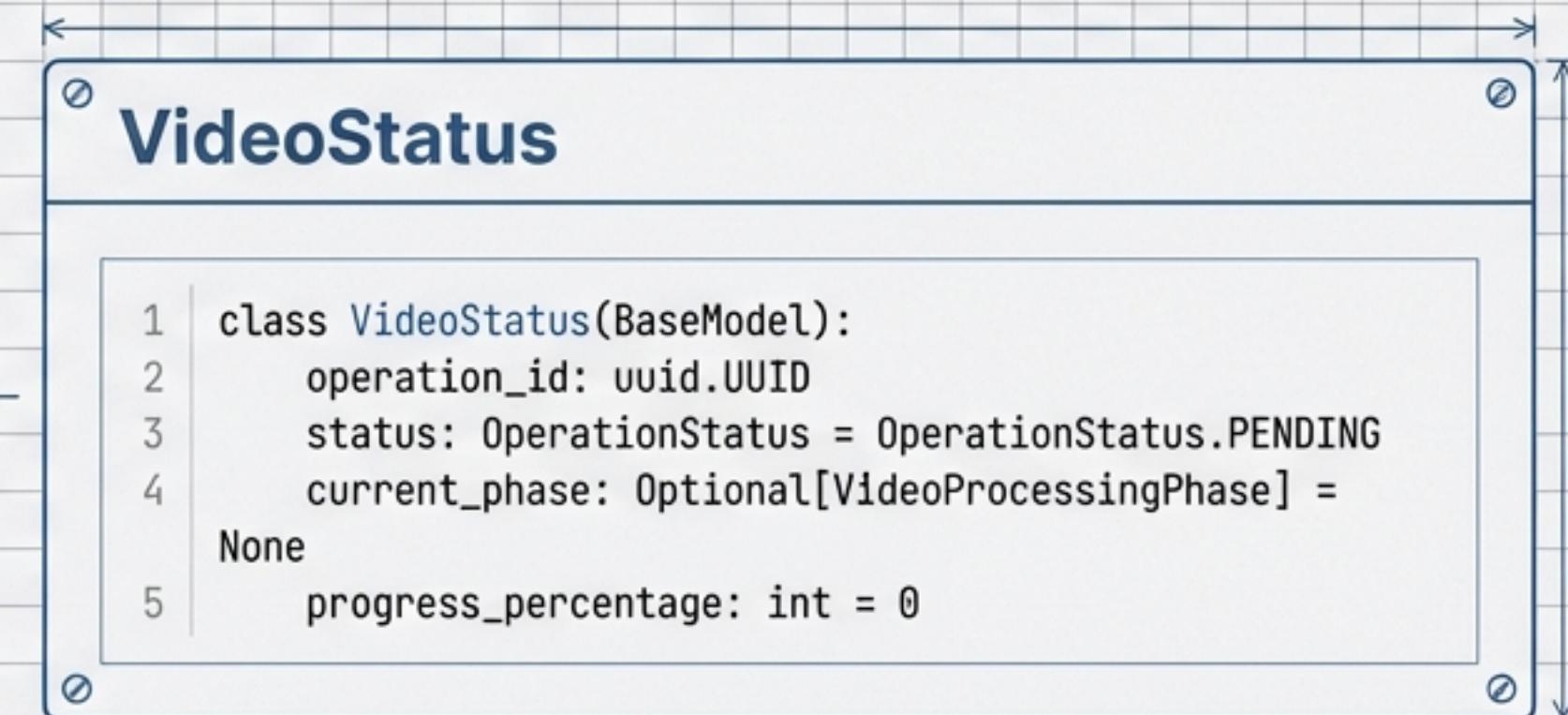
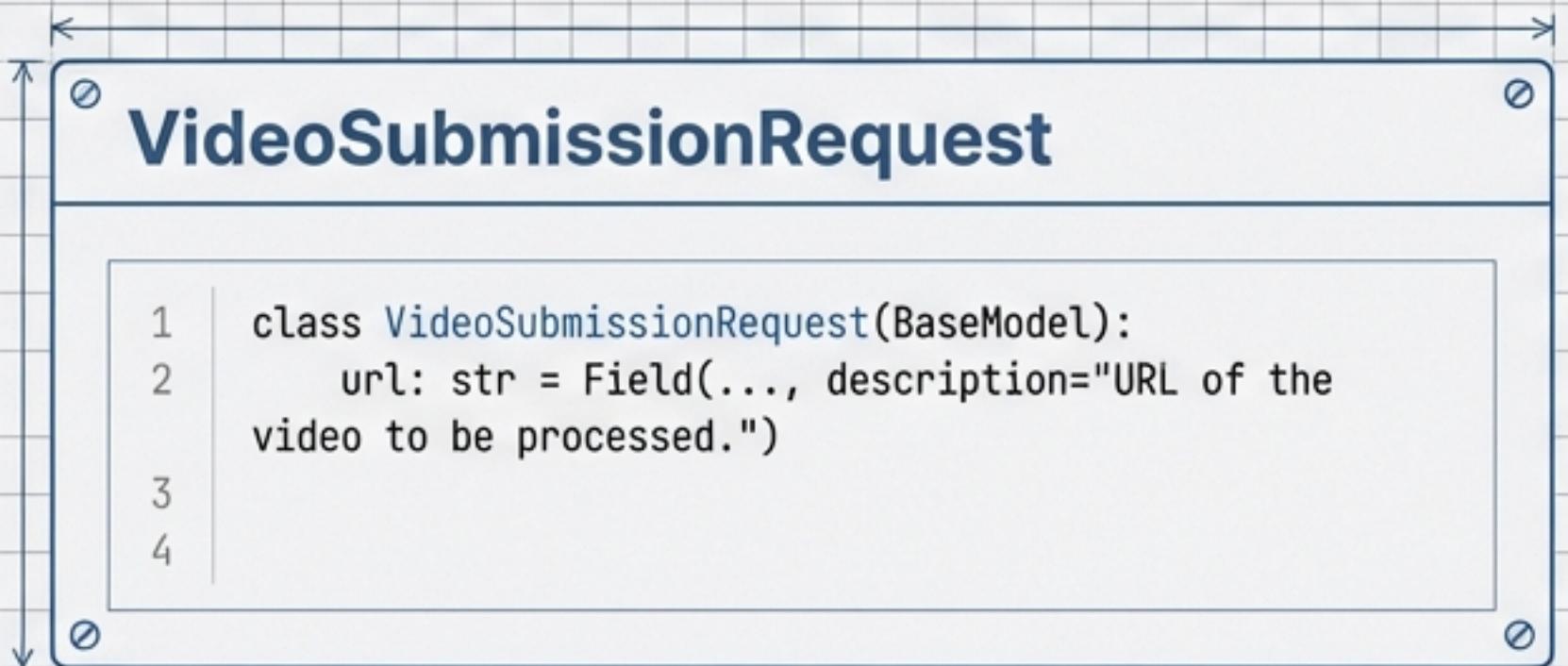
app.add_middleware(
    CORSMiddleware,
    allow_origins=origins,
    allow_credentials=True,
    allow_methods=["*"],
    allow_headers=["*"],
)

class VideoProcessingPhase(str, enum.Enum):
    INGEST = "Ingest"
    SEGMENT = "Segment"
    ENHANCE = "Enhance"
    ACTION = "Action"
```

Key Design Pattern: Using Enums to strictly define the 4-step workflow state.

DATA MODELING & INTEGRITY

Enforcing strict schema validation with Pydantic.



By defining `'operation_id'` as `UUID` and `'progress_percentage'` as `int`, the system eliminates runtime type errors before they happen.

API ENDPOINT STRATEGY

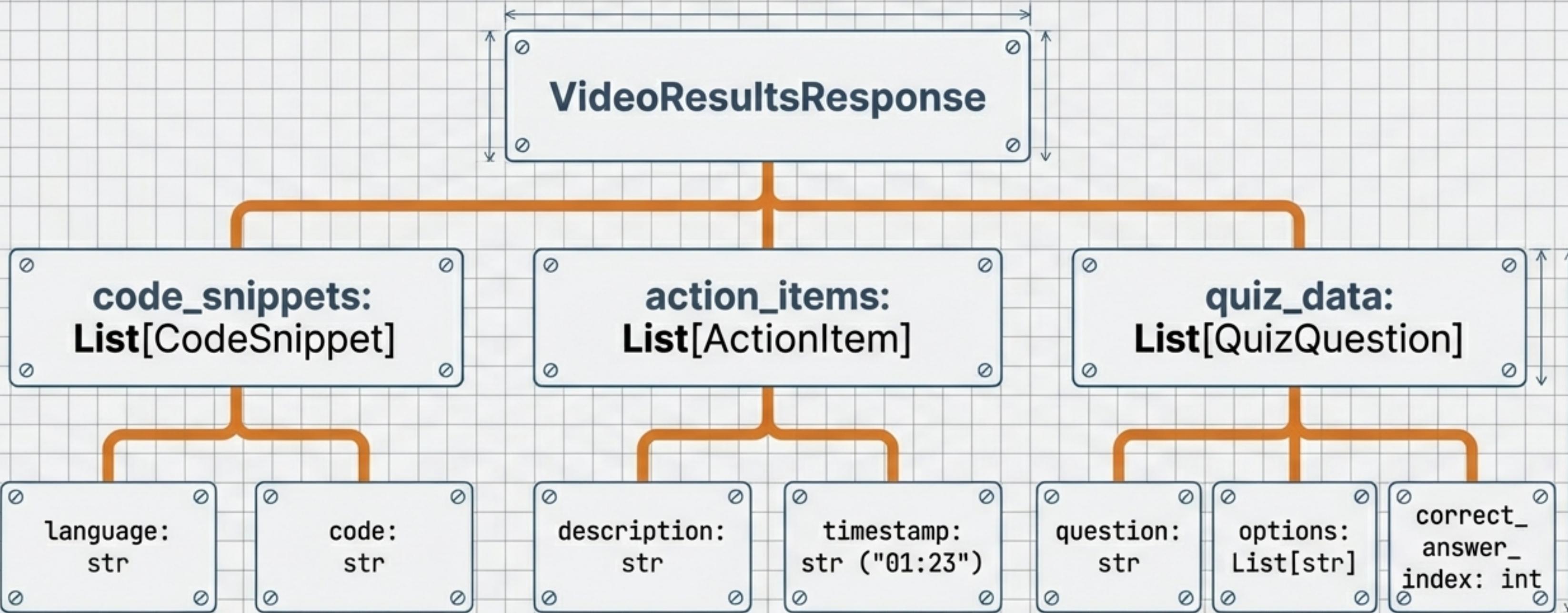
RESTful architecture designed for asynchronous polling.

METHOD	ENDPOINT	FUNCTION
POST	/videos	Initiate workflow. Returns `operation_id`.
GET	/videos/{id}/status	Poll for progress (Ingest -> Action). Real-time tracking.
GET	/videos/{id}/results	Retrieve final payload (Code, Actions, Quiz).
POST	/videos/{id}/quiz/submit	Submit user answers for server-side evaluation.



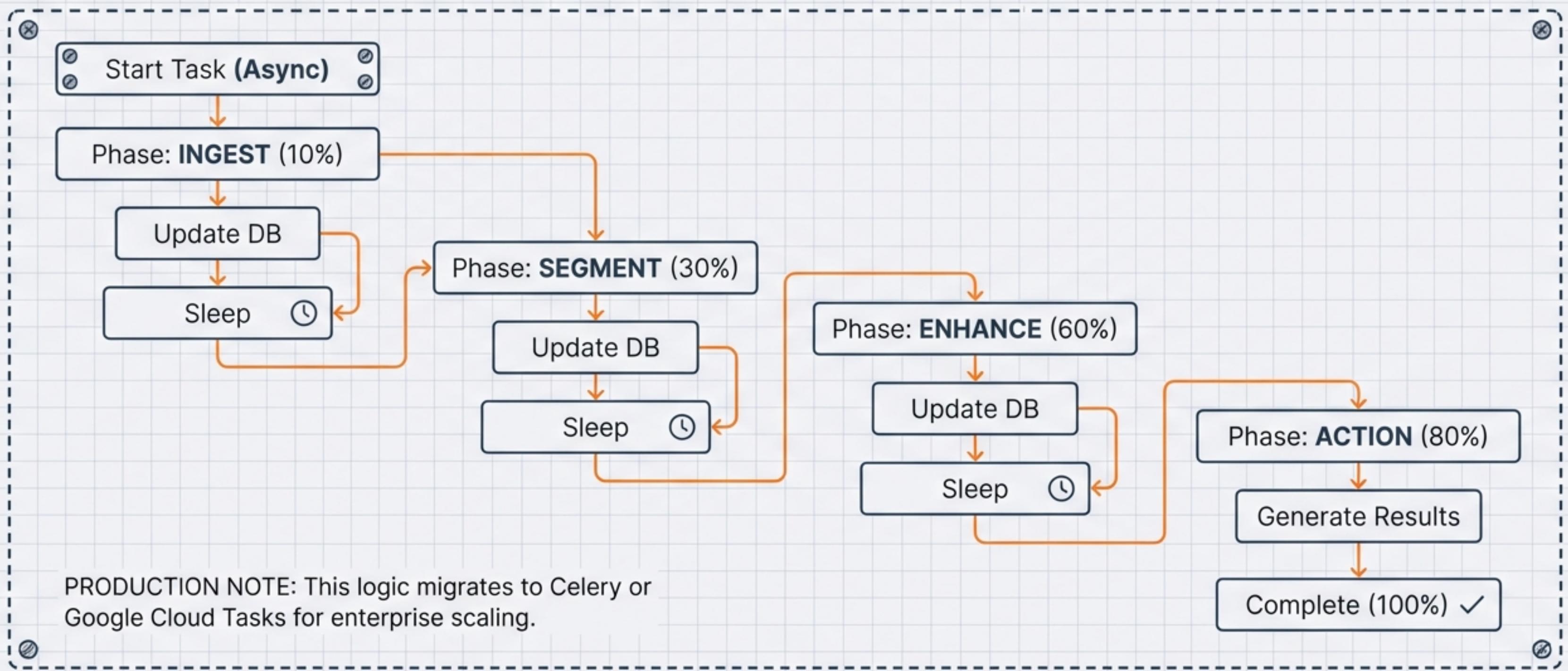
THE RESULTS PAYLOAD

Structuring AI output for frontend consumption.



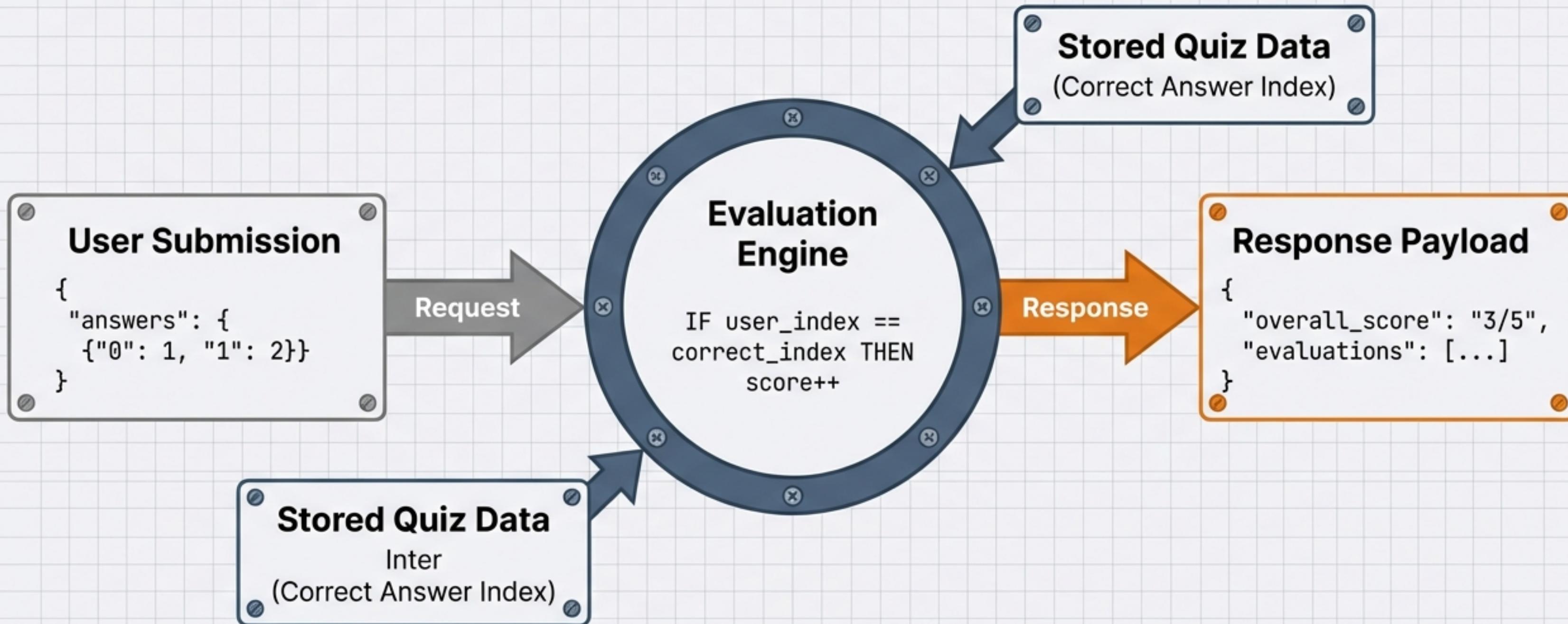
SIMULATING THE ASYNC LOOP

Handling long-running ‘Digital Refinery’ tasks.



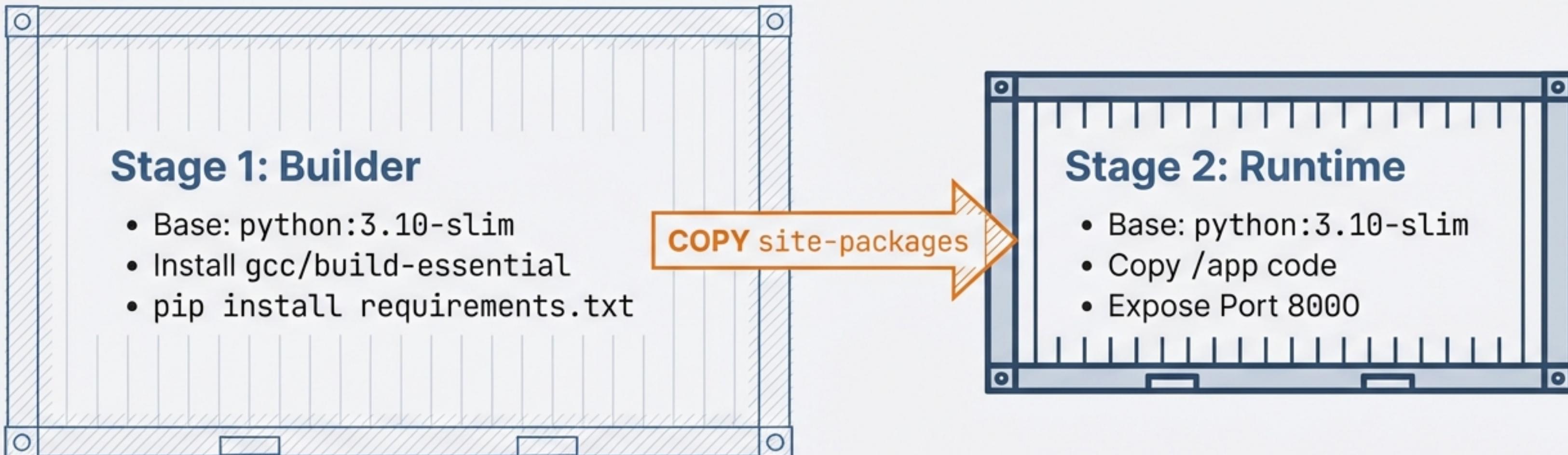
INTERACTIVE LOGIC: QUIZ ENGINE

Endpoint: POST /videos/{id}/quiz/submit



DEPLOYMENT: CONTAINERIZATION

Optimized Multi-Stage Docker Build.



```
> CMD ["gunicorn", "main:app", "--workers", "4", "--worker-class", "uvicorn.workers.UvicornWorker"]
```

FROM BLUEPRINT TO ENTERPRISE SCALE UVAI

