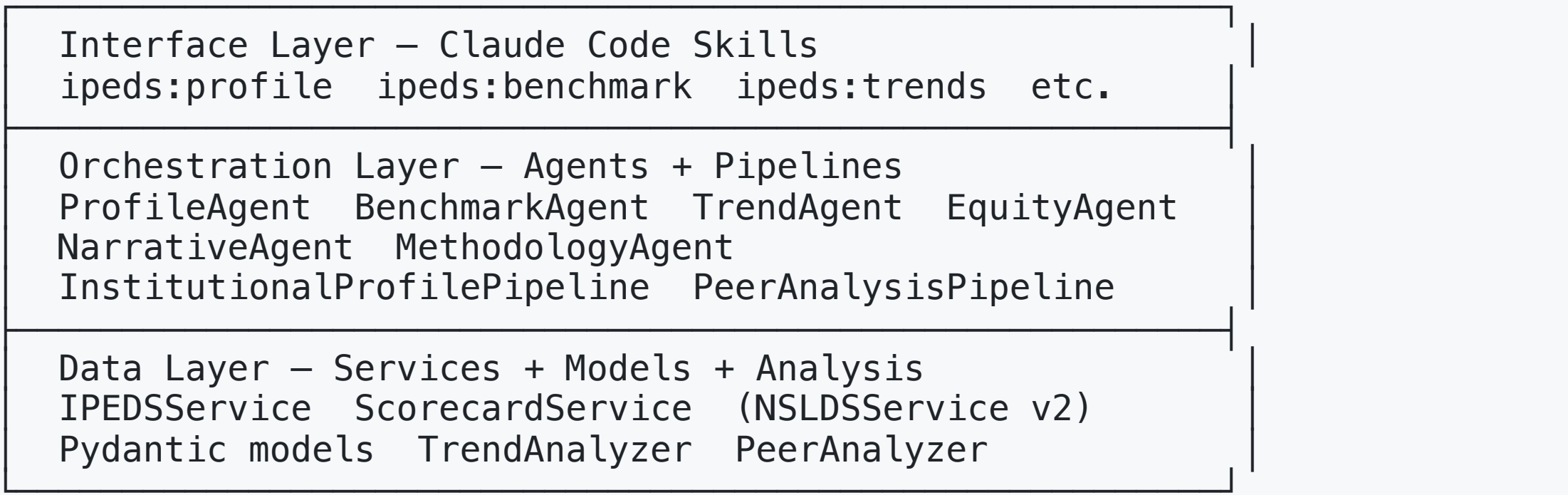
IPEDSBrain

Technical Architecture & Delivery Plan

John Dimatos — February 2026

System Architecture

Three layers with strict separation of concerns:



Key constraint: Services never call LLMs. Agents optionally do. All data processing is deterministic and auditable regardless of the AI layer.

Data Layer

Services (one per data source)

Service	Source	Data Type	Priority
`IPEDSService`	Bundled CSVs	All survey components, 2019-2023	v1
`ScorecardService`	College Scorecard API	Earnings, debt, completion by income	v1
`NSLDSService`	NSLDS flat files	Loan repayment, default rates	v2
`CDSService`	Common Data Set	Self-reported institutional data	v2
		State-specific student	

Data Layer: Analysis Utilities

Pure functions — no LLM, no side effects, fully deterministic:

Utility	What It Computes
`TrendAnalyzer`	Slopes, inflection points, anomaly detection across years
`PeerAnalyzer`	Percentile ranks, z-scores, outlier identification within groups
`EquityAnalyzer`	Gap analysis across demographics (graduation, retention, Pell)
`ClusterAnalyzer`	Group similar institutions by configurable dimensions

These produce the raw analytical results. Agents add interpretation and narrative.

Orchestration Layer: Agents

Each agent has a single analytical lens and a defined autonomy level:

Agent	Role	Autonomy
`ProfileAgent`	Synthesize institutional portrait from all sources	Show Your Work
`BenchmarkAgent`	Compare institution against peers, identify gaps	Show Your Work
`TrendAgent`	Narrate changes over time, flag inflection points	Show Your Work
`EquityAgent`	Analyze access and outcome gaps across demographics	Show Your Work
`NarrativeAgent`	Turn structured findings into polished prose	Constrained

Orchestration Layer: Pipelines

Pipelines chain agents into multi-step workflows:

InstitutionalProfilePipeline

1. Fetch all data for institution → `IPEDSService`, `ScorecardService`
2. Compute 5-year trends → `TrendAnalyzer`
3. Generate profile narrative → `ProfileAgent`
4. Document methodology → `MethodologyAgent`
5. Output: report + script + data bundle

PeerAnalysisPipeline

1. Load peer group → `Portfolio`
2. Compute percentile ranks → `PeerAnalyzer`

Citation & Validation System

Every LLM-generated claim is validated before inclusion:

```python
class DataCitation(BaseModel):
    claim: str                # "Graduation rate fell 8 points"
    source: str               # "IPEDS.DRVGR2023"
    field: str                # "BAGR150"
    institution: int          # UNITID
    year: int                 # 2023
    claimed_value: float      # -8.0
    actual_value: float       # System-computed
    matches: bool             # Validated automatically
    comparison_base: str      # "DRVGR2019.BAGR150"
```

The pipeline validates every citation against actual data before passing results to the narrative layer. Mismatches are flagged, not silently included.

Output `citations.json` allows third-party verification of every claim in the report.

Reproducibility System

Every analysis produces a self-contained, verifiable bundle:

```
analysis-2026-03-15.zip
├── README.md              # Instructions for third-party verification
├── report.md              # Narrative report with inline citations
├── analysis.py            # Standalone script — pandas only, zero deps
├── data/                  # Exact CSV slices (not full IPEDS dump)
│   ├── institutions.csv
│   └── programs.csv
├── methodology.md         # Cohort defs, filters, peer selection, methods
└── citations.json         # Every claim → source → actual value
```

`analysis.py` properties:

- **Zero framework dependency** — imports only pandas
- **Self-contained data** — reads from bundled `data/` directory
- **Deterministic** — same input → same output, every time

Project Structure

```
ipeds-brain/
├── .claude/
│   ├── settings.json
│   └── skills/ipeds/                  # 12-15 skill definitions (.md)
├── src/ipeds/
│   ├── models/                        # Pydantic schemas (8 modules)
│   ├── services/                      # Data source adapters (5 modules)
│   ├── analysis/                      # Pure function analyzers (4 modules)
│   ├── agents/                        # LLM-powered actors (6 modules)
│   ├── pipelines/                     # Multi-step orchestrators (4 modules)
│   ├── outputs/                       # Report, dashboard, slides generators
│   └── config.py
├── data/ipeds/                        # Bundled IPEDS CSVs by year
├── portfolios/                        # Saved institution groups (.yaml)
├── outputs/                           # Generated artifacts
├── templates/                         # Report, deck, dashboard templates
├── tests/
├── CLAUDE.md
└── pyproject.toml
```

Infrastructure Requirements

Requirement	Detail
Runtime	Claude Code (Anthropic) — terminal-based AI assistant
Language	Python 3.10+
Dependencies	pandas, pydantic, httpx (for Scorecard API)
Visualization	Observable Framework + Node.js 18+
Presentations	Marp (markdown → PDF/PPTX)
Reports	Markdown → PDF via pandoc
Storage	~500MB for 5 years of bundled IPEDS data
Network	Optional — only for College Scorecard API
Servers	None

Security & Data Governance

Data classification

All data sources are **public federal data** — no PII, no FERPA concerns, no restricted data. IPEDS is published by NCES for public use. College Scorecard is a public API.

AI interaction boundary

- Data layer is **fully deterministic** — no LLM involvement in data processing
- LLM is used only for narrative interpretation and methodology documentation
- Every LLM-generated claim is validated against actual data
- Reproducible scripts recreate all numbers without any AI

Outputs

- Generated reports, bundles, and dashboards contain only public data

Delivery: Phase 1 (Weeks 1-4)

Core Engine + First Report

Deliverable	Detail
IPEDSBrain installation	Configured for your institution
IPEDS data bundle	2019-2023, all survey components
Scorecard integration	Earnings, debt, completion by income
`ipeds:profile`	Full institutional profile generation
`ipeds:benchmark`	Peer comparison across all metrics
Reproducibility system	Every analysis → verifiable bundle
Custom portfolios	3-5 peer groups built with IR team

Delivery: Phase 2 (Weeks 4-7)

Trends, Equity + Dashboards

Deliverable	Detail
`ipeds:trends`	Multi-year trend analysis with anomaly detection
`ipeds:equity`	Outcome gap analysis across demographics
Dashboard generator	Standalone Observable Framework dashboards
Second report	Trend analysis + equity audit + standing dashboard

Accreditation relevance

Most regional accreditors require:

- Evidence of outcomes tracking over time → `ipeds:trends`

Delivery: Phase 3 (Weeks 7-9)

Output Templates + Training

Deliverable	Detail
Report templates	Customized to institutional branding (MD + PDF)
Slide deck generator	Board, accreditation, internal planning templates
Portfolio management training	Create, edit, share institution groups
IR team proficiency	Each team member runs analyses independently
Third deliverable	Board-ready presentation generated by IR team

Success metric: The IR team produces a complete analysis — report, methodology doc, and presentation — without consultant involvement.

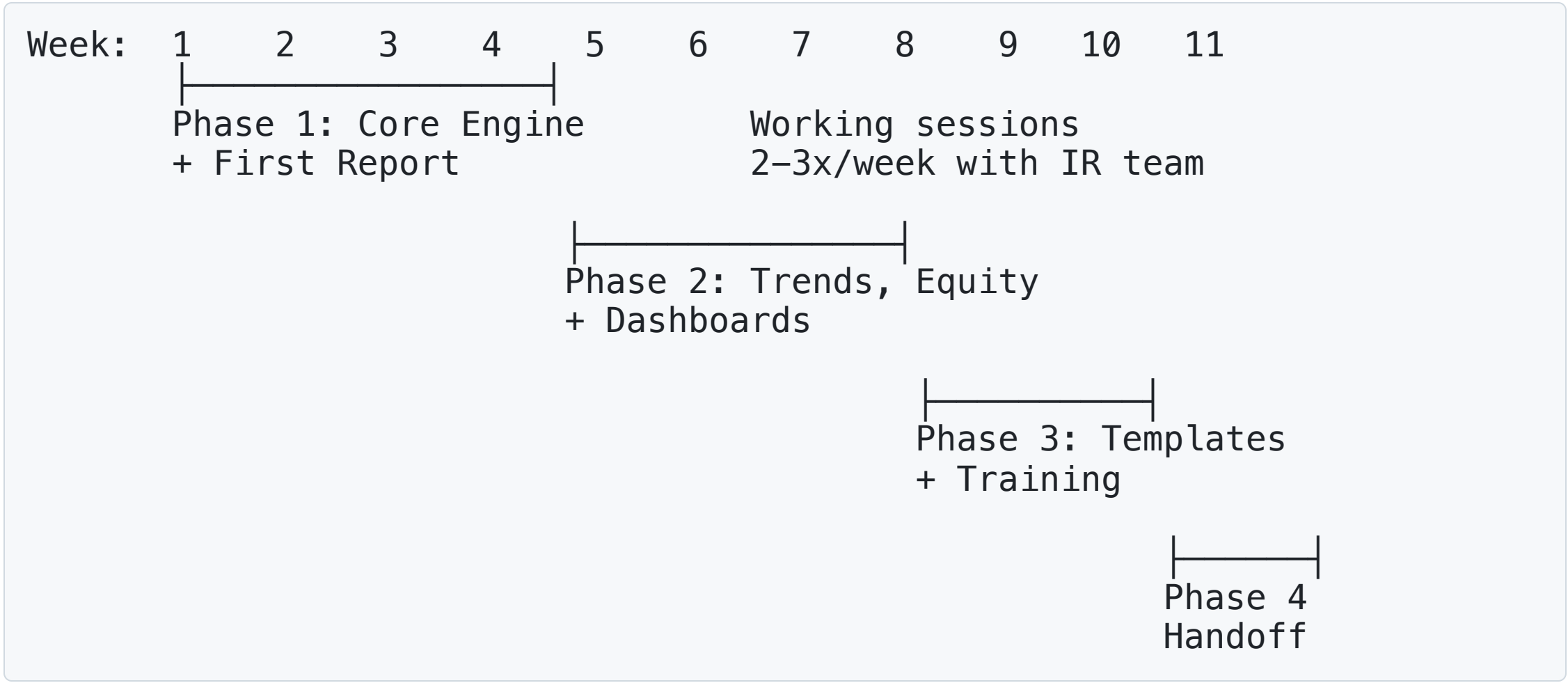
Delivery: Phase 4 (Weeks 9-11)

Handoff + Retainer Transition

Documentation deliverables

Document	Audience	Content
User Guide	IR team	Command reference, workflows, portfolio management
Methodology Reference	Analysts, auditors	Data sources, definitions, methods, limitations
Architecture Document	IT staff	System design, data flow, extension points, security
Runbook	IR team + IT	Data refresh, troubleshooting, environment

Timeline



Total: 8-12 weeks. Each phase produces immediately usable deliverables.

Ongoing Support (Retainer)

Quarterly

- IPEDS data refresh (download new vintage, rebuild, validate)
- Review methodology changes needed for IPEDS survey updates

On demand

- New analyses for emerging institutional questions
- Additional peer portfolios as strategic context shifts
- New data source integration (NSLDS, Common Data Set, state data)
- Feature additions (new capabilities, output formats, custom metrics)
- Accreditation self-study cycle support

Annual

Extensibility Roadmap

Phase	Adds	Value
v1 (initial build)	IPEDS + Scorecard, 4 pipelines, reports + bundles	Core analytical engine
v2	NSLDS, Common Data Set, dashboard + slide generators, auto-peer selection	Richer data, polished outputs
v3	State longitudinal data, custom metrics, saved analysis templates	Deep institutional customization

The modular service architecture means new data sources plug in without modifying existing pipelines. The skill layer means new capabilities are exposed to users without retraining.

IPEDSBrain

8-12 weeks. Runs on a laptop. Every analysis is reproducible.

Technical questions welcome.