# Hybrid Dual-Partner Wiring Diagram: PaLMs{} + LLM (Partner‑L) + Quant (Partner‑Q)

## Overview

This document presents a tidy, implementation-ready wiring diagram for marrying PaLMs{} with a Dual Design Partners loop: Partner‑L (LLM) for proposals, rationale, and orchestration; Partner‑Q (quantitative engines) for simulation, optimization, and metrics. The system enforces incremental redesign via frozen baselines, schema-validated messages, numeric gates, and human sign‑off.

## System Wiring Diagram (ASCII)

┌────────────────────────────┐  
 │ Human Lead │  
 │ Goals, Constraints, Sign │  
 │ ─────────────────────── │  
 │ Approve / Freeze Baseline │  
 └─────────────┬──────────────┘  
 │  
 (1) goals/constraints  
 │  
┌──────────────────────────────┐ │ ┌──────────────────────────────┐  
│ PaLMs{} Hub │◄────────────┼─────────────►│ Artifact Store │  
│ (Orchestrator + Registry) │ │ │ CAD/Mesh/BOM, Logs, Hashes │  
│ • Workflow States & Gates │ │ │ Results Bundles, Manifests │  
│ • Versioning / Baselines │ │ └──────────────────────────────┘  
│ • Schema Validation │ │  
│ • Provenance / Round‑Trip │ │  
└───────────┬───────────┬──────┘ │  
 │ │ │  
 (2a) prompt/spec │ │  
 │ │ │  
 ┌──────▼──────┐ │ (2b) baseline\_ref/artifacts  
 │ Partner‑L │ │ │  
 │ (LLM) │ │ │  
 │ Proposals & │ │ │  
 │ DOE Plans │ │ │  
 └──────┬──────┘ │ │  
 │ JSON: proposal │  
 │ (deltas, DOE, rationale) │  
 │ │  
 │ (3) enqueue jobs │  
 │ via adapters │  
 │ │  
 ┌──────▼──────────────────────────────────▼───────┐  
 │ Partner‑Q Adapters │  
 │ (CFD/FEA/Controls/Cost/Optimizer Runners) │  
 │ • Pull proposal.json │  
 │ • Launch sims/opt │  
 │ • Emit results.json (metrics+CI+violations) │  
 └──────┬───────────────────────────────────┬───────┘  
 │ │  
 (4) results.json (4b) results bundle  
 │ │  
 ┌──────▼──────┐ │  
 │ Partner‑L │ │  
 │ Synthesis │ │  
 │ & Pruning │ │  
 └──────┬──────┘ │  
 │ (5) recommendation │  
 │ │  
┌───────────▼───────────┐ │  
│ PaLMs{} Gates │◄──────────────────────┘  
│ • Numeric checks │  
│ • Constraint guards │  
│ • Dual sign‑off │  
└───────────┬───────────┘  
 │ (6) promote baseline vN→vN+1  
 ▼  
 New Frozen Baseline

## Components

• PaLMs{} Hub: Orchestrator with workflow states/gates, schema validation, provenance, and round‑trip export.

• Partner‑L (LLM): Generates proposals (deltas), DOE plans, narrative rationales; synthesizes Partner‑Q results.

• Partner‑Q Adapters: Execute CFD/FEA/controls/cost/optimizers; return metrics with confidence intervals and violation flags.

• Artifact Store: Immutable storage for CAD/meshes/BOM, logs, manifests, hashes, and result bundles.

• Human Lead: Accepts goals, reviews recommendations, signs decisions, and freezes baselines.

## Message Contracts (Canonical Examples)

1) Baseline (immutable reference)

{  
 "design\_id": "car-aero-v7.3",  
 "hash": "sha256-…",  
 "fidelity": "CFD-RANS-v5",  
 "constraints": {  
 "drag\_delta\_pct\_target": -3.0,  
 "mass\_delta\_pct\_max": 0.5,  
 "yield\_delta\_pct\_min": -0.2,  
 "cooling\_margin\_min": 1.1  
 },  
 "artifacts": {  
 "cad\_uri": "s3://designs/car-aero-v7.3/step/",  
 "mesh\_uri": "s3://designs/car-aero-v7.3/meshes/",  
 "bom\_uri": "s3://designs/car-aero-v7.3/bom.csv"  
 }  
}

2) Proposal (Partner‑L → Partner‑Q)

{  
 "proposal\_id": "prop-2025-09-06-001",  
 "baseline\_ref": "car-aero-v7.3",  
 "deltas": [  
 {  
 "name": "rear-fairing-extension",  
 "component": "rear\_fairing",  
 "param\_changes": { "length\_mm": { "delta": 12, "bounds": [0, 20] } },  
 "expected\_effects": { "drag\_pct": -1.2, "mass\_pct": 0.1 },  
 "risks": ["possible cooling recirculation increase"]  
 },  
 {  
 "name": "wheel-arch-lip-radius",  
 "component": "wheel\_arch",  
 "param\_changes": { "lip\_radius\_mm": { "delta": 3, "bounds": [0, 5] } },  
 "expected\_effects": { "drag\_pct": -0.9, "mass\_pct": 0.0 }  
 }  
 ],  
 "doe\_plan": {  
 "strategy": "latin\_hypercube",  
 "samples": 24,  
 "fidelity": ["fast-CFD", "confirmatory-CFD"],  
 "stopping\_rule": "no further improvement in 2 rounds"  
 },  
 "rationale": "Reduce drag with minimal mass increase while preserving cooling margin."  
}

3) Results (Partner‑Q → Partner‑L)

{  
 "proposal\_id": "prop-2025-09-06-001",  
 "results": [  
 {  
 "delta\_name": "rear-fairing-extension",  
 "metrics": { "drag\_pct": -1.35, "mass\_pct": 0.08, "cooling\_margin": 1.11 },  
 "uncertainty": { "drag\_pct\_ci95": 0.25, "mass\_pct\_ci95": 0.03 },  
 "violations": []  
 },  
 {  
 "delta\_name": "wheel-arch-lip-radius",  
 "metrics": { "drag\_pct": -0.82, "mass\_pct": 0.00, "cooling\_margin": 1.13 },  
 "uncertainty": { "drag\_pct\_ci95": 0.22, "mass\_pct\_ci95": 0.01 },  
 "violations": []  
 }  
 ],  
 "summary": {  
 "pareto\_front": ["rear-fairing-extension", "wheel-arch-lip-radius"],  
 "recommendation": ["rear-fairing-extension", "wheel-arch-lip-radius"]  
 },  
 "env": {  
 "solver": "cfd-rans v5.4",  
 "mesh\_hash": "sha256-…",  
 "container\_digest": "ghcr.io/org/cfd@sha256:…"  
 }  
}

4) Decision Record (PaLMs{} Gate + Human)

{  
 "decision\_id": "dec-2025-09-06-a",  
 "baseline\_ref": "car-aero-v7.3",  
 "accepted\_deltas": ["rear-fairing-extension", "wheel-arch-lip-radius"],  
 "justification": "Joint drag reduction −2.17% within mass & cooling constraints.",  
 "signoff": {  
 "human": "LeadDesigner@company",  
 "timestamp": "2025-09-06T22:15:03Z",  
 "quant\_ok": true  
 },  
 "promoted\_baseline": "car-aero-v7.4"  
}

## Orchestration Steps (Numbered)

1) Human publishes goals/constraints; PaLMs{} records/validates and references the current frozen baseline.

2) Partner‑L converts intent into proposal.json (small deltas + DOE plan + rationale), referencing baseline artifacts.

3) Partner‑Q adapters read proposal.json, run sims/opt, and emit results.json with metrics, CI bands, and violation flags; results bundle stored with hashes.

4) Partner‑L synthesizes results, prunes weak deltas, and returns a recommendation set with readable trade‑offs.

5) PaLMs{} gate enforces numeric checks and requires dual sign‑off (Quant OK + Human).

6) On approval, PaLMs{} promotes vN→vN+1 and archives all artifacts (prompts, configs, solver manifests, hashes).