Blue Whale Master Plan v1.0

Cetacean Labs - Domain-Specific Intelligence Layer

Document Version: 1.0 **Created:** November 1, 2025 **Owner:** Chris McGrath, Cetacean Labs **Status:** Strategic Foundation Document

----

Executive Summary

Blue Whale represents Cetacean's **domain-specific intelligence layer** - a curated library of specialized small language models (3B-13B parameters) that inject expert knowledge into agents, applications, and the Oceanic platform itself. Unlike general-purpose LLMs, Blue Whale models are optimized for specific verticals (Legal, Medical, Finance, Logistics, etc.), delivering superior performance at 10-100x lower cost and <50ms latency.

The Three-Component Intelligence Ecosystem

Blue Whale operates as the centerpiece of a three-part system:

```
┌─────────────────────────────────────────────────────────────
───┐
│                    OCEANIC PLATFORM
│
│              Multi-Cloud AI Infrastructure
│
└─────────────────────────────────────────────────────────────
───┘
                            │
                            │
        ┌───────────────────┼───────────────────┐
        │                   │                   │
    ┌───▼────┐         ┌────▼───────┐      ┌────▼─────┐
    │DOLPHIN │         │BLUE WHALE  │      │PORPOISE  │
    │        │◄────────┤            ├─────►│          │
    │Agents  │  Uses   │SLM Library │Creates│Training │
    │        │         │            │      │Pipeline  │
    └────────┘         └────────────┘      └──────────┘
```

1. Blue Whale (The Library): Curated catalog of production-ready domain SLMs **2. Porpoise (The Factory):** Training pipeline that creates new Blue Whale models **3. Dolphin (The Consumer):** Agent framework that uses Blue Whale for specialized tasks

Strategic Value Proposition

Metric	General LLMs	Blue Whale SLMs
Cost per 1M tokens	$5-30	$0.10-0.50

Inference Latency	500-2000ms	<50ms
Domain Accuracy	60-75%	85-95%
Data Privacy	External APIs	On-premises option
Deployment Flexibility	Cloud-only	Edge/Cloud/Hybrid

Year 1 Business Model

Phase 1 (Months 1-3): Esteemed Ecosystem Licensing

- Dolphin → Esteemed Agents (legal SLMs)
- Porpoise → Esteemed Digital (custom training)
- Orca → Esteemed Ventures (financial SLMs)
- **Revenue Target:** $1.3M from internal licensing

Phase 2 (Months 4-6): Anchor Customer Validation

- DiligenceGPT uses entire stack (Blue Whale + Porpoise + Dolphin)
- Validates product-market fit
- Generates case studies for enterprise sales

Phase 3 (Months 7-12): External Customer Expansion

- Target: 5-10 enterprise customers
- Focus: Finance, Healthcare, Legal verticals
- **Revenue Target:** $2-3M ARR by end of Year 1

Architecture Overview

Blue Whale Library Structure

Blue Whale organizes SLMs into **domain categories**, each containing multiple specialized models:

```
Blue Whale Library
│
├── Legal
│   ├── SaulLM-7B (contract analysis)
│   ├── LegalBERT-3B (case law search)
│   └── ContractNER-3B (entity extraction)
│
├── Medical
│   ├── MedAlpaca-13B (clinical notes)
│   ├── BioGPT-7B (biomedical research)
│   └── DrugGPT-3B (drug interactions)
│
├── Finance
│   ├── FinGPT-8B (financial analysis)
│   ├── BloombergGPT-lite-7B (market data)
│   └── CreditRisk-3B (underwriting)
│
├── Logistics
│   ├── RouteOpt-3B (route optimization)
│   ├── InventoryGPT-7B (inventory forecasting)
│   └── SupplyChain-3B (supply chain modeling)
│
└── Technical
    ├── CodeLlama-7B (code generation)
    ├── SQLCoder-3B (SQL generation)
    └── DevOps-3B (infrastructure code)
```

Multi-Level Intelligence Injection

Blue Whale injects domain expertise at **three levels**:

Level 1: Platform-Level Intelligence

Built into Oceanic platform core functionality:

```
platform_intelligence:
  infrastructure_generation:
    model: "DevOps-3B"
    function: "Terraform/K8s code generation"
    integration: "Oceanic App Builder"

  cost_optimization:
    model: "CloudOpt-3B"
    function: "Multi-cloud cost analysis"
    integration: "Infrastructure orchestration"

  security_compliance:
    model: "SecOps-7B"
    function: "Security policy generation"
    integration: "Compliance monitoring"
```

Level 2: Agent-Level Intelligence

Enhances Dolphin agents with specialized capabilities:

```
agent_intelligence:
  department_agents:
    legal_department:
      primary_model: "SaulLM-7B"
      fallback: "GPT-4"
      use_cases:
        - "Contract review and redlining"
        - "Compliance checking"
        - "Legal research assistance"
    finance_department:
      primary_model: "FinGPT-8B"
      fallback: "Claude 3.5"
      use_cases:
        - "Financial statement analysis"
        - "Investment research"
        - "Risk modeling"

    hr_department:
      primary_model: "HRPolicy-3B"
      fallback: "GPT-4"
      use_cases:
        - "Policy interpretation"
        - "Benefits Q&A"
        - "Onboarding automation"

  individual_agents:
    customization: "Users can select domain SLMs for personal agents"
    example: "Sales agent + FinGPT for financial prospect analysis"
```

Level 3: Application-Level Intelligence

Available via API for custom applications:

```
application_intelligence:
  api_access:
    endpoint: "https://api.oceanic.ai/blue-whale/v1/inference"
    authentication: "Bearer token"
    rate_limits:
      free_tier: "1,000 requests/day"
      professional: "100,000 requests/day"
      enterprise: "Unlimited"

  sdk_support:
    languages: ["Python", "JavaScript", "Go", "Java"]
    example_use: |
      from oceanic import BlueWhale

      client = BlueWhale(api_key="...")
      result = client.inference(
          model="SaulLM-7B",
          prompt="Analyze this contract for liability
clauses",
          context=contract_text
      )
```

Technical Architecture

Model Serving Infrastructure

```
┌───────────────────────────────────────────────────────────┐
│                                                           │
│                  Blue Whale Control Plane                 │
│                                                           │
│   (Model registry, routing, authentication, monitoring)   │
│                                                           │
└───────────────────────────────────────────────────────────┘
                             │
         ┌───────────────────┼───────────────────┐
         │                   │                   │
         ▼                   ▼                   ▼
    ┌────────────┐    ┌──────────────┐    ┌────────────┐
    │   vLLM     │    │    Ollama    │    │ llama.cpp  │
    │   Cluster  │    │    Cluster   │    │  Cluster   │
    │ (GPU-heavy)│    │   (Mixed)    │    │ (CPU-only) │
    └────────────┘    └──────────────┘    └────────────┘
          │                  │                   │
    ┌─────▼──────────────────▼───────────────────▼─────┐
    │         Model Storage (S3/GCS/Azure Blob)        │
    │    - GGUF files (quantized models)               │
    │    - SafeTensors (training checkpoints)          │
    │    - ONNX (optimized inference)                  │
    └──────────────────────────────────────────────────┘
```

Performance Characteristics:

Infrastructure	Best For	Latency	Throughput	Cost
vLLM (GPU)	8B+ models, high throughput	50-200ms	100 req/s/GPU	$
Ollama (Mixed)	General-purpose serving	100-300 ms	50 req/s/node	
llama.cpp (CPU)	Edge deployment, 3B models	200-500 ms	10 req/s/node	$

Natural Language Domain Detection

Blue Whale includes an intelligent routing layer that automatically selects the appropriate domain SLM:

```
# Simplified routing logic
class BlueWhaleRouter:
    def route_request(self, prompt: str) -> str:
        """
        Analyzes prompt and selects optimal domain SLM
        Falls back to general-purpose LLM if no domain match
        """
        # Quick domain classification (< 10ms)
        domain = self.classifier.predict(prompt)

        # Domain confidence scoring
        if domain.confidence > 0.8:
            return self.get_domain_model(domain.category)
        else:
            return "general-llm"  # GPT-4 or Claude fallback

    def get_domain_model(self, category: str) -> str:
        domain_models = {
            "legal": "SaulLM-7B",
            "medical": "MedAlpaca-13B",
            "finance": "FinGPT-8B",
            "logistics": "RouteOpt-3B",
            # ... additional domains
        }
        return domain_models.get(category, "general-llm")
```

Detection Examples:

Input	Detected Domain	Selected Model	Confidence
"Review this NDA for exclusivity clauses"	Legal	SaulLM-7B	0.95
"Analyze Q4 revenue trends"	Finance	FinGPT-8B	0.89
"Optimal route for 20-stop delivery"	Logistics	RouteOpt-3B	0.92

"What's for lunch?"	General	GPT-4	0.45 (fallback)

How the Three Components Relate

The Complete Intelligence Lifecycle

```
┌──────────────────────────────────────────────────────────────────────
───────┐
│                          INTELLIGENCE LIFECYCLE
│
└──────────────────────────────────────────────────────────────────────
───────┘

STEP 1: NEED IDENTIFICATION
User/Agent: "I need a legal contract analysis capability"
    │
    ▼
STEP 2: MODEL CREATION (Porpoise)
┌──────────────────────────────────────────────────┐
│  Porpoise Training Pipeline                      │
                                                    │
│  1. Select base model (Mistral 7B, Llama 8B)   │
│  2. Upload domain data (10K legal contracts)     │
│  3. Fine-tune using LoRA/QLoRA                   │
│  4. Evaluate on test set                         │
│  5. Optimize (4-bit quantization)                │
│  6. Export (GGUF, SafeTensors, ONNX)           │
└──────────────────────────────────────────────────┘
    │
    ▼
STEP 3: CATALOGING (Blue Whale)
┌──────────────────────────────────────────────────┐
│  Blue Whale Registry                             │
                                                    │
│  Model: "CustomLegal-7B"                         │
│  Domain: "Legal - Contract Analysis"             │
│  Performance: 89% accuracy on test set           │
│  Pricing: $0.20 per 1M tokens                    │
│  Deployment: vLLM cluster, 100 req/s             │
└──────────────────────────────────────────────────┘
    │
    ▼
```

```
STEP 4: CONSUMPTION (Dolphin)
┌─────────────────────────────────────────────────┐
│  Dolphin Legal Agent                            │
│                                                  │
│  Task: "Review 50 vendor contracts"             │
│  Model Selection: CustomLegal-7B (Blue Whale)   │
│  Execution: Parallel processing across pod      │
│  Output: Structured analysis + risk flagging    │
└─────────────────────────────────────────────────┘
```

Integration Patterns

Pattern 1: Plug-and-Play Integration (Dolphin ↔ Blue Whale)

Dolphin agents can seamlessly use Blue Whale models without configuration:

```
# Dolphin agent automatically uses Blue Whale for domain
tasks
from dolphin import Agent
from blue_whale import auto_inject

@auto_inject(domain="legal")
class LegalAgent(Agent):
    def analyze_contract(self, contract_text: str):
        # Blue Whale automatically provides SaulLM-7B
        # No explicit model selection needed
        analysis = self.think(
            f"Analyze this contract for risks:
{contract_text}"
        )
        return analysis
```

Pattern 2: Custom Training Integration (Porpoise → Blue Whale)

Porpoise-trained models automatically publish to Blue Whale:

```
porpoise_workflow:
  training_job:
    name: "CustomFinance-3B"
    base_model: "Llama-3.2-3B"
    training_data: "s3://cetacean/finance-data-v2"
    method: "LoRA"
    output_destination: "blue_whale://finance/
CustomFinance-3B"

  auto_publish:
    enabled: true
    visibility: "organization"  # or "public" or "private"
    pricing: "inherited"  # or custom pricing
    deployment: "automatic"  # Deploy to inference cluster
```

Pattern 3: Multi-Model Orchestration (Dolphin Pod)

Dolphin pods can coordinate multiple Blue Whale models:

```
# Complex task requiring multiple domain models
from dolphin import Pod
from blue_whale import BlueWhale

class DiligencePod(Pod):
    def __init__(self):
        self.legal_model = BlueWhale.get("SaulLM-7B")
        self.finance_model = BlueWhale.get("FinGPT-8B")
        self.tech_model = BlueWhale.get("CodeLlama-7B")

    def full_diligence(self, company_data):
        # Parallel execution using specialized models
        legal_review = self.legal_agent.analyze(
            company_data.contracts,
            model=self.legal_model
        )

        financial_review = self.finance_agent.analyze(
            company_data.financials,
            model=self.finance_model
        )

        tech_review = self.tech_agent.analyze(
            company_data.codebase,
            model=self.tech_model
        )

        # Synthesis using general LLM
        return self.synthesize([legal_review,
financial_review, tech_review])
```

Data Flow Architecture

```
┌───────────────────────────────────────────────────────────
───┐
│                       USER REQUEST
│
```

```
│ "Analyze this legal document for compliance issues"
│
└────────────────────────────────────────────────────
───┘
                        │
                        ▼
┌────────────────────────────────────────────────────
───┐
│                DOLPHIN ORCHESTRATOR
│
│ - Receives request
│
│ - Determines domain (Legal)
│
│ - Selects appropriate agent (Legal Department Agent)
│
└────────────────────────────────────────────────────
───┘
                        │
                        ▼
┌────────────────────────────────────────────────────
───┐
│                BLUE WHALE ROUTER
│
│ - Classifies task → "Legal/Compliance"
│
│ - Checks available models:
│
│   ✓ SaulLM-7B (confidence: 0.92)
│
│   ✓ CustomCompliance-3B (confidence: 0.85)
│
│ - Selects SaulLM-7B (higher confidence)
│
└────────────────────────────────────────────────────
───┘
                        │
                        ▼
┌────────────────────────────────────────────────────
───┐
│             INFERENCE INFRASTRUCTURE
│
```

```
│
│   vLLM Cluster
│
│   - Load SaulLM-7B from model storage
│
│   - Execute inference (latency: 120ms)
│
│   - Return structured analysis
│
└────────────────────────────────────────────────────────────
───┘
                              │
                              ▼
┌────────────────────────────────────────────────────────────
───┐
│                    RESULT SYNTHESIS
│
│   Dolphin Agent:
│
│   - Receives SLM output
│
│   - Validates against task requirements
│
│   - Formats for user presentation
│
│   - Logs usage for billing
│
└────────────────────────────────────────────────────────────
───┘
                              │
                              ▼
┌────────────────────────────────────────────────────────────
───┐
│                      USER RESPONSE
│
│   Structured compliance analysis with risk scores
│
└────────────────────────────────────────────────────────────
───┘
```

Simplified Implementation Timeline

Phase 0: Foundation (Weeks 1-2) - COMPLETE

Build in Esteemed Agents before forking to Oceanic

```
week_1-2:
  status: "COMPLETE - Already in Esteemed Agents"
  deliverables:
    infrastructure:
      - "✅ Model serving infrastructure (vLLM/Ollama)"
      - "✅ Basic domain detection"
      - "✅ S3/GCS model storage"

    initial_models:
      - "✅ SaulLM-7B (Legal)"
      - "✅ FinGPT-8B (Finance)"
      - "✅ CodeLlama-7B (Technical)"

    integration:
      - "✅ Esteemed Agents can call Blue Whale models"
      - "✅ Manual model selection working"
```

Phase 1: Core Platform (Weeks 3-8)

Fork to Oceanic and build Blue Whale as standalone product

```
weeks_3-4_oceanic_fork:
  objective: "Create Oceanic MVP with Blue Whale
integration"
  tasks:
    - "Fork Esteemed Agents codebase → Oceanic Platform"
    - "Rebrand Blue Whale as standalone product"
    - "Build Blue Whale marketplace UI"
    - "Implement usage tracking & billing"

  deliverables:
    - "Blue Whale catalog interface (browse models)"
    - "API key generation for external access"
    - "Usage analytics dashboard"

weeks_5-6_porpoise_integration:
  objective: "Connect Porpoise training to Blue Whale"
  tasks:
    - "Build Porpoise → Blue Whale auto-publish pipeline"
    - "Implement model versioning & rollback"
    - "Create training job monitoring"
  deliverables:
    - "One-click model publishing from Porpoise"
    - "Blue Whale registry with version history"
    - "Training cost tracking"

weeks_7-8_dolphin_enhancement:
  objective: "Enhanced Dolphin ↔ Blue Whale integration"
  tasks:
    - "Implement automatic domain detection in Dolphin"
    - "Build model selection optimization"
    - "Create fallback logic (SLM → LLM)"

  deliverables:
    - "Dolphin agents auto-select Blue Whale models"
    - "Intelligent cost optimization"
    - "Performance benchmarking framework"
```

Phase 2: Expansion (Weeks 9-16)

```
weeks_9-12_domain_expansion:
  objective: "Expand Blue Whale catalog to 10+ domains"
  domains:
    legal:
      - "✅ SaulLM-7B (contracts)"
      - "➕ LegalBERT-3B (case law)"
      - "➕ ContractNER-3B (entity extraction)"

    medical:
      - "➕ MedAlpaca-13B (clinical notes)"
      - "➕ BioGPT-7B (research)"
      - "➕ DrugGPT-3B (interactions)"

    finance:
      - "✅ FinGPT-8B (analysis)"
      - "➕ BloombergGPT-lite-7B (markets)"
      - "➕ CreditRisk-3B (underwriting)"

    logistics:
      - "➕ RouteOpt-3B (routing)"
      - "➕ InventoryGPT-7B (forecasting)"

    technical:
      - "✅ CodeLlama-7B (code gen)"
      - "➕ SQLCoder-3B (SQL)"
      - "➕ DevOps-3B (infrastructure)"

  deliverables:
    - "15+ production-ready SLMs"
    - "Benchmarking against GPT-4/Claude"
    - "Performance documentation"

weeks_13-16_enterprise_features:
  objective: "Enterprise-grade capabilities"
  features:
    deployment:
      - "On-premises deployment option"
      - "Air-gapped environment support"
      - "Custom model fine-tuning service"

    security:
      - "SOC 2 Type II compliance"
      - "HIPAA-compliant medical models"
      - "Role-based access control"

    management:
      - "Model lifecycle management"
      - "A/B testing framework"
      - "Cost allocation & chargeback"

  deliverables:
    - "Enterprise deployment guide"
    - "Security certification documentation"
    - "Admin management console"
```

Phase 3: Scale & Monetization (Weeks 17-24)

```
weeks_17-20_customer_acquisition:
  objective: "Onboard first 5 external customers"
  activities:
    esteemed_ecosystem:
      - "Esteemed Agents (legal SLMs) - Already using"
      - "Esteemed Digital (custom training) - Sales cycle"
      - "Esteemed Ventures (financial SLMs) - Pilot phase"

    anchor_customer:
      - "DiligenceGPT (full stack validation)"
      - "Case study development"
      - "Reference customer program"

    external_prospects:
      - "5 enterprise POCs (Finance, Healthcare, Legal)"
      - "SMB segment exploration"

  revenue_target:
    q1: "$325K (Esteemed ecosystem)"
    q2: "$650K (Esteemed + DiligenceGPT)"
    q3_q4: "$1.3M (external customers)"

weeks_21-24_optimization:
  objective: "Improve performance & reduce costs"
  initiatives:
    performance:
      - "Model quantization (int4 across all models)"
      - "Inference optimization (target: <50ms p95)"
      - "Batching & caching strategies"

    cost:
      - "GPU utilization optimization"
      - "Auto-scaling based on demand"
      - "Reserved instance planning"

    quality:
      - "Continuous model evaluation"
      - "Feedback loop integration"
      - "Model retraining automation"

  targets:
    - "50% cost reduction (vs. Week 8 baseline)"
    - "2x throughput improvement"
    - "95%+ customer satisfaction"
```

Year 1 Success Metrics

```
technical_metrics:
  catalog_size: "15+ domain SLMs in production"
  performance:
    latency_p95: "< 200ms"
    throughput: "100 req/s per GPU"
    availability: "99.9%"

  integration:
    dolphin_agents: "30+ agents using Blue Whale"
    porpoise_models: "10+ custom models trained"
    api_customers: "20+ external integrations"

business_metrics:
  revenue: "$1.3M ARR (Year 1)"
  customers:
    internal: "3 (Esteemed entities)"
    anchor: "1 (DiligenceGPT)"
    external: "5-10 (enterprises)"

  usage:
    daily_requests: "1M+"
    monthly_active_users: "500+"
    models_deployed: "15+"

operational_metrics:
  team_size: "12 (from 4)"
  infrastructure_cost: "< $50K/month"
  gross_margin: "70%+"
  customer_churn: "< 5%"
```

Competitive Positioning

Market Landscape

```
┌──────────────────────────────────────────────────────────
────┐
│              DOMAIN-SPECIFIC AI MARKET MAP
│
└──────────────────────────────────────────────────────────
────┘

GENERAL-PURPOSE LLMs (Not Direct Competitors)
├── OpenAI (GPT-4o) - $30/1M tokens
├── Anthropic (Claude 4.5) - $15/1M tokens
├── Google (Gemini Pro) - $7/1M tokens
└── Meta (Llama 3.3 405B) - Self-hosted

EMERGING DOMAIN SLM PLAYERS (Competitors)
├── Cohere for AI (Custom models) - $$$
├── Hugging Face (Model hosting) - $$
├── Together AI (Inference) - $$
└── Replicate (Model deployment) - $

CETACEAN BLUE WHALE (Our Position)
├── Pre-built domain SLMs (15+ models) ✓
├── Custom training pipeline (Porpoise) ✓
├── Integrated agent framework (Dolphin) ✓
├── Multi-level intelligence injection ✓
└── Pricing: $0.10-0.50/1M tokens ✓
```

Differentiation Matrix

Capability	OpenAI	Cohere	HuggingFace	**Blue Whale**
Domain Specialization	❌ General	✅ Custom	⚠️ DIY	✅ Pre-built
Training Pipeline	❌	✅	⚠️ Complex	✅ Porpoise
Agent Integration	❌	❌	❌	✅ Dolphin
Multi-Cloud	❌	❌	⚠️ Limited	✅ Native
Data Privacy	❌ External	⚠️ Hybrid	✅ Self-host	✅ On-prem option
Cost	$	$	$ (DIY)	$
Time to Production	Immediate	Weeks	Months	Days
Support & Managed Services	✅	✅	❌	✅

Competitive Advantages

1. **Integrated Ecosystem:** Only solution combining SLM library + training

pipeline + agent framework

2. Validated by Production: DiligenceGPT proves real-world viability from day one

3. Multi-Level Injection: Intelligence at Platform/Agent/Application levels

4. Rapid Deployment: Days to production vs. weeks/months for competitors

5. Cost Structure: 10-100x cheaper than general LLMs for domain tasks

6. Esteemed Ecosystem: Built-in customer base and revenue from day one

---

Deep Dive References

Technical Specifications

- **Oceanic Platform Technical Specification v2.1**
  - Complete platform architecture
  - Dolphin, Porpoise, Blue Whale detailed specs
  - Infrastructure, security, compliance
  - 5-year product roadmap
- **Oceanic Platform Technical Specification v2.0**
  - Previous iteration with foundational architecture
  - Performance benchmarks and scaling limits
  - Integration specifications
- **Oceanic Platform Technical Specification v1.0**
  - Original vision and architecture
  - DevPanel partnership rationale
  - Initial product suite definitions

Business Documentation

- **Cetacean Platform Product Roadmap**
  - 2026-2030 product release strategy
  - Market positioning and TAM analysis
  - Revenue projections and customer segments
- **Cetacean Executive Summary**
  - Investment thesis and fundraising materials
  - Team, technology, and traction
  - Financial projections

Intelligence System

- **Orca Intelligence README**
  - SAFLA, Ruv-FANN, Goalie specifications
  - Enterprise intelligence use cases

    - Integration with Blue Whale SLMs

Strategic Context

- **Golden Path Analysis**
    - Sub-2-minute deployment validation
    - Infrastructure automation patterns
    - Competitive advantage documentation

Investment Highlights

Why Blue Whale Matters for Fundraising

1. Clear Path to Revenue

- $1.3M ARR from Esteemed ecosystem (Year 1)
- DiligenceGPT as anchor customer validation
- Proven 10-100x cost advantage vs. general LLMs

2. Defensible Technology

- 5-8 patent applications (agent orchestration, SLM training, multi-cloud optimization)
- Proprietary Porpoise training pipeline
- Integrated Dolphin agent framework

3. Large Addressable Market

- $10-14 trillion autonomous economy opportunity
- Blue Whale positions Cetacean as infrastructure backbone
- Vertical expansion: Finance → Healthcare → Legal → Energy → Space

4. Rapid Time-to-Market

- Fork existing Esteemed Agents (5+ years of development)
- 6-week implementation timeline
- Production-validated from day one

5. Capital Efficient Growth

- Licensing model within Esteemed ecosystem
- Channel partner approach for external scaling
- Managed services create recurring revenue

Funding Requirements

Seed Round: $5Mfor7$**71.4M post-money)**

Use of Funds:

- **$2M** - Engineering team expansion (12 → 25 people)
- **$1.5M** - Infrastructure & model training costs
- **$1M** - Sales & marketing (enterprise customer acquisition)
- $500K - Legal (patents, contracts, compliance)

Milestones:

- Month 6: $650K ARR, DiligenceGPT case study
- Month 12: $1.3M ARR, 5-10 external customers
- Month 18: $3M ARR, 20+ external customers
- Month 24: $7M ARR, Series A positioning

Conclusion

Blue Whale represents the **intelligence layer of the autonomous economy** - a curated library of domain-specific SLMs that make AI both affordable and effective for real-world enterprise use cases. By integrating tightly with Porpoise (training) and Dolphin (agents), Blue Whale creates a comprehensive intelligence ecosystem that no competitor can match.

Next Steps

Immediate Actions (Week 1):

1. Executive alignment on Phase 1 roadmap
2. Finalize Porpoise → Blue Whale integration architecture
3. Begin customer discovery for external expansion
4. Initiate patent filing process

Near-Term Priorities (Weeks 2-8):

1. Fork Esteemed Agents → Oceanic Platform
2. Build Blue Whale marketplace UI
3. Onboard DiligenceGPT as anchor customer
4. Expand catalog to 10+ domain SLMs

Strategic Objectives (Months 3-12):

1. Achieve $1.3M ARR from combined sources
2. Generate 3-5 enterprise case studies
3. File 5-8 patent applications
4. Position for Series A fundraising

Document Owner: Chris McGrath, Founder & CEO, Cetacean Labs **Last Updated:** November 1, 2025 **Version:** 1.0 **Status:** Strategic Foundation Document

This document is confidential and proprietary to Cetacean Labs. Distribution limited to executives, board members, and approved investors.