

VELAR

AI Core

System Architecture

Modular architecture for context-aware business intelligence, decision support, recommendations, actions, automation and workflows.

ARCHITECTURE STATUS Existing foundation + extensible design

August 2026

01

Architecture Overview

VELAR AI Core is organized as a modular, layered architecture rather than a single monolithic AI component.

The architecture connects business data to contextual processing, memory, intelligence, decision support, reasoning, recommendations, actions, automation and workflow execution. The design creates explicit boundaries between layers so that individual capabilities can be extended or replaced without redesigning the entire application.

Current architecture should be distinguished from future product development. The presence of an engine, service or interface in the architecture does not by itself mean that VELAR currently provides a production-scale autonomous AI system or its own trained LLM.

BUSINESS DATA → AI CORE

A structured pathway from operational CRM / ERP information toward business outcomes.

Existing architecture

02

Primary Architecture Flow

Core conceptual path supplied for the VELAR AI Core.

BUSINESS DATA

CRM / ERP records and application activity

Existing architecture



CONTEXT

Context assembly and current-state awareness

Existing architecture



MEMORY

Persistent and decision-oriented memory architecture

Existing architecture



INTELLIGENCE

Decision, learning, reasoning and recommendation components

Existing architecture





03

Context Layer

The contextual layer determines what information is relevant to the current application state.

- ContextEngine
- ContextBuilder
- AIPageTracker

- AIContextProvider
- AIWatcher

The layer provides an architectural mechanism for collecting, assembling and exposing relevant context to downstream AI components. Its purpose is to reduce the coupling between UI state / business data and the intelligence layers.

CONTEXT

Current page + application state + relevant business information

Existing architecture

04

Event System

Events provide a mechanism for communicating meaningful state changes across the AI architecture.

- AIEvent
- AIEventBus

The event system supports an event-driven architecture in which components can react to application or AI-related events without requiring direct point-to-point coupling.

EVENT BUS

AIEvent → AIEventBus → subscribed AI / workflow components

Existing architecture

05

Memory Layer

Memory provides the architecture for retaining relevant state, decisions and feedback.

- MemoryStore
- DecisionMemory
- PersistentMemory
- MemoryListener
- MemoryOptimizer

Memory is intended to support continuity across relevant operations and provide historical information for decision-oriented processing. This should not be represented as evidence that VELAR contains a proprietary trained language model.

COMPONENT	ARCHITECTURAL ROLE	STATUS
MemoryStore	Central memory abstraction	Existing
DecisionMemory	Decision-related memory	Existing
PersistentMemory	Persistent memory abstraction	Existing
MemoryListener	Memory/event observation	Existing
MemoryOptimizer	Memory maintenance / optimization concept	Existing

06

Decision Engine

A dedicated component for structured decision processing.

DECISION ENGINE

Context + Memory + Business Data → DecisionEngine → decision output

Existing architecture

DecisionEngine forms a central intelligence boundary. It can consume information assembled by upstream layers and provide decision-oriented output to reasoning, recommendations and downstream action pathways.

07

Learning Engine

Feedback-oriented learning and decision-outcome tracking architecture.

LEARNING ENGINE

Decision / outcome feedback → LearningEngine → improved future decision context

Existing architecture

The Learning Engine should be described conservatively: it represents an architectural foundation for tracking outcomes and feedback. It should not be described as a self-trained LLM or as autonomous model training unless such functionality is separately implemented and verified.

08

Reasoning Engine

A separate reasoning layer keeps higher-order processing distinct from data collection and execution.

REASONING ENGINE

Context + Memory + Decision signals → structured reasoning → conclusion

Existing architecture

The separation creates an extension point for future model integrations and reasoning strategies.

09

Recommendation Engine

The recommendation layer converts intelligence into actionable suggestions.

RECOMMENDATION ENGINE

Decision + Reasoning + Knowledge → recommended next step

Existing architecture

This layer is intended to separate the generation of recommendations from their eventual execution, allowing human-in-the-loop or automated operating models.

10

Knowledge Layer

Structured business entities and relationships.

- KnowledgeGraph
- entities



The knowledge layer provides a foundation for connecting otherwise isolated business records and creating richer context for intelligence components.

11

Action Layer

A controlled boundary between recommendations and execution.



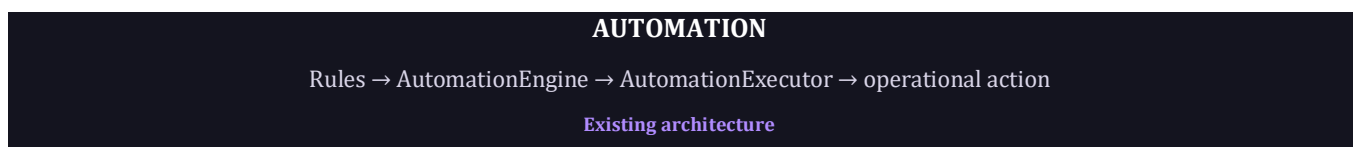
ActionEngine provides a dedicated architectural point at which business actions can be defined and executed.

12

Automation Layer

Automation separates reusable rules from execution.

- AutomationEngine
- AutomationExecutor
- AutomationRules



The architecture supports repeatable operations and creates a pathway for future AI-assisted automation.

13

Workflow Engine

Workflow architecture organizes reusable business processes.

- WorkflowEngine
- WorkflowRunner
- WorkflowRegistry
- WorkflowContext
- WorkflowCondition
- WorkflowAction
- WorkflowHistory
- WorkflowScheduler
- WorkflowLoader
- ClientFollowUpWorkflow

WORKFLOW

Context + Conditions → Actions → Runner → History / Schedule

Existing architecture

The workflow layer is designed to make business processes explicit, reusable and trackable. ClientFollowUpWorkflow is an example of a domain-specific workflow component in the current architecture.

14

AI-driven Automation

The long-term architecture connects intelligence to controlled operational execution.

STAGE	FUNCTION	BOUNDARY
1	Observe	Context / events
2	Understand	Memory / knowledge / reasoning
3	Decide	Decision / recommendation
4	Act	Action / automation
5	Orchestrate	Workflow / history / scheduling

15

AI Dashboard

The user-facing monitoring and intelligence surface.

- AIDashboard
- AIActivity
- AIClientTable

- AICoreHero
- AIHealth
- AIStatCard
- LearningScore
- AIHistory
- AIInsightPanel
- AIPanel

The AI Dashboard brings the architecture into the application interface, providing areas for AI activity, health, history, insights and client-oriented intelligence views.

AI DASHBOARD

AI Core state + activity + insights → user-facing interface

Existing architecture

16

Modular Design

VELAR separates architectural responsibilities into identifiable components.

PRINCIPLE	IMPLEMENTATION DIRECTION	VALUE
Separation of concerns	Distinct context, memory, intelligence and execution layers	Easier extension
Event-driven communication	AIEvent / AIEventBus	Lower coupling
Composable workflows	Registry, context, conditions, actions and runner	Reusable processes
Dedicated execution boundaries	Action / Automation / Workflow layers	Controlled operations
UI separation	AI Dashboard components above core services	Clear presentation boundary

17

Extensibility

Designed to accommodate future AI models, persistence and deployment strategies.

- AI model/provider integration can be added behind appropriate intelligence boundaries.
- A production database can replace or extend current data persistence mechanisms.
- Cloud / SaaS services can be introduced around the desktop foundation.
- Additional workflow types can be registered without redesigning the workflow concept.
- Advanced automation can connect recommendations to controlled action execution.

These items are development potential, not claims that the corresponding production infrastructure is already implemented.

18

Current Architectural Status

Clear separation between present architecture and future development.

AREA	CURRENT STATUS	FUTURE EXTENSION
Context / events	Architecture and components present	Deeper context sources
Memory	Memory components present	Production persistence / optimization
Decision / reasoning / recommendation	Modular engine architecture	Advanced model-backed intelligence
Learning	Feedback-oriented architecture	Advanced learning loops
Knowledge	KnowledgeGraph / entities	Richer enterprise knowledge graph
Actions / automation	Dedicated architecture	Broader execution integrations
Workflow	Engine, runner, registry and related components	Autonomous / multi-stage workflows
Cloud / SaaS	Not claimed as complete	Future migration / deployment

19

Architectural Value

Why the architecture matters in an acquisition context.

- Provides a coherent AI-oriented architecture rather than isolated UI features.
- Separates business data from context and intelligence processing.
- Creates explicit memory and decision boundaries.
- Creates a path from recommendation to controlled execution.
- Supports reusable automation and workflow concepts.
- Leaves room for buyer-selected AI models, database and deployment architecture.

The value proposition is therefore the combination of an existing desktop product foundation and a structured architecture that can be extended into a broader AI-enabled business platform.

20

Architecture Boundaries & Limitations

Important due-diligence framing.

- VELAR is not presented as a trading terminal.
- VELAR is not presented as having its own trained LLM.
- A fully production-grade cloud SaaS infrastructure is not claimed.
- A fully implemented production database is not claimed.
- Architecture components should be validated against the source code during technical due diligence.
- Future AI autonomy, model performance and scalability depend on subsequent engineering and infrastructure choices.

These boundaries are intentional and form part of a technically accurate acquisition description.

21

Architectural Conclusion

A modular foundation for continued AI product development.

VELAR AI Core establishes a layered path from business data to business results: context, memory, intelligence, decision, reasoning, recommendation, action, automation and workflow. The architecture is sufficiently modular to

provide clear extension points for future AI models, production data infrastructure, deeper intelligence and cloud deployment.

VELAR AI CORE

Business Data → Context → Memory → Intelligence → Decision → Reasoning → Recommendation → Action → Automation →
Workflow → Business Result → Feedback / Memory

Existing architecture

VELAR PLATFORM / SYSTEM ARCHITECTURE / AUGUST 2026