

Kein Feuerlöschern mehr. Endlich wissen.

aprevisAI überwacht Ihre Oracle-Umgebung kontinuierlich, erkennt subtile Verhaltensänderungen in Echtzeit und erklärt Anomalien mit strukturierter, evidenzbasierter Ursachenanalyse.

WARUM

Klassisches Oracle-Monitoring ist reaktiv

Oracle-Datenbanken betreiben Ihre kritischsten Systeme. Dennoch verlassen sich die meisten Tools auf statische Schwellenwerte und isolierte Metriken — Teams bleiben permanent einen Schritt zurück.

| PROBLEM | OPERATIVE AUSWIRKUNG |
|--|--|
| <ul style="list-style-type: none"> Späte Erkennung | Performance-Einbußen erst sichtbar, wenn Nutzer bereits betroffen sind |
| <ul style="list-style-type: none"> Alert-Müdigkeit | Signalrauschen übertönt echte, handlungsrelevante Probleme |
| <ul style="list-style-type: none"> Manuelle Ursachenanalyse | Stundenlange Untersuchungen verbrauchen wertvolle DBA-Zeit |
| <ul style="list-style-type: none"> Plan-Instabilität | Execution-Plan-Änderungen erst nach dem Schaden entdeckt |
| <ul style="list-style-type: none"> Operative Überlastung | Steigende Belastung für DBAs — nicht skalierbar im Enterprise-Bereich |

Moderne Datenbank-Operationen brauchen Verhaltensintelligenz — nicht nur Alerts.

WIE

Entwickelt für echte Oracle-Umgebungen

Eine speziell entwickelte Pipeline von sicherer Telemetrie-Erfassung bis zur KI-generierten Ursachenanalyse — konzipiert für produktionskritische Oracle-Umgebungen.

- Sammeln**
 Read-only Collector streamt operative Telemetrie aus Ihrer Oracle-Datenbank. Keine Agents. Vernachlässigbare Workload-Auswirkung.
- Lernen**
 KI-Modelle bauen datenbankspezifische Verhaltens-Baselines aus historischen Mustern. Das System lernt, was „gesund“ für Ihre Umgebung bedeutet — keine generischen Schwellenwerte.
- Erkennen & Korrelieren**
 Multivariate Analyse verbindet Plan-Änderungen, Wait-Events, Query-Regressionen und Workload-Verschiebungen zu einer einzigen Incident-Timeline — bevor sie eskalieren.
- Erklären & Empfehlen**
 Rangierte Hypothesen, Evidenzlisten und nächste Schritte — mit voller Nachverfolgbarkeit. Menschliche Freigabe immer erforderlich. Keine Auto-Remediation.



VORHER & NACHHER

Von der Alert-Flut zur Incident-Klarheit — in Minuten

OHNE APREVISAI — ALARMFLUT

- 14:32:01 **CRIT** CPU-Auslastung 94% auf RAC-Node 1
- 14:32:03 **CRIT** Response-Time-SLA-Verletzung — Orders API
- 14:32:05 **WARN** DB-Time-Spitze: 3,2x Baseline
- 14:32:07 **WARN** Lock-Wait-Events +340% letzte 5 Min
- 14:32:07 **WARN** Lock-Wait-Events +340% letzte 5 Min
- 14:32:09 **CRIT** Full Table Scan erkannt — orders tbl
- 14:32:12 **WARN** Statistiken zuletzt vor 9 Tagen aktualisiert
- 14:32:14 **INFO** Execution-Plan-Hash geändert: orders_q1
- 14:32:14 **WARN** Buffer-Cache-Hit-Ratio auf 71% gefallen

⌛ 60 Min Isolierte Alerts ⌛ 8h MTTR
ZEIT BIS ROOT CAUSE KEIN KONTEXT NACH NUTZERAUSWIRKUNG



APREVISAI — KORRELIERTE INCIDENT-TIMELINE

- 14:29:14 **Execution-Plan-Änderung erkannt** **EVIDENZ**
 orders_q1 von Index Range Scan auf Full Table Scan gewechselt. Plan-Hash: 8f3a→2cfd. Ausgelöst durch veraltete Statistiken auf ORDERS (letzte Aktualisierung: 9 Tage)
- 14:31:47 **Lock-Contention-Kaskade** **EVIDENZ**
 Full Scan hält Row-Level-Locks 3,8x länger als Baseline. Lock-Wait-Events +340%. 14 Sessions hinter orders_q1 in Warteschlange.
- 14:32:01 **CPU- & Response-Time-Auswirkung** **EVIDENZ**
 CPU 94% (x3,2 Baseline). Orders-API-Latenz verletzt SLA. Buffer-Cache-Hit-Ratio fällt auf 71% durch Full-Scan-I/O.
- 14:33:19 **Root Cause identifiziert** **ROOT CAUSE**
 Veraltete Optimizer-Statistiken auf ORDERS verursachten Plan-Regression 3 Min vor Nutzerauswirkung. Alle nachgelagerten Alerts sind Symptome dieser einen Änderung.
- 14:36:32 **Empfohlene Aktion** **AKTION**
 DBMS_STATS.GATHER_TABLE_STATS auf ORDERS ausführen. Stats-Refresh-Zeitplan für High-Volume-Tabellen prüfen. Menschliche Freigabe erforderlich.

< 2 Min 1 Ursache ⌛ 3min MTTD
ZEIT BIS ROOT CAUSE VS. 8 ALERTS VOR NUTZERAUSWIRKUNG

DEPLOYMENT & VERTRAUEN

Enterprise-Ready von Anfang an

- ☑ DSGVO-konform — entwickelt und betrieben in Deutschland
- ☑ Optional vollständig On-Premise-Deployment
- ☑ Keine invasiven Agents — nur Read-only-Datenbankzugriff
- ☑ Menschliche Validierung immer erforderlich — keine Auto-Remediation
- ☑ Vollständiger Incident-Audit-Trail (menschliche Aktionen + KI-Vorschläge protokolliert)
- ☑ Oracle RAC & Multi-Tenant

KI-GESTÜTZTE ANALYSE

Ihr KI-Oracle-Analyst

Stellen Sie Fragen in natürlicher Sprache und erhalten Sie strukturierte, evidenzbasierte Antworten direkt aus Ihrer Datenbank-Telemetrie.

- Fragen Sie in natürlicher Sprache zu jedem Incident
- Erkunden Sie Anomalie-Timelines mit vollem Signal-Kontext
- Prüfen Sie beteiligte Execution Plans und Wait Events
- Validieren Sie Root-Cause-Hypothesen mit korrelierter Evidenz

Von DBAs, für DBAs