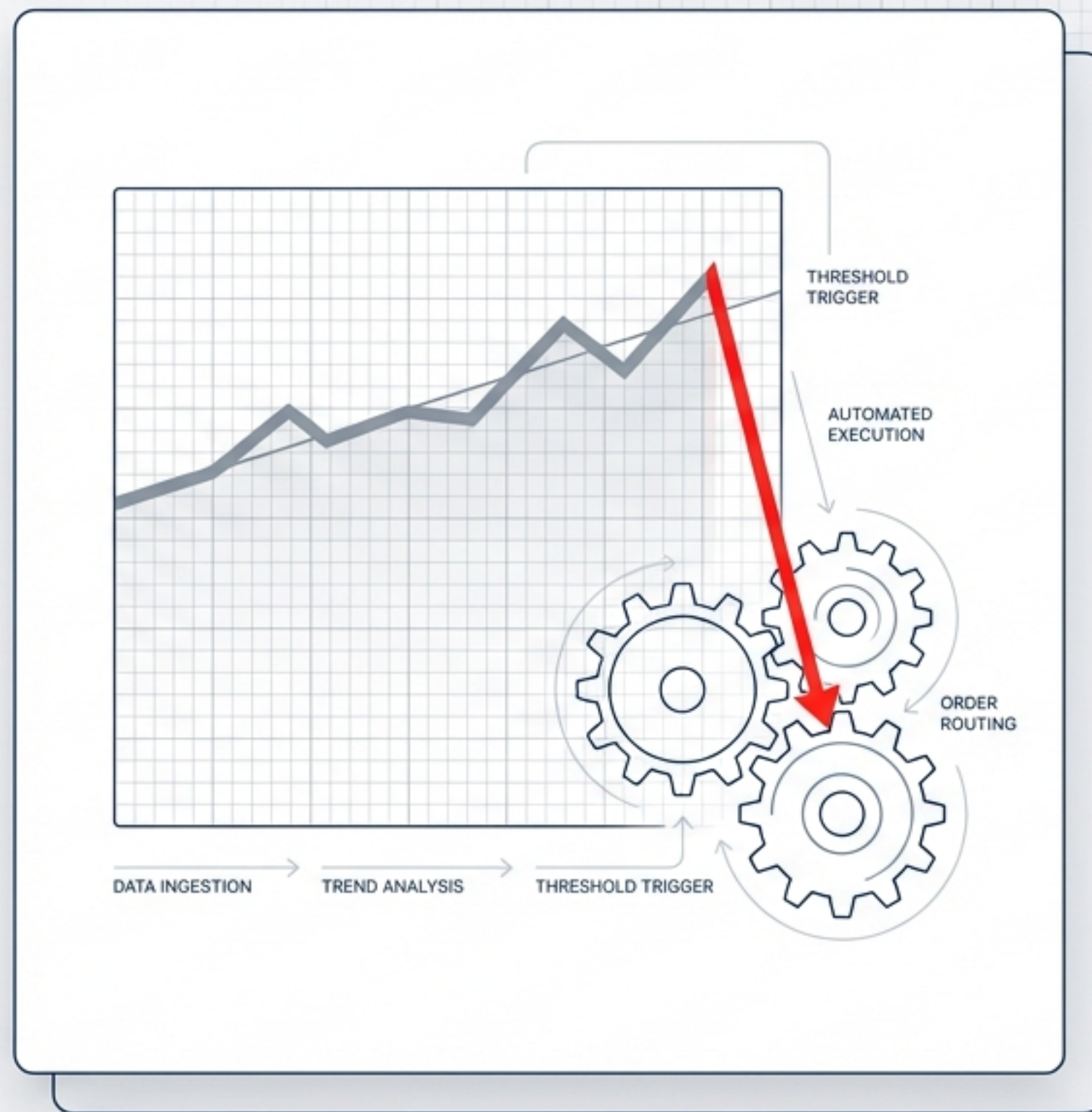


JRA オッズ変動率トラッキング & 自動投票システム

運用アーキテクチャ・ブループリント



コアコンセプトと3つの自動化フェーズ



1. 追跡 (Tracking)

リアルタイムでのオッズ変動監視。当日の全レースを対象に、netkeiba.comから出馬表を取得し、スケジュールに沿って正確なタイミングでオッズをトラッキング。



2. 検知 (Detection)

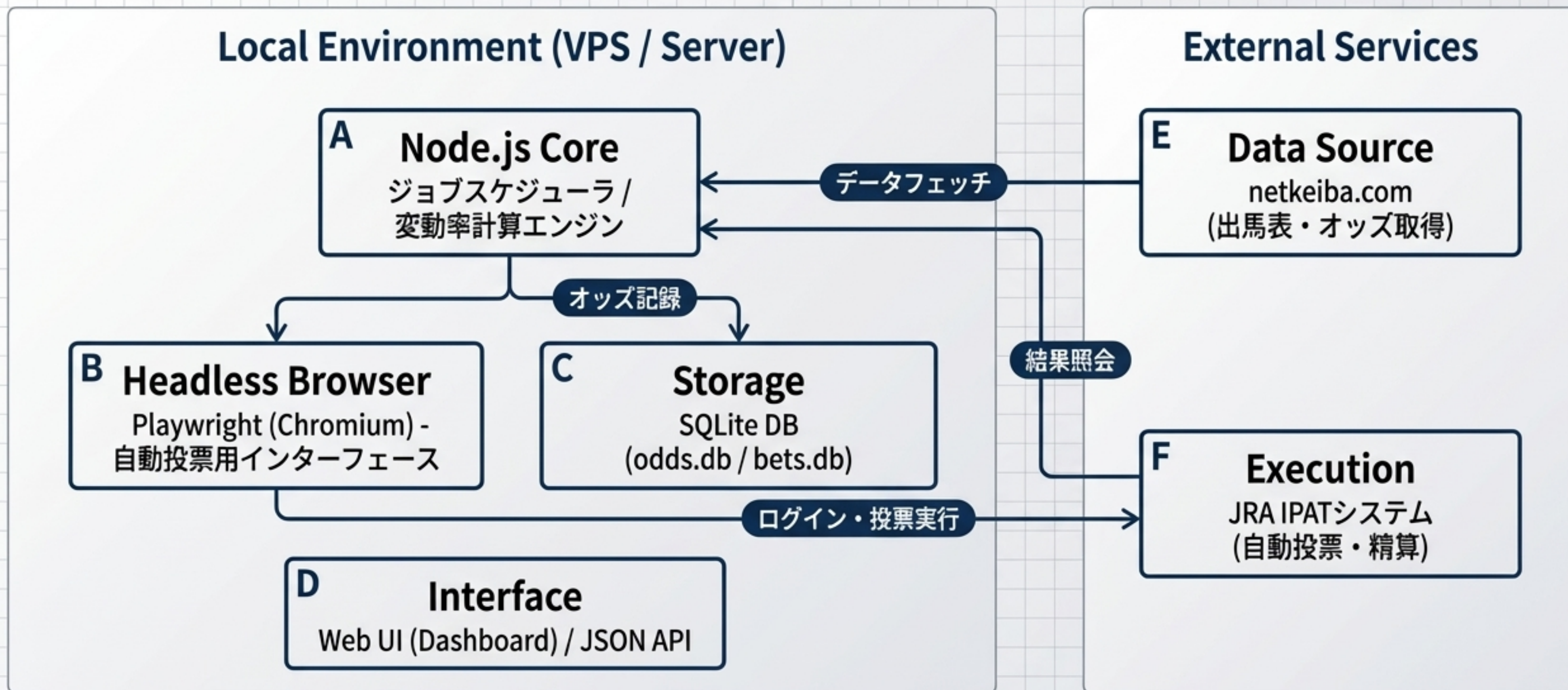
締切直前の不自然なオッズ急落（大口投票による人気集中）を捕捉。過去の変動率データに基づき、アルゴリズムが自動的に異常値を判定。



3. 実行 (Execution)

JRA IPATとの連携による完全自動の単勝投票と精算処理。Playwrightを活用し、UIを介さないブラウザ自動化で確実なエントリーを実現。

システムアーキテクチャ全体図



マクロ・タイムライン：1日の完全自動運用フロー

08:30

日中（レース進行）

18:00

初期化。netkeiba.comから当日レース一覧・出馬表を取得。各レースのオッズ取得ジョブをシステムに一括登録。

レースごとの自動処理ループ（オッズ取得～投票判定）。
※次スライドで詳細化

精算処理。全レース結果を一括取得し、当日の投票精算を完了。

08:30以降にシステムを起動した場合

起動時に当日のスケジュール取得を実行。既に過ぎた時間のジョブはスキップされ、起動時刻以降（例: 14:00起動なら14:00以降の発走レースのみ）のジョブが自動登録・追従されます。

マイクロ・タイムライン：レース直前の自動化シーケンス

投票締切 = 発走1分前



アルゴリズム投票ロジック (Decision Funnel)

1. タイミング
(When)

投票締切1分前（発走2分前）の最終データ取得完了時。

2. 判定基準
(What)

暫定変動率（締切6分前 → 1分前）が **-10%以下** に急落しているか。

3. 対象抽出
(Who)

条件を満たす馬が複数いる場合、
「最も変動率が低い（＝最も人気が集中した）」
馬を1頭のみ厳選。

4. 実行
(Action)

IPATにて「単勝」「100円」を
完全自動投票し、成否に関わらず
bets.db に記録。

自動エントリー完了

ダッシュボード・インターフェース：オッズ変動一覧 (/)

中山 11R - 皐月賞 - 15:40発走

📅 ?date=YYYY-MM-DD

馬名	6分前	3分前	1分前	確定	変動率
★ エフフォーリア	3.1	3.0	2.9	2.8	-3.2%
2 コントレイル	2.5	2.3	2.1	2.0	-12.5%
3 グランアレグリア	4.0	4.2	4.5	4.6	+15.2%
4	3.0	4.0	5.3	3.2	+1.7%
5	3.0	4.7			

戦略別統計

-10%: 33.3% / 105.2%
-15%: 41.2% / 112.8%
-20%: 50.0% / 120.1%

① 日付指定・レースメタデータ表示

② 1着馬は「★マーク・金背景」で強調表示


③ 急落率別の的中率・回収率をリアルタイム集計

変動率カラーコード

- 赤系: 10%超の下落 (大口投票シグナル)
- オレンジ系: 0~10%の下落
- 黄緑系: 0~10%の上昇
- 緑系: 10%超の上昇 (人気離散シグナル)

ダッシュボード・インターフェース：投票履歴 (/bets)

全期間サマリー (All-Time Summary)

総投資額 / 総払戻額 
¥15,400 / ¥18,200

総投票数 / 的中数 
154 / 32

的中率
20.7%

↗ ROI (回収率)
118.1%

システム全体の投資対効果を俯瞰するKPIボード。

日別リスト (Daily Betting Log)

</bets?date=YYYY-MM-DD>

レース情報	馬番	投票額	対象時オッズ	変動率	結果	払戻額
中山11R 皐月賞	4	¥1,000	5.3	+1.7%	的中	¥5,300
中山11R 皐月賞	9	¥1,000	2.0	-12.5%	不的中	¥0

日付指定で特定日のトランザクションを詳細に検証。

システムのインストールとビルド環境構築

1. 前提条件 (Prerequisites)

Node.js 24.x 以上
および npm のイン
ストール。



2. 依存パッケージ 導入

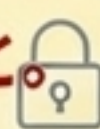
コアパッケージと、自
動投票用の Headless
Browser である
Chromium
(Playwright経由) を
インストール。



3. 環境変数の準備 ⚠

プロジェクトルートに
.env ファイルを作成。

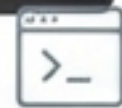
**警告: セキュリティリ
スク防止のため、.env
ファイルは絶対にGitへ
コミットしないこと。**



4. ビルド実行 (Build)

ターミナルで npm
run build を実行し、
本番用の実行ファイ
ルを生成。

```
> npm run build
```



環境変数設定：自動投票と認証情報のセキュアな管理

.env

AUTO_VOTE_ENABLED=true

INET_ID=*****

USER_ID=*****

PASSWORD=*****

P_ARS=*****

自動投票トグルスイッチ



AUTO_VOTE_ENABLED=true を指定した場合のみ、締切1分前の自動投票プロセスがアクティブになります。falseの場合はトラッキングのみ実行されます。

IPAT認証情報（厳重管理）



実際の資金を動かすため、JRAから発行されたINET_ID、USER_ID、PASSWORD、P_ARS（暗証番号）を正確に記述してください。
外部流出に厳重に注意してください。

本番環境（VPS）へのデプロイメントとデーモン化



1. ファイル転送

ビルド済みのファイル群をVPS環境へ安全にアップロード。



2. サーバー側セットアップ

VPS上でのパッケージ解決と初期構成の完了。



3. systemd によるサービス化

サーバー再起動時にもシステムを自動復旧させるため、systemd を用いてプロセスをデーモン化（常駐化）。





⚠ 運用上の重要事項：.env の配置パス

.env ファイルの配置場所に注意してください。

node dist/index.js を実行するディレクトリ（systemd設定における WorkingDirectory パス）に確実に配置する必要があります。

配置ミスは環境変数の読み込みエラーに直結し、システムが起動しません。

データベース・アーキテクチャと運用管理

		
	data/odds.db	data/bets.db
役割	市場データ・解析基盤	トランザクション・財務基盤
格納内容	レース情報、オッズスナップショット（時系列）、変動率計算結果、レース着順結果。	投票実行記録、精算結果ステータス。




リセット機能

万が一DBファイルが破損した場合、該当ファイルを直接削除してシステムを再起動することで、自動的にクリーンな状態で再生成（初期化）されます。

バックアップ推奨

運用履歴の長期保全のため、定期的なファイルコピーによるバックアップを推奨します。

トラブルシューティング診断マトリクス

症状 (Symptom)	原因と解決策 (Root Cause & Resolution)
 起動不可 / .env読み込み失敗	WorkingDirectory の指定パスと .env の配置場所の不一致。 systemdのパス設定を再確認する。
 自動投票が 実行されない	<ul style="list-style-type: none">① ログで Auto-vote: ENABLED を確認。② 認証情報の不備を確認。③ ログに No horses with fluctuation rate <= -10% がある場合、単に「条件を満たす馬が不在」でありシステムは正常稼働。
 502 Bad Gateway エラー	プロセスがクラッシュして停止している状態。ログから原因を特定を後、ターミナルで <code>systemctl restart odds-fluctuations</code> コマンドを実行しサービスを再起動。

システムの安全性と拡張インターフェース



Graceful Shutdown (安全な停止プロセス)

Ctrl+C や SIGINT/SIGTERM シグナルを受信した際、プロセスを即座に強制終了するのではなく、実行中のトランザクションを待機し、DB の書き込みを安全に保存してから自動的にシャットダウンする保護機能を搭載しています。



JSON API エンドポイント (/api/results)

蓄積されたデータをブラウザUIだけでなく、機械可読なJSONフォーマットで提供します。外部プログラムや独自の分析スクリプトからのデータ抽出・連携をシームレスに実現します。

```
{ "status": "success", "data": [...] }
```

システム運用開始に向けた最終確認

- ✓ **環境構築:** Node.js 24+, Chromium依存関係のインストール完了
- ✓ **セキュリティ:** .env ファイルの設定完了およびGit管理からの除外確認
- ✓ **デーモン化:** systemdの稼働確認および WorkingDirectory のパス整合性
- ✓ **自動実行:** AUTO_VOTE_ENABLED=true の設定およびIPAT連携ログの正常出力

堅牢なインフラストラクチャとアルゴリズム制御に基づく、完全無人のオッズトラッキング & 自動利益化サイクルが確 されました。システムの監視はダッシュボードからリアルタイムに行えます。