

Zabbix 3.0+

Where do we go

私達は何を開発しているのか？

- 信用できる監視プラットフォーム
- ITに限らず
- 信用：信頼、安定、正確、予測可能
- いかなる規模の環境にも適する

1998 年に
遡ってみましょう

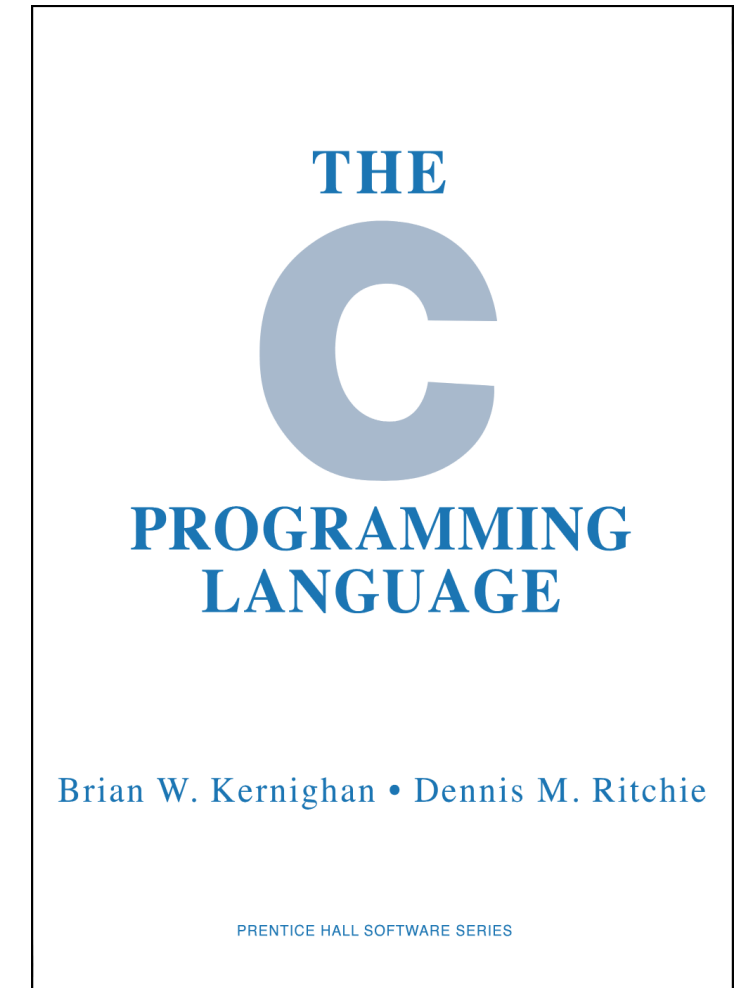
Will we meet Perl there?

1998 年からのツールボックス

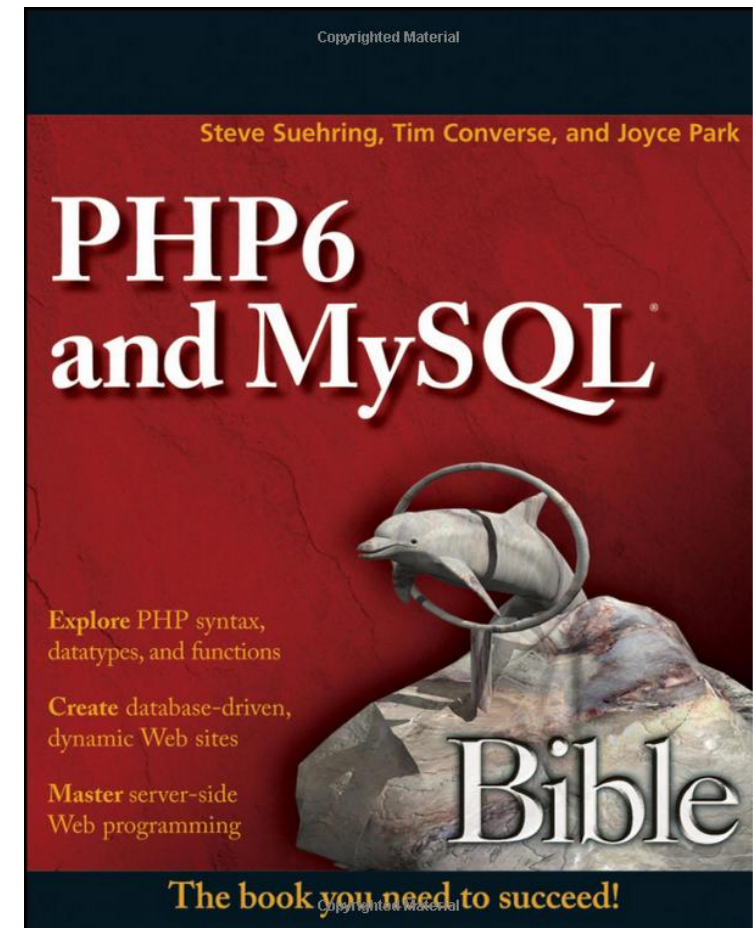
- 当初は Perl を使用、そして下記へ 転換
 - C 言語をすべてのクリティカルなパートに使用
 - PHP 言語を Web インターフェースに使用
 - SQL をバックエンドに使用

C 言語

- + 低水準言語
- + 効率のよいコード：速い
- + リソース使用率が最も低い
(CPU/ メモリ)
- + 依存するものもほとんどない
- + 一度コンパイルすれば、
どこでも実行できる
- 開発が遅い
- メモリ、ロック、ポインタ関連エラー



- + 短期間で習得が可能
- + あらゆるプラットフォームで使用可能
- + 現在もとても活発に開発されている
- 動的型付け
- 良いコードを書くためには規律が必要
- 構文解釈：エラーが実行時に出てくる傾向にある



SQL バックエンド

MySQL, PostgreSQL, Oracle, DB2, SQLite

- + トランザクションストレージエンジン：一貫性
- + 標準 API: SQL
- + 導入が簡単
- スケーラビリティ
- 高可用性

SQL ([/'ɛs kju: 'ɛl/](#),^[4] or [/'si:kwəl/](#); **Structured Query Language**^{[5][6][7][8]}) is a [special-purpose programming language](#) designed for managing data held in a [relational database management system](#) (RDBMS).

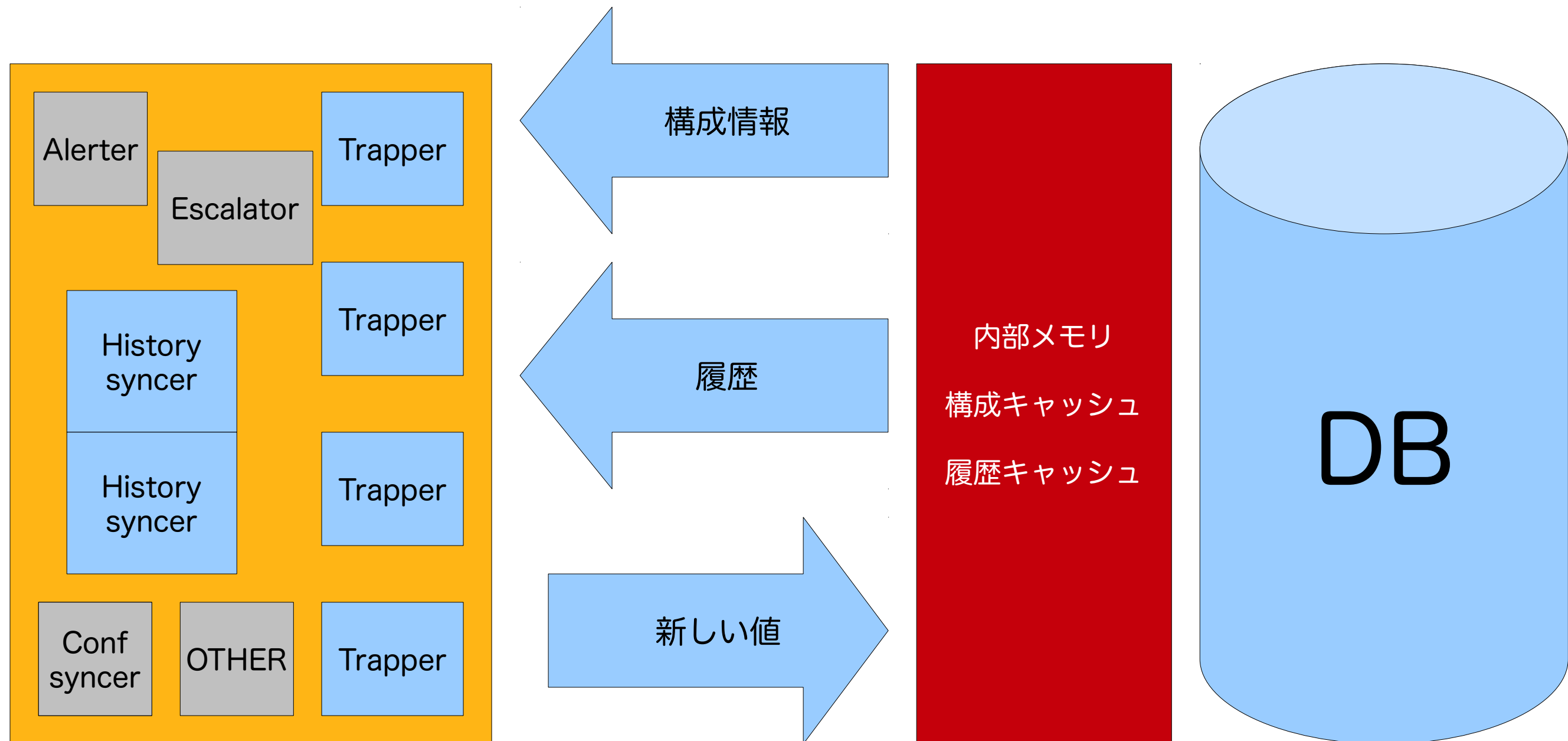
C,PHP,SQL が Zabbix に どう作用するか

+ Zabbix は非常にコンパクト

```
alex@alex: /tmp/zabbix-2.4.0
alex@alex:/tmp/zabbix-2.4.0$ ls -l src/zabbix_server/zabbix_server
-rwxrwxr-x 1 alex alex 1391128 Sep 13 10:09 src/zabbix_server/zabbix_server
alex@alex:/tmp/zabbix-2.4.0$ ls -l src/zabbix_agent/zabbix_agentd
-rwxrwxr-x 1 alex alex 356800 Sep 13 10:09 src/zabbix_agent/zabbix_agentd
alex@alex:/tmp/zabbix-2.4.0$
```

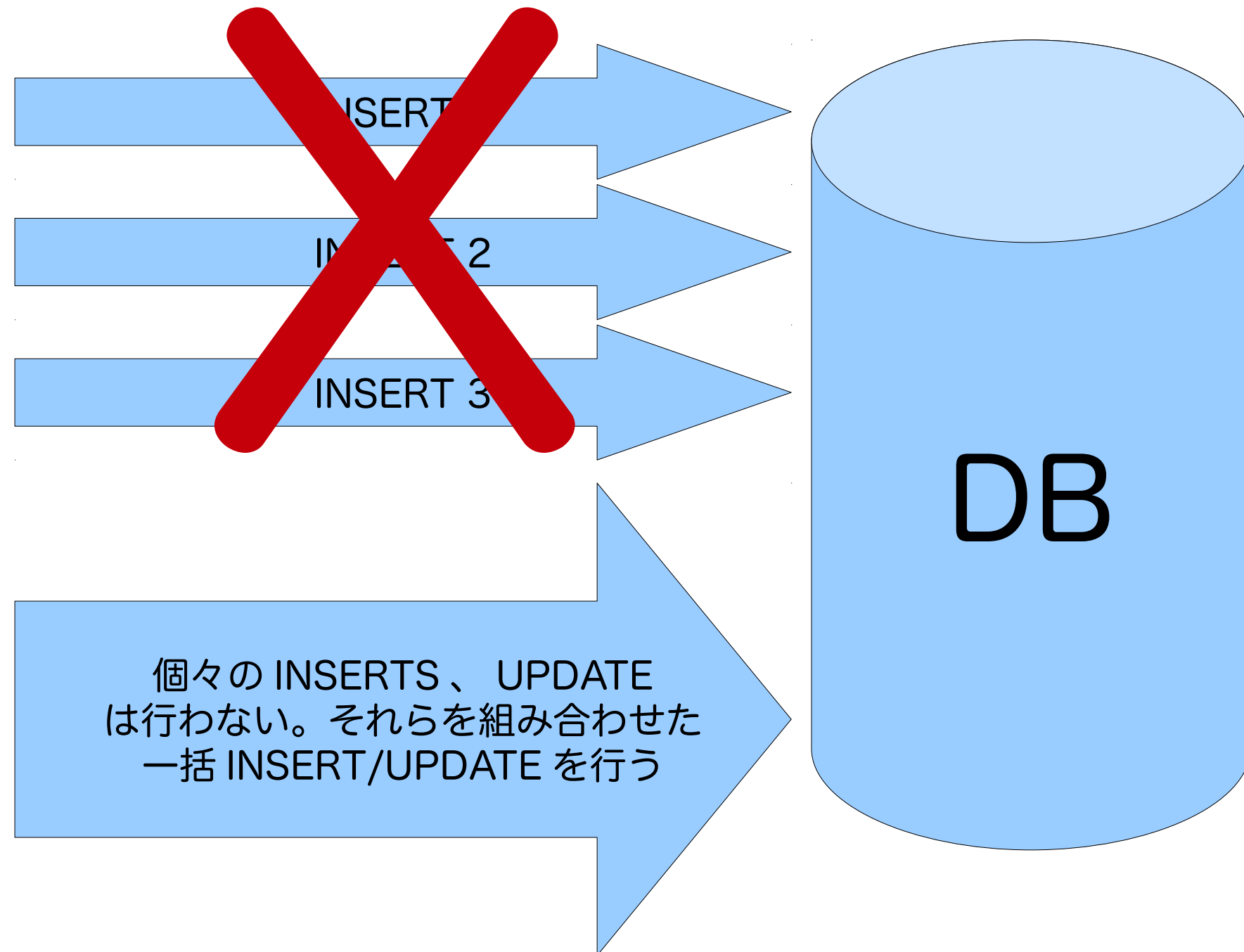
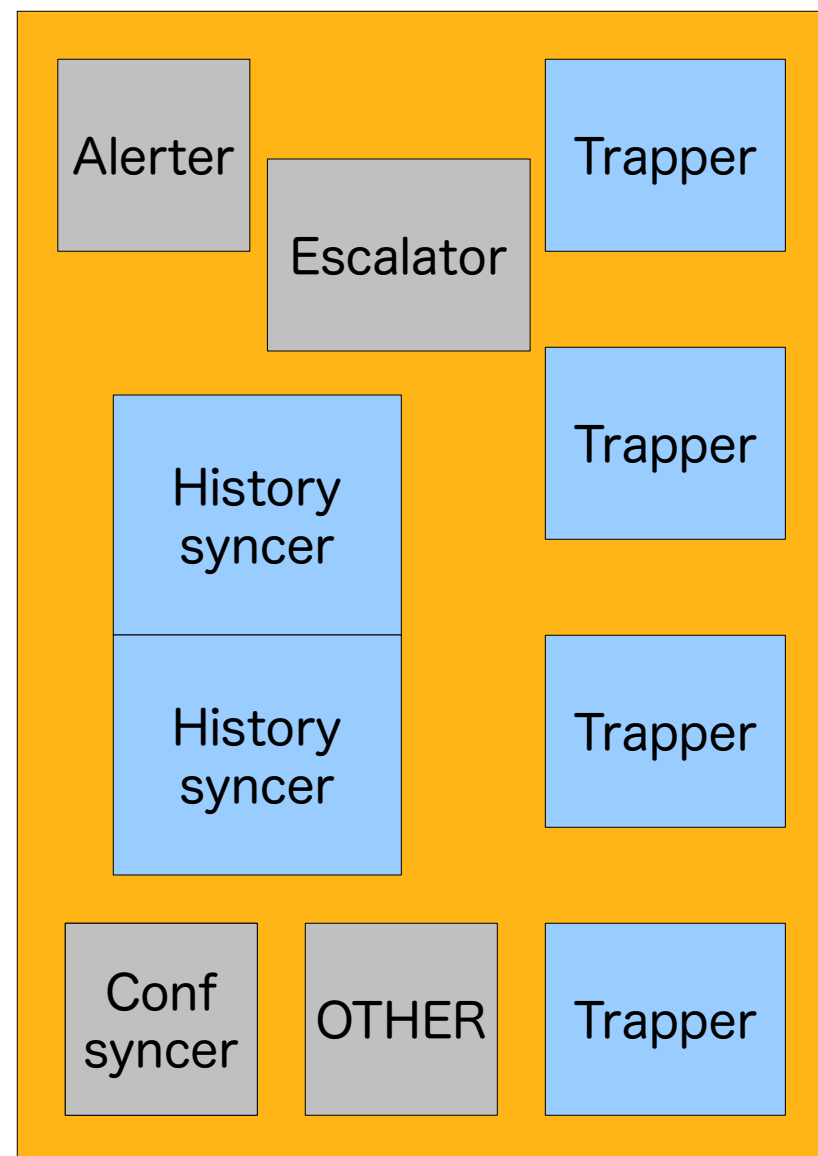
C,PHP,SQL が Zabbix に どう作用するか

- + 依存するものがほとんどない
- + メンテナンスが容易
- + 高パフォーマンス
- + 低リソース利用
- + すべての Unix プラットフォームに対応
- リグレッションと予期せぬ不具合 (undefined variables)



- バックエンドにのみ使用（サーバー、プロキシ）

いくつかのテクニック：一括操作



- バックエンドにのみ使用（サーバー、プロキシ）

アーキテクチャー：良い部分

- + 最適なロジック分割：
データ収集、障害検知、通知、可視化、API など
- + マルチ・プロセス・アプリケーション：
すべての使用可能な CPU コアにスケール
- + データが一貫した状態に常にある

1998 年には良い基盤でした

果たして今は？

現在の課題

コードの重複（バックエンド、フロントエンド）

```
alex@alex: ~
alex@alex: ~/public_html/2.4.0/src/libs/zbxdbhigh

*
* Function: validate_host
*
* Description: Check collisions between host and linked template
*
* Parameters: hostid      - [IN] host identifiicator from database
*             templateids - [IN] array of template IDs
*
* Return value: SUCCEED if no collisions found
*
* Author: Alexander Vladishev
*
* Comments: !!! Don't forget to sync the code with PHP !!!
*
*****/
static int validate_host(zbx_uint64_t hostid, zbx_vector_uint64_t *templateids,
                        char *error, size_t max_error_len)
{
    const char    *__function_name = "validate_host";
    DB_RESULT     tresult;
    DB_RESULT     hresult;
    DB_ROW        trow;
    DB_ROW        hrow;
    char          *sql = NULL, *name_esc;
    size_t        sql_alloc = 256, sql_offset;
    ZBX_GRAPH_ITEMS *gitems = NULL, *chd_gitems = NULL;
    size_t        gitems_alloc = 0, gitems_num = 0,
                chd_gitems_alloc = 0, chd_gitems_num = 0;
}
```

570,2

12%

現在の課題

異なるテクノロジー： バックエンドとフロントエンド

開発のスピード

不具合と品質

高品質維持： テスト vs より良いツール

昔ながらの PHP コード

パフォーマンス

Zabbix の改善したい 5 つのこと

... to start with

WEB インターフェース：現状

- ナビゲーションが効率的ではない：メニュー！
 - 基本的なワークフローでの多すぎるクリック
- “Click until die(死ぬまでクリック)”,
Lukas, Zabbix Conference 2014*
- まとまりのない情報
 - 監視 / 管理を完全に分離
 - ドロップダウン：メモリ使用率、パフォーマンス、ユーザビリティ

WEB インターフェース : UX

- ユーザビリティの改善
- ナビゲーション : オブジェクト中心に
 - ホストの選択 → ホストに関する全情報がワンクリック
- 情報は相互連携されるべき
- より早く (API のパフォーマンスにも関わる)

API: いくつかの事実

- 非常に遅くなる可能性がある
- 非常に多くの SQL クエリを生成
- 厳密なバリデーションを行っていない
- 貧弱なエラーレポート

API: より早く、より高信頼に

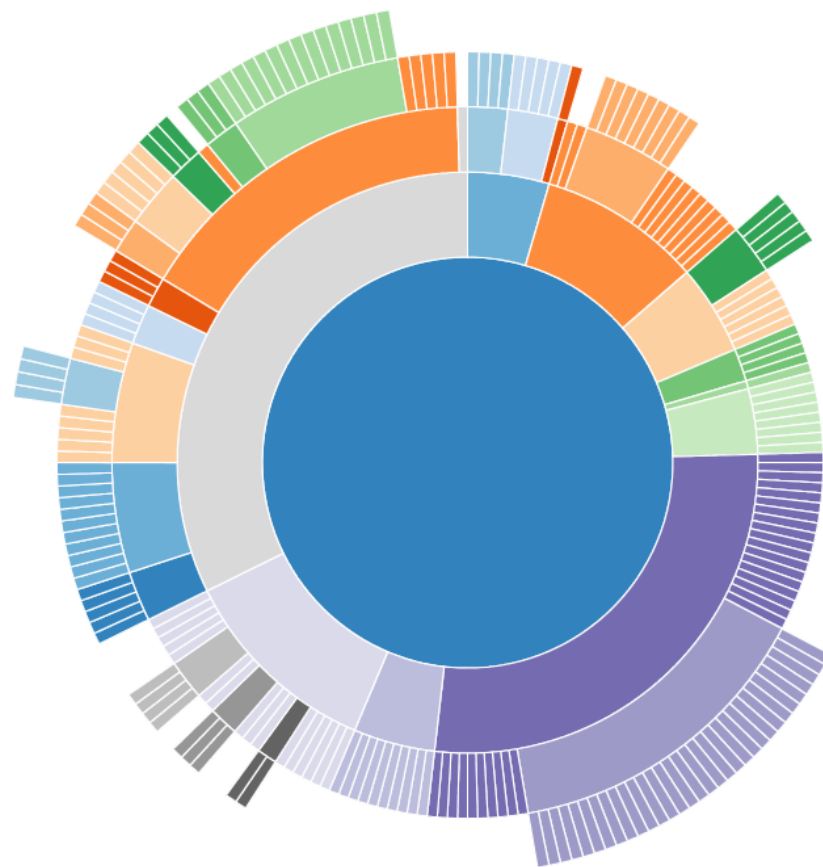
- 10-100 倍早くする
 - より効率のよいアルゴリズム
 - バルクオペレーション
- ファーストクラスオブジェクトの作成:
可能であれば Zabbix サーバー側に移動
 - 最も効率的なコードがある場所
- 厳密なバリデーションの実施
- エラーレポート
- 構成を可能に

レポート：現状

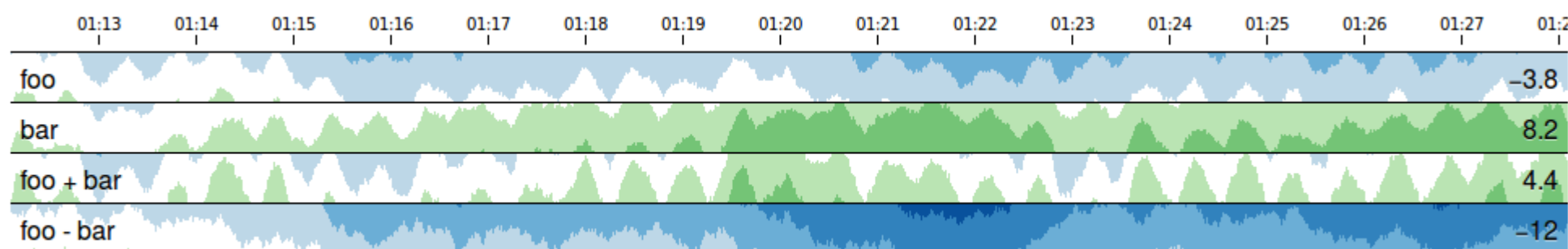
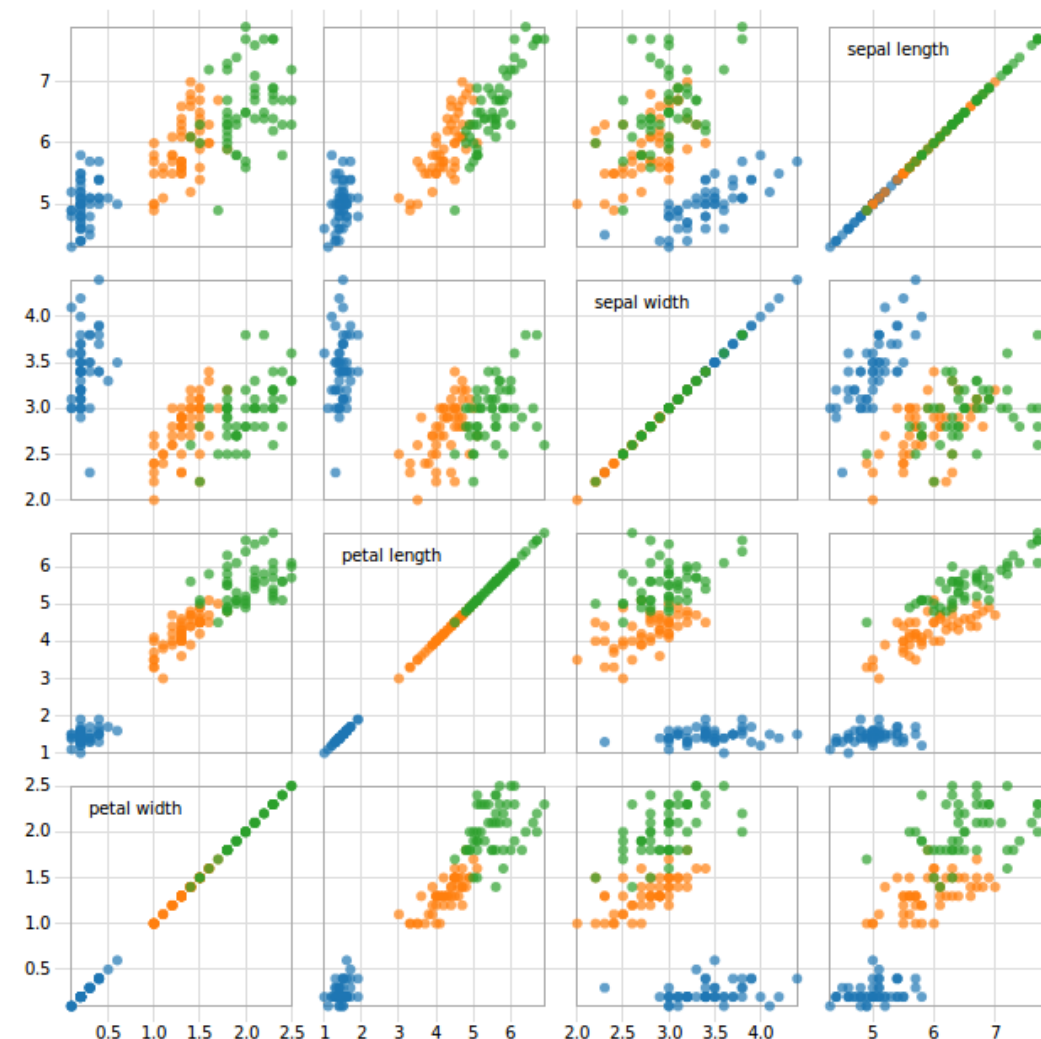
データベースには貴重なデータがたくさん、しかし…

- 非常に限られたレポート機能
- 分析機能がない
- アドホック・レポートを作る方法がない
- 実行したレポートのパラメータを保存する方法がない

レポート：可視化



- リアルタイム分析
- 重要：レスポンスタイムとスループット



スケーラビリティ：現状

- データ量が増えると Zabbix は（非常に）遅くなる
- スケールさせるために特殊な技術が必要
（データベース・パーティショニング）
- HA 構成 / 冗長構成を作るのが簡単ではない

- ストレージエンジンの水平方向への拡張
 - 標準のSQLエンジンではドライバは使用しない
- 履歴データは別ストレージに
- 新しい分散監視
- 秒以下の応答がフロントエンドの性能に重要

暗号化：現状

- 暗号化や認証は対応していない
- サードパーティーツールを使うことで実装可能 (stunnel、OpenVPN など)

暗号化と認証

- 製品の一部としてあるべき
- 有効にして維持することが容易でなければならない
- SSL/TLS 暗号化をエージェントに実装するかは未定

既存のツールと設計の決定に
満足をしてるか？

既存のツールと設計の決定に
満足をしてるか？
そうでもない

どんな変更が Zabbix 3.0
で実装されるか？

どんな変更が Zabbix 3.0
で実装されるか？

現在検討中

ロードマップは近日公開予定

Any questions?

:)