

WHITE PAPER:  
超高速データベース  
ZAP-IN  
ご紹介

VERSION 1.0

株式会社ターボデータラボラトリー

## 目次

0. 文書の概要 .....	1
1. 概要 .....	2
1-1. Zap-In Technology とは .....	2
1-2. Zap-In Technology の特長 .....	3
1-3. 応用例 .....	5
1-4. 他のデータベース技術との比較 .....	6
1-5. 4種類のシステム構成 .....	7
1-6. 特許 .....	8
2. 技術説明 .....	9
2-1. 超高速処理 .....	9
2-2. 他のデータベース技術との比較（速度） .....	11
2-3. 他のデータベース技術との比較（機能） .....	13
2-4. Zap-In Technology が超高速な理由 .....	14
2-5. 処理速度ベンチマーク .....	16
2-6. 自動プログラミング機能 .....	23
2-7. 高機能ビジネス・インテリジェンス・ツール .....	24
2-8. 高コストパフォーマンス .....	28
2-9. 技術詳細 .....	29
3. 用途：システム構築例 .....	34
3-1. 典型的なシステム構築例 .....	34

3-2. 導入事例.....	37
4. 製品紹介.....	43
4-1. Zap-In シリーズ AktblitzⅢ 製品概要.....	43
4-2. Zap-In シリーズ AktblitzⅣ（仮称）.....	45
5. 会社概要.....	46
5-2. 販売パートナー様.....	48

## 0. 文書の概要

- この文書は、株式会社ターボデータラボラトリーの超高速データベースシステム技術である Zap-In Technology について、その概要、特長、技術、用途、製品、会社概要を説明します。

## 1. 概要

### 1-1. ZAP-IN TECHNOLOGY とは

- Zap-In Technology は、従来比 10 倍～100,000 倍の速度を持つ超高速データベース技術です。
- 最大 320 億レコードのビッグデータを超高速で処理できます。
- その他の特長として、「超短期システム構築機能」「高機能ビジネス・インテリジェンス・ツール」「高コストパフォーマンス」「クライアント・サーバー構成対応」「クラスタ・サーバー構成対応」「超高速で高機能なインデックス」「並列処理」などを備えています。
- 日本国内で多数納入し、その効果に高評価をいただいています。
- 主な用途としては、「データ処理」「データ分析」「調達管理」「データ・ウェアハウス」などがあります。
- 上記の高性能・高機能は、独自開発の「成分分解法」理論に基づくデータ構造とアルゴリズムによるものです。
- Zap-In Technology は、SAP AG (独)、NEC、富士通ビー・エス・シーを始めとする各社へ技術ライセンスとして提供しています。

## 1-2. ZAP-IN TECHNOLOGY の特長

### 1. 圧倒的な超高速

- Zap-In は、超高速データベースと言われる Hadoop/Spark の 20 倍以上も高速です。
- Zap-In は、通常のデータベースシステムに比べては 10 倍～10,000 倍も高速です。JOIN 処理では 100,000 倍の速度になる場合もあります。ビッグデータになるとその差は大きくなります。
- 通常のデータベースシステムで例えば一昼夜かかっていた処理が、Zap-In では 1 分以内に終わる例もあります。これだけ速度を上げると、業務の質を格段に進化させることができます。  
この高速性と、ビッグデータ対応・高機能・インタラクティブ性によって、様々なビジネスプロセスを大きく進化させることができます。

### 2. 巨大ビッグデータ対応

- Zap-In は、シングルマシン構成で 20 億レコード、クラスタ構成では 320 億レコードの巨大ビッグデータを取り扱えます。
- 一般のデータベースはデータが大きくなると極端に低速になりますが、Zap-In は速度低下がゆるやかなのが特長です。

### 3. 超短期システム構築

- 自動プログラミング機能と高効率デバッグ過程によって、システム構築が超短期 (~1/10) に完成します。
- ユーザーのグラフィカル・ユーザーインターフェースによるデータベース操作をマクロ記録し、その操作をプログラムに自動変換します。この機能によって、プログラムを書くこと無く、データベースの構築が超短期で完了します。SQL プログラミングで 1 ヶ月かかったシステム構築が、3 日で完成出来た例もあります。

---

#### 4. 高機能ビジネス・インテリジェンス・ツール

- 今までにない高機能のビジネス・インテリジェンス機能を備えています。対話型バッチ処理・クレンジング・EXCEL 連携などワンストップで分析できます。

---

#### 5. 高コストパフォーマンス

- 同じ性能をはるかに小規模のハードウェアでローコスト(1/10~1/1000)に構築できます。

## 1-3. 応用例

### 1. ビッグデータ・分析システム

- 巨大な業務データを分析して業務修正対応・業務戦術立案などに役立てるシステムでは、以下の様な点が重要になります。このような用途に Zap-In は利用され、大きな成果を生み出しています。
  - ビッグデータが取り扱えること
  - ビッグデータを超高速で処理できること
  - 操作がしやすく、わかりやすいこと
  - 処理が・多機能であること
  - 処理を柔軟に変更できること

### 2. 超高速データベース

- 処理が高速であることが重要な利用分野では、Zap-In は他に替えがたいデータベースシステム製品です。特に、データ量が大きくて他システムでは処理速度に問題が出る場合でも、高速処理を提供できます。

### 3. データウェアハウス

- 蓄積した過去の大量のデータを、利用分析するにあたって、高機能・超高速のデータ処理機能が重要です。また、強力なインデックス機構と超高速性により、データマートなしですむのも大きなメリットです。

### 4. データクレンジング

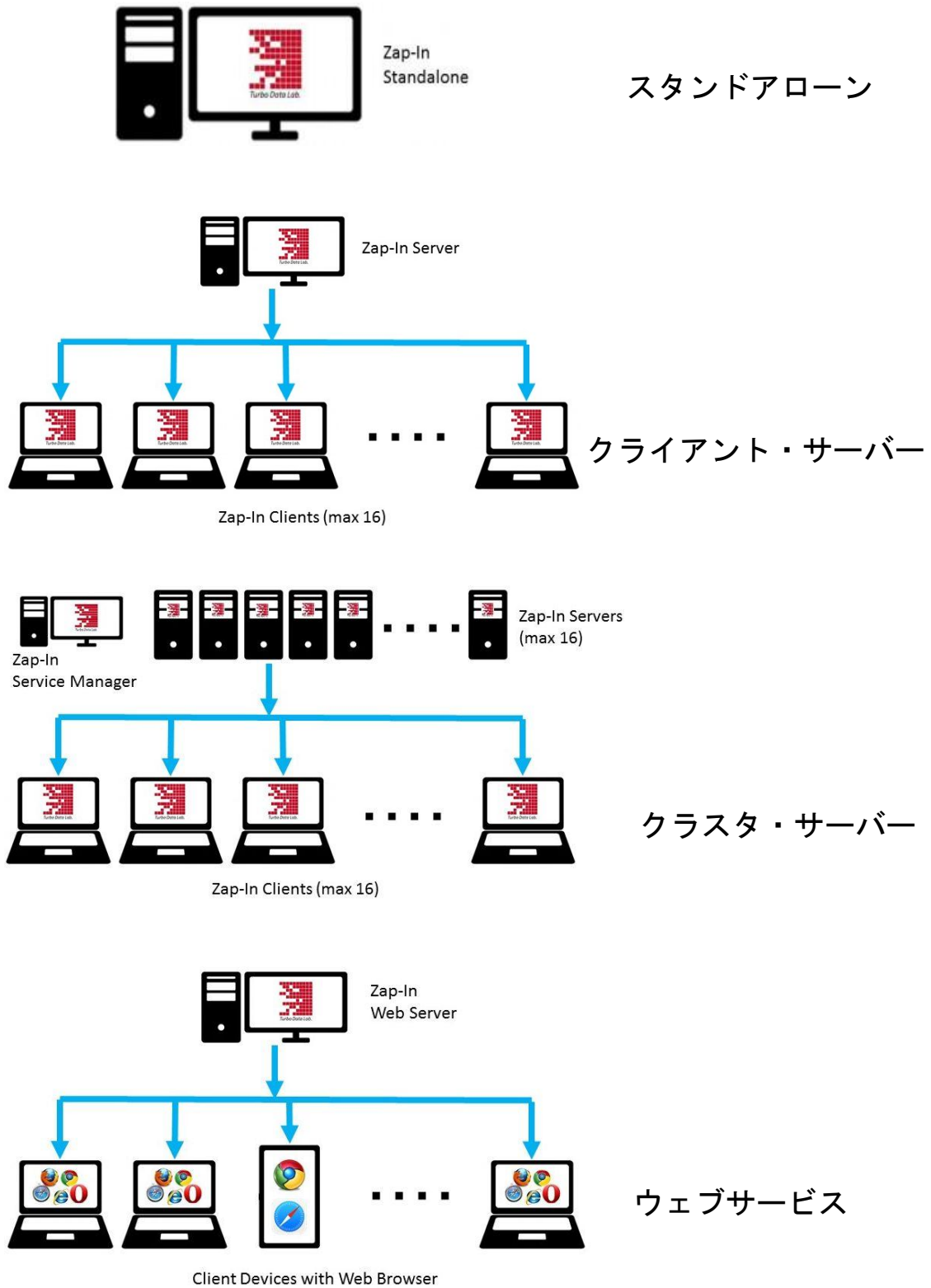
- Zap-In は、高速で多機能な強力なデータクレンジング機能を提供しています。



## 1-4. 他のデータベース技術との比較

評価項目	RDB(ディスク) 型 データベース	RDB(インメモ リ)型データ ベース	NoSQL 型 データ ベース	全文検索型 データ ベース	表計算 ソフト	<b>ターボ Zap-In</b>
速度	1	10	50	1-10	0.1	<b>10-1000,000</b>
機能	◎	◎	△	△	△	○
操作性	×	×	△	△	◎	○
価格				-	◎	○

1-5. 4種類のシステム構成



## 1-6. 特許

- ターボデータの超高速ビッグデータ技術は、日本国内・海外で特許化されています。
- 特許件数
  - Zap-In 関連特許
    - 日本国内 11件
    - 海外 26件
- 以下の各社へ技術ライセンスを提供しています
  - SAP AG (ドイツ)
  - 株式会社富士通ビー・エス・シー
  - 日本電気株式会社
  - 株式会社セック
  - 日本ソフト開発株式会社
  - ほか

## 2. 技術説明

### 2-1. 超高速処理

- Zap-In technology の一番の特長は圧倒的な超高速処理です。
- データベース処理が超高速
  - 通常のデータベースシステムに比べて、10倍から1,000倍、JOINの処理では10万倍の処理速度が得られます。
  - 超高速データベースの代表と言われる Hadoop/Spark と比べては20倍以上も高速です。
  - 独自開発技術の「成分分解法」のデータ構造とアルゴリズムにより、従来にない超高速なデータベース処理ができます。
- ビッグデータでも超高速
  - 最大20億行のビッグデータにも対応できます。データをハードディスクではなくメインメモリに保持するインメモリ型データベースですが、高効率データ構造によって巨大なデータにも対応できます。  
クラスタ構成ならば16台のサーバーを用いて、総計320億行の巨大ビッグデータに対応できます。
- データ量と処理時間が比例関係
  - 通常のリレーショナルデータベースシステムでは、データ量が増えると処理時間は急激に長くなる基本性質があります（データ量  $n$  に対して  $o(n \cdot \log(n))$ ）。このため、データ量が巨大になるビッグデータでは処理時間が非常に長くなってしまいます。  
これに対して Zap-In Technology では、処理時間はデータ量に比例になります（データ量  $n$  に対して  $o(n)$ ）。ビッグデータでは、非常に大きな差がつかます。
- 高速なデータ取り込み

- CSV フォーマット等の元データファイルを取り込むスピードは、一般のデータベースよりも桁違いに高速です。独自データ構造による高速インデックス生成と、マルチコア CPU による並列処理によります。

## 2-2. 他のデータベース技術との比較（速度）

- 現在の世の中で使われている各種のデータベースシステムと Zap-In Technology との、速度の比較です。

処理	RDB(ディスク)型データベース	RDB(インメモリ)型データベース	NoSQL 型データベース	全文検索型データベース	<b>ターボ Zap-In</b>	(注釈: Zap-In の特徴)
CSV データ取り込み	1	10	50	0.05	<b>100</b>	
JOIN 演算	1	10	-	-	<b>1000-100,000</b>	カーディナリティが大でも高速
SORT	1	10	-	-	<b>100-100,000</b>	カーディナリティが大でも高速
SEARCH	1	10	100-1000 (キー項目完全一致の場合)	-	<b>10-1000</b>	データ量が大でも高速
BOM 展開	1	10	-	-	<b>500-700</b>	
カテゴリ化	1	10	-	-	<b>10-1000,000</b>	特に高速 O(n)
集計	1	10	-	-	<b>1000-100,000</b>	カーディナリティが大でも高速

計算・更新	1	10	-	-	0.1-10,000	1件ずつ更新は低速、まとめて更新は超高速
EXPORT	1	10	1 (キー項目完全一致データ1件の場合)	1-10 (ヒットしたドキュメントの取り出し)	100-1000	ヒット件数が大なら特に高速
全文検索	1	10	-	100-10000	10-1000	ヒット件数が大でも高速

## 2-3. 他のデータベース技術との比較（機能）

- 現在の世の中で使われている各種のデータベースシステムと Zap-In Technology との、機能の比較です。
  - **Zap-In（インメモリ型超高速データベース）**
    - 特徴： 超高速で、ほぼ RDB 同等の高機能。高機能で高速な全文検索も持つ。トランザクション機能は持たない。
  - **RDB(ディスク)型データベース**
    - 製品例： MySQL, OracleDatabase
    - 特徴： 現在のデータベースの主流。多機能だがビッグデータでは速度低下。
  - **RDB(インメモリ)型データベース**
    - 製品例： HANA(SAP 社), TimesTen(Oracle 社), Spark(Apache)
    - 特徴： RDB(ディスク)より高速(10 倍程度)。
  - **NoSQL 型データベース**
    - 製品例： DynamoDB(Amazon 社), Cassandra(Apache)
    - 特徴： 高速だが機能が少ない。
  - **全文検索型データベース**
    - 製品例： SecureEnterpriseSearch(Oracle 社), Namazu
    - 特徴： 全文検索の単機能だが高速。
  - **バッチ処理型データベース**
    - 製品例：
    - 特徴： プログラミングが手軽で容易だが低速。



## 2-4. ZAP-IN TECHNOLOGY が超高速な理由

### ZAP-IN TECHNOLOGY はインメモリ・データベースの最進化系

- データをハードディスクではなく全てメインメモリに置き、その高速性を活かすのがインメモリ型データベースです。インメモリ型データベースは各社から提供されています。データ構造や処理アルゴリズムが従来のもので、インメモリ型にすればそれなりに高速なデータベースになります(~10倍程度)。しかし、これだけではメインメモリの高速性を十分に活かし切っているとは言えません。
- ターボデータラボラトリーは、データ構造と処理アルゴリズムを根本から考え直してインメモリ型データベースを高速化しました。独自開発の成分分解法(\*)に基づくアルゴリズムとデータ構造を基本にしたことで、従来比10倍~10,000倍、処理によっては100,000倍の高速性が得られています。また、このデータ構造により、最大20億レコード(クラスタ版では160億レコード)のビッグデータを取り扱うことができます。

### 成分分解法 (LFM)

- ターボデータは、インメモリ処理の有効性をさらに大きく活用できる独自の「成分分解法」を開発しました。さらに、それに基づく新しいデータ構造と処理アルゴリズムを用いて、超高速データベース技術 Zap-In Technology を完成させました。成分分解法とそのアルゴリズムは、世界各国で特許化されています。

### マルチコア GPU を効率的に活用

- 現在のPCには、演算コアが4~16機も搭載されたマルチコアCPUが使われています。この複数の演算コアを有効に活用すれば、処理速度を大きく向上させることができます。このためには、データベース処理を並列に処理するようなソフトウェアにする必要があります。  
しかし、一般のデータベースシステムでは、データ構造やアルゴリズムの点で、処理を並列化することが一般的には困難で、マルチコアを有効に活用でき

ません。

それに対して成分分解法では、そのデータ構造とアルゴリズムの性質から、ほとんどのデータベース処理を並列処理化し高速に行えます。Zap-In Technology では、マルチコアを最大限に活用して、データベース処理ばかりでなくデータ取り込みなどの処理でも超高速を実現しています。

---

## 全項目インデックス

- インデックス設計はシステム構築時のデータベース高速化の鍵です。従来のデータベースシステムでは、効率的で高速な処理をするために時間をかけてインデックス設計をする必要があります。データベースのどの項目にインデックスをつけて高速化するかが重要です。最適なインデックス設計をしても十分な速度が得られないことも多くありますし、運用開始後に新機能が必要になった場合にはインデックス設計を全面やりなおさないと十分な処理速度が得られないこともあります。

Zap-In Technology では、成分分解法の基本的性質から、全項目にインデックスが付いており、インデックス付加のための処理も不要です。このため、どの項目に対しても高速なデータベース処理を行うことができます。また、システム構築時にはインデックス設計が不要ですので、開発が短期間に行え、しかも運用開始後に新機能が必要になった場合にも、インデックス再設計が不要なため、素早いシステム改良ができます。

---

## アルゴリズム

- Zap-In Technology では、成分分解法による超高速データベース処理を実現していますが、この理論を実際のソフトウェア製品にするために、成分分解法を使いこなすためのさまざまなアルゴリズム・周辺技術・ソフトウェア技術が駆使されています。これらも、成分分解法とともに各国で特許化されています。

## 2-5. 処理速度ベンチマーク

### 2-5-1. HADOOP/SPARK と ZAP-IN TECHNOLOGY との速度比較

#### 概要

- 大規模データ高速処理エンジンとして有名な [Hadoop/Spark](#) との処理速度の比較をしました。
- データ読み込み・検索・ジョイン・集計・データ書き出しの各処理において、Zap-In は 最大で 181 倍、合計タイムで 25 倍の速度を記録しました。この差は、データが大きくなればさらに劇的に拡大するものです。

#### 処理マシン

- - CPU -----  
Intel (R) Xeon (R) CPU E5-2403 v2 @ 1.80GHz  
cpu MHz : 1201.148  
cache size : 10240 KB  
fpu : yes  
fpu\_exception : yes
- - Memory -----  
MemTotal: 32726272 kB  
MemFree: 28515008 kB  
MemAvailable: 30948488 kB  
Buffers: 884 kB  
Cached: 2568612 kB  
SwapCached: 0 kB
- - OS -----  
CentOS Linux release 7.1.1503 (Core)

#### データ

- ・ A 営業マンマスター.csv 1,000 件
- ・ B 製品マスター.csv 100 件
- ・ C セールスログ.csv 6,070,000 件
- ・ D エクスポート.csv 73,000 件

### 処理課題

- 1. データインポート
- 2. サーチ
- 3. ジョイン
- 4. 集計
- 5. データエクスポート

処理		Zap-In	Hadoop/Spark	速度比
データインポート	1. A データ	13 ms	2,352 ms	x181
	1. B データ	2 ms	150 ms	x75
	1. C データ	1,788 ms	24,424 ms	x14
処理	2. サーチ	19 ms	495 ms	x26
	3. ジョイン	658 ms	234 ms	x0.4 (*)
	4. 集計	125 ms	208 ms	x1.7
データエクスポート	5. D データ	1,586 ms	78,973 ms	x50 (*)
合計タイム		4,192 ms	106,839 ms	x25

---

## 2-5-2. BIGQUERY と ZAP-IN TECHNOLOGY との速度比較

---

### 概要

- クラウド上の大規模データ分析システムとして有名な [BigQuery](#) との処理速度の比較をしました。
  - データ読み込み・検索・ジョイン・集計・データ書き出しの各処理において、Zap-In は 最大で 826 倍、合計タイムで 27 倍の速度を記録しました。この差は、データが大きくなればさらに劇的に拡大するものです。
- 

### 処理マシン

#### BIGQUERY 側

---

- Google クラウド

#### ZAP-IN 側

---

- - CPU -----  
Intel (R) Xeon(R) CPU E5-2403 v2 @ 1.80GHz  
CPU Clock : 1201.148 MHz  
cache size : 10240 KB  
fpu : yes  
fpu\_exception : yes
- - Memory -----  
MemTotal: 32726272 kB  
MemFree: 28515008 kB  
MemAvailable: 30948488 kB  
Buffers: 884 kB  
Cached: 2568612 kB  
SwapCached: 0 kB
- - OS -----  
CentOS Linux release 7.1.1503 (Core)

---

## データ

- A 営業マンマスター.csv 1,000 件
  - B 製品マスター.csv 100 件
  - C セールスログ.csv 6,070,000 件
  - D エクスポート.csv 73,000 件
- 

## 処理課題

- 1. データインポート
- 2. サーチ
- 3. ジョイン
- 4. 集計
- 5. データエクスポート

処理		Zap-In	BigQuery	速度比
データインポート	1. A データ	13 ms	1,000 ms	x77
	1. B データ	2 ms	1,000 ms	x500
	1. C データ	1,788 ms	71,100 ms	X40
データ処理	2. サーチ	19 ms	15,700 ms	X826
	3. ジョイン	658 ms	10,200 ms	x15
	4. 集計	125 ms	3,000 ms	X24
データエクスポート	5. D データ	1,586 ms	12,000 ms	X8
合計タイム		4,192 ms	114,000 ms	X27

---

### 2-5-3. ZAP-IN TECHNOLOGY の単独の速度測定

---

#### 概要

- Zap-In のカテゴリライズ処理速度の測定をしました。
  - 10億件のデータを10種類のカテゴリに分類するのにわずか 187 ms で完了しました。
- 

#### 処理マシン

- - CPU -----  
AMD Phenom2 CPU x4 925(4core) @ 2.80GHz  
CPU Clock: 1201.148 MHz  
L1 : 2MB  
L2 : 6MB
  - - Memory -----  
MemTotal: 8 MB (1333MHz)
  - - OS -----  
Windows 7 Ultimate 64bit
- 

#### データ

- データ件数 1億件
  - 各データの項目数 8項目
- 

#### 処理課題

- 10種類のカテゴリに分類
- 

#### 結果

- 処理時間 187 ms (5.35億更新/秒)

---

## 2-5-4. ZAP-IN TECHNOLOGY の単独の速度測定

---

### 概要

- Zap-In の検索・ソート・集計・上書き更新の各処理速度の測定をしました。
- 例えば、1億件の文字列データをソートするのに、わずか 5,659 ms で完了しました。

---

### 処理マシン

- - CPU -----  
Intel Xeon x2 (2core) @ 3.16GHz  
L1 : 64kB  
L2 : 1MB/1core
- - Memory -----  
32 MB (1333MHz)
- - OS -----  
Windows Server 2008 Standard

---

### データ

- データ件数            1億件
- 各データの項目数    8項目
  - 項目1 : 整数, 1億種類の値, シーケンシャル
  - 項目2 : 整数, 1万種類の値, ランダム
  - 項目3 : 整数, 100種類の値, ランダム
  - 項目4 : 整数, 100種類の値, ランダム
  - 項目5 : 文字列, 1億種類の値, ランダム
  - 項目6 : 文字列, 100種類の値, ランダム
  - 項目7 : 倍精度浮動小数, 1億種類の値, ランダム
  - 項目8 : 10進固定小数(38桁), 1億種類の値, ランダム

---

### 処理課題



- 処理 1. サーチ： 文字列（ユニークでランダム）をサーチ
- 処理 2. ソート： 項目 5（ユニークでランダムな文字列）をソート
- 処理 3. 集計： 1次元（100種類の値）、1測度（ユニークでランダムな浮動小数）の集計
- 処理 4. 上書き更新： 100万ヶ所を上書き更新

---

## 結果

処理（データ 1 億件）	Zap-In
1. サーチ 10 件ヒット	0 ms
1. サーチ 1 万件ヒット	0 ms
1. サーチ 100 万件ヒット	7 ms
2. ソート	5,659 ms
3. 集計	9,312 ms
4. 上書き更新 100 万件	10,784 ms

## 2-6. 自動プログラミング機能

### 超短期でシステム構築

- Zap-In Technology は、マクロ記録による自動プログラミング機能を提供しています。

表計算型のグラフィカル・ユーザーインターフェースをマウスで操作してデータベース処理を行わせると、同時にその操作を記録し、Python のプログラムに変換します。これで所要の操作のデータベース処理プログラムが出来たこととなります。こうしてプログラミングを行わずにシステム構築ができることとなります。操作を記録してプログラムに変換することは、Python が手続き型言語であるために可能になりました。

- さらに、その操作でのデータベース処理は 超高速で行われてすぐに結果が表示されますので、その操作が正しかったかどうかその場で分かり、デバッグも短時間で高い効率で行うことができます。

実例としては、通常のデータベースシステムでは 1 ヶ月以上かかったシステム構築が、Zap-In Technology では 3 日で完成した例もあります。

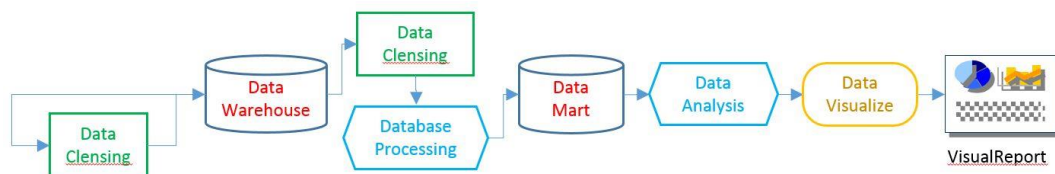
### PYTHON を採用

- 通常のリレーショナルデータベースでは、システム構築時のプログラミングに SQL 言語を用いるのが一般的です。SQL 言語は非手続き型言語なので、ループが書けずプログラミングが難しい、変数が露出しないためデバッグに手間取りやすいという問題がありました。
- Zap-In Technology では、手続き型言語である Python を採用し、上記の問題を解決しました。Python は、高いプログラミング効率と信頼性を備え、大規模な標準ライブラリによる利便性も備えています。  
ジョブスケジューラから Python プログラムを呼び出すこともできます。

## 2-7. 高機能ビジネス・インテリジェンス・ツール

### 2-7-1. ビジネス・インテリジェンスとは

- ビジネス・インテリジェンスとは、社内データベースに蓄積されたデータを企業の意思決定に活用するためのものです。膨大なデータから必要なデータを検索、多次元分析して有用な情報を取り出しあるいはデータマイニングして、理解しやすいグラフや表でレポートにまとめ、状況把握や意思決定を支援します。
- データ検索や分析の過程では、有用な情報を取り出すために検索や分析の方法をいろいろに試してみる作業が重要になります。このために、システムとしては操作性・多機能性・処理速度などが重要になります。
- このような分析のためには、元データを整理整形（データクレンジング）して、巨大データベース（データウェアハウス）に蓄積し、その中から分析に必要なデータの加工抽出処理を構築して使いやすいサイズのデータベース（データマート）に収めておきます。それをいろいろな手法で高速にデータ分析します。



### 2-7-2. 上記の分析の部分のシステム構築には、以下の手法があります。

- **一般のデータベースシステムを利用**  
データ分析が多機能で高速だが、分析のプログラミングが必要なので専門家が必要。
- **ビジネス・インテリジェンス・ツール (BI ツール) を利用**  
グラフィカルな操作で分析できるが、扱えるデータ量が限られ、分析機能や速度も高くない。

- **表計算ソフト（EXCEL 等）を利用**  
手軽に分析できるが、扱えるデータ量が限られ、分析機能や速度も高くない。
- **Zap-In Technology を利用**  
データ分析が多機能で高速。表計算ライクな操作で分析ができる。大規模データでも高速なのでデータマートが不要。クレンジング機能も強力。



## 2-7-3. ビジネス・インテリジェンス機能の比較

処理	一般データベースシステム	ビジネス・インテリジェンス・ツール	表計算ソフト (EXCEL 等)	<b>Zap-In Technology</b>
クレンジング機能	ある	なし	なし	<b>強力</b>
データベース分析機能	強力	値の種類が多いと対応できない	JOIN 処理などができない	<b>強力</b>
操作	SQL を用いる (専門家による)	グラフィカルで容易	表形式で容易	<b>表形式で容易</b>
処理プログラミング	SQL 言語プログラミング (専門家による)	不要 (または専門家による)	不要 (またはある程度できる)	<b>自動プログラミング (専門家は不要)</b>
処理速度	高速	低速	低速	<b>超高速</b>
大規模データ	対応できる	対応できない	対応できない	<b>対応できる</b>
データマート	大規模 データでは必要	必要	必要	<b>大規模データでも不要</b>
視覚化・グラフ化	可視化ツールを利用	強力	強力	<b>EXCEL に接続して表示</b>

---

#### 2-7-4. ZAP-IN TECHNOLOGY のビジネス・インテリジェンス機能

Zap-In Technology は、優れたビジネス・インテリジェンス機能を提供しています。

- **ワンストップ処理で高い業務効率**

クレンジング・データ分析・分析結果可視化の流れのすべてがスムーズに出来ます。

データ分析で必須となるクレンジング処理は、多くのビジネス・インテリジェンス・ツールが苦手としていますが、Zap-In Technology は多機能・高速なクレンジング処理を持っています。データベース処理が超高速でビッグデータ対応できるので、データマートが不要です。分析結果を EXCEL ヘススムーズに連携する機能を持っていますので、EXCEL の豊富な機能を利用してさまざま可視化を行うことが出来ます。

- **ビッグデータでも対話型処理で高い業務効率**

一般のビジネス・インテリジェンス・ツールは、ビッグデータでは対話型とは言えなくなるほどに処理時間が長くなり、業務効率が低下してしまいます。

Zap-In Technology のビジネス・インテリジェンス機能は、ビッグデータでも超高速に対話型で処理ができ、高い業務効率が得られます。

- **データ操作機能が充実し、しかも高速**

一般の BI ツールが苦手とするデータ変換（単位換算、カテゴリ割付など）も高速処理ができます。ファクト・ディメンションテーブルの関係を用いたデータの転送も瞬時にできます。Excel ライクな Formula も、カテゴリイズも、型変換も超高速です。

また、一般のビジネス・インテリジェンス・ツールはグラフで視覚化して対話型処理操作を行います。例えば1万種の氏名が入っている顧客名簿データをグラフ化することは細かくなりすぎて現実的ではありません。グラフ内の項目をクリックできません。Zap-In Technology (AktblitzⅢ)は、対話型処理操作に表形式表示を採用し、高速で使いやすい環境を提供しています。

## 2-8. 高コストパフォーマンス

同じ性能をはるかに小規模のハードウェアでローコスト (1/10~1/1000) に構築できます。

## 2-9. 技術詳細

### 2-9-1. 成分分解法について

#### 1. 成分分解法の概要

成分分解法 (Linear Filtering Method, LFM) は、ターボデータラボラトリーが開発した革新的なデータベース技術です。多数の特許を日本国内・海外で取得しています。

元来、データベースシステムに対しては以下の性能・機能向上の要求がありました。

- ・ データを安全に更新する
- ・ 桁違いに大量のデータを扱える
- ・ 桁違いに高速に処理する
- ・ アナリティクスにも適用できる
- ・ 複雑な計算処理を行うバッチ処理に適用できる
- ・ 分散環境にも対応できる

成分分解法はこれらに対して、最も優れた解であると言えます。

#### 2. 成分分解法の長所

成分分解法は以下の長所を持っています。

##### 1. 計算量が $O(n)$

ほとんどすべてのアルゴリズムの計算量を  $O(n)$  に引き下げることができます。(n はレコード数) 従来のデータベースシステムでは  $O(n \cdot \log(n))$  ですから、オーダーの違う高速性を実現します。これはビッグデータを扱う際にきわめて有利な特性です。

##### 2. 高機能・高速なインデックス

データベースの高速化・効率化の鍵となるインデックスに関するさまざま課題を解決しました。何段階も処理を重ねても有効なインデックスを実現しました。BOM 展開などの多段階処理では圧倒的な高速性を発揮します。

##### 3. 並列処理



ほとんどすべてのデータベース処理を並列化できます。近年では普通になったマルチコア CPU の並列処理機能を最大限に活かして、処理を何倍にも高速化できます。

---

## 2-9-2. 成分分解法によるインデックス

---

### 1. データベースのインデックス概要

データベースの性能を上げるための、最も一般的で効果的な手法がインデックスです。一般のデータベースシステムでは、「ハッシュ」あるいは「B-Tree」によるインデックスが良く使われます。

データベースの項目にインデックスを付けると、その項目に関する検索等の速度は大幅に向上します。一方、その項目のデータを変更・更新した場合には、そのインデックスの更新が必要になるため、速度低下の原因になります。そのため、インデックスを付加する項目は少数にしておくのが一般的です。

データベースを構築するときには、このインデックス設計が性能を上げるために非常に重要です。どの項目にインデックスを付けるかを、処理のバランスを考慮して決定する必要があります。後に処理の内容を変更する場合には、インデックス設計をやり直すことになり、長期の再開発期間が必要になります。

また、ある項目をデータベース処理した結果に得られた新しい項目にはインデックスは付いていませんので、必要であればその処理の後でインデックスを付ける必要があります。

---

### 2. 成分分解法によるインデックス

Zap-In/Zap-Over で採用している成分分解法では、インデックスがデータ構造に自然な形で組み込まれていますので、すべての項目にインデックスが付いています。このため、すべての項目についての処理を高速に行うことができます。また、インデックス設計が不要になるので、データベース構築が効率的に行えます。

また、データベース処理した結果で新しい項目が生成されても、その新しい項目にもインデックスが自動的に付いています。このため、その後の処理も高速に行われます。インデックス再設計も必要ありません。

---

### 3. 各方式の比較

「ハッシュによるインデックス」「B-Treeによるインデックス」「成分分解法」の3つについて、データベース処理の速度の比較を表にしました。

各処理で、成分分解法が非常に優れた特性を持ち、特にデータ量が大きくなった場合には非常に大きな優位性があることがわかります。また、成分分解法ではマルチコア CPU によるマルチスレッドが効果的に作用することも示されています。

### インデックス方式の比較

インデックス方式	ハッシュ	B-Tree	成分分解法 (pは並列スレッド数)
インデックス付加する項目	主キーのみ	少数	全項目
インデックス生成時間	小さくない	大きい	初回のみ
必要なメモリ量	$O(n)$	$O(n \cdot \log(n))$	なし
SEARCH	$O(1)$ 完全一致、1件のみ	$O(m \cdot \log(n))$ mはヒット件数	$O(n/p)$ : デフォルト時 $O(\log(n)/p)$ : 高速化時
SEARCH (部分集合内) sは部分集合のサイズ	$O(s)$ 完全一致のみ	$O(s) + O(m \cdot \log(n))$	$O(s/p)$
SORT	効果なし	$O(n \cdot \log(n))$	$O(n/p)$
SORT (部分集合内) sは部分集合のサイズ	効果なし	$O(n \cdot \log(n))$	$O(n/p)$

UNION n1, n2 はテーブルのサイズ	効果なし	効果なし	$o((n1+n2)/p)$
UNION (部分集合) s1, s2 は部分集合のサイズ	効果なし	効果なし	$o((s1+s2)/p)$
JOIN n1, n2 はテーブルのサイズ	$O(n1)$ (SEARCH JOIN)	$O(n1*\log(n2))$ (SEARCH JOIN)	$o((n1+n2)/p)$ (SORT JOIN)
JOIN (部分集合) s1, s2 は部分集合のサイズ	効果なし	効果なし	$o((s1+s2)/p)$ (SORT JOIN)
集計	効果なし	効果なし	$o(n/p)$
集計 (部分集合)	効果なし	効果なし	$o(n/p)$
インデックス更新	逐次的	面倒	不要

### 3. 用途：システム構築例

#### 3-1. 典型的なシステム構築例

- Zap-In は単独で、あるいは他システムと組み合わせて、超高速なデータベース処理機能を提供します。

Zap-In の特長と高性能・高機能を活かす典型的なシステム構築例をご紹介します。

---

#### 1. データ分析用の超高速データベースに応用

- 通常のデータベースシステムでは丸一日かかるような処理が、Zap-In なら数分で完了するかもしれません。数十分かかっていた処理が数秒で完了するようになるかもしれません。そうなった時の業務効率アップはどれ程のものになるでしょうか？

月次レポートが日次レポートになれば、業務分析・経営判断で驚くほどの進化があるでしょう。分析データを得るまでに数十分も待っていたのが数秒になれば、アナリストの業務効率も劇的に向上するはずです。

---

#### 2. ビッグデータシステム

- データが巨大になると一般のデータベースシステムは処理速度が著しく低下します。Zap-In なら、処理速度低下のために諦めていたビッグデータベースシステムを構築できます。

---

#### 3. データクレンジングに応用

- データクレンジングとは、データベースの中の重複・誤記・表記の揺れなどを探し出し、削除・修正・正規化して、データベースの品質を高めることです。データベースシステムの運用には不可欠なものです。
- Zap-In は、高速で多機能な強力なデータクレンジング機能を提供しています。データをチェックしながら対話的に素早く作業を進めるインタラクティブ・データ・クレンジング、定常的に全自動で処理を行うオートマチック・クレンジングのどちらにも対応しています。

---

#### 4. データウェアハウスに応用

- データウェアハウスは、過去の大量のデータを分析して意志決定に役立てるための、データを整理して蓄積・利用する「データの倉庫」です。大量のデータに対して、さまざまなデータ処理ができることが要求されます。
- Zap-In と一般データウェアハウスシステムの比較
  - 一般のデータウェアハウスシステム
    - ハードウェア： 多数のモジュール（CPU やストレージ）で構成する専用システム
    - 価格： 高価
    - 複雑な処理の性能： 複雑な処理では性能低下（多数のモジュール間のデータ参照等）
  - Zap-In によるデータウェアハウス
    - ハードウェア： 汎用の PC サーバーを利用できる
    - 価格： 安価（データウェアハウスの 1/10 以下）
    - 複雑な処理の性能： 高速に処理できる

---

## 5. データ解析に応用

- Zap-In は、ビッグデータのデータ解析に威力を発揮します。
- 表計算ソフトは、表形式でデータ操作がやりやすいが、大規模データを扱えず・JOIN 等の処理ができない欠点があります。BI ツールはグラフ化が容易ですが、高機能なデータ処理は持っていません。一般データベースでは処理にプログラミングが必要です。

Zap-In は、表形式で高機能なデータ操作・大規模データ対応・超高速処理・自動プログラミング機能を備え、ビッグデータの分析・解析に威力を発揮します。表計算ソフトと連携できますので、処理結果配布やグラフ化にも対応できます。

---

## 6. ウェブサービスのデータベースサーバーに応用

- Eコマースなどのウェブサービスでは、大量の商品データを取り扱うデータベースサーバーが必要になります。この処理速度・ターンアラウンドタイム・高機能が、ウェブ

サービスの鍵になります。高機能な商品サーチ機能などを提供することが必須なサービスでは、顧客へのレスポンスタイムが長くなってしまいがちです。

- Zap-In をウェブサービスのデータベースサーバーに採用すれば、その超高速性・高機能性が非常に有効に活躍します。

## 3-2. 導入事例

### 1. 全国規模フランチャイズチェーン企業 A 社様

#### 売上、在庫管理システム

- システム概要
  - 移り変わりの激しい商品を迅速に把握し、流行の立ち上がり時に商品を手配し、各支店に配布。余った商品は売却。売却時の値段の決定も行う。このようにして最適な商品を最適な量、各支店に配置する。
- 月次バッチ処理時間を数十分の一に短縮できた
  - 従来は20日間程かかっていた月次処理を、6時間に短縮できた。
  - 月次報告書を短時間で発行できるようになったため、問題対応をすばやく行えるようになった。
  - 各担当者の意思決定スピードが向上した。
  - 早期に買い付けが必要な流行しつつあるタイトルの、流行の立ち上がりを見逃さなくなった。
  - 貸し出されないのに棚を占有していた不要在庫を早期に処分できるようになった。
- 開発期間と費用を非常に少なくできた
  - 処理対象の実データを使ったインタラクティブな開発(Zap-Inの自動プログラミングの機能による)ができたため、間違いのない開発が短期間に完了できた。
  - 既存ハードウェアを有効利用して、類似のバッチ処理へ展開中。

### 2. 化学工業品製造会社 B 社様

#### 4工場の製造原価管理システム

- システム概要



- 化学工業品の製造原価管理は、化合物の BOM (bill of materials) 計算が複雑で時間がかかりすぎるため、ラフ計算しか行うことができなかった。
- 計算の精密化ができた
  - Zap-In によって計算速度を数百倍高速化し、ラフではない完全な計算を短時間できるようになった。  
この結果、
    - ラフ計算でやっていた時には、余剰生産品が生まれていたが、それをほぼ 0 にした。
    - 原価計算単位を細分化して詳細な原価計算を実現できた。正確な原価計算ができるため、競合他社より精密な見積りが可能で、赤字にならず受注できるようになった。
- バッチ処理時間を削減し、月次処理から日次処理へ進化できた
  - 毎月末にラフ計算でも 4 日間かかっていた処理を、完全な計算で 4 時間に短縮し、毎月 1 回だった集計を日次の集計に進化できた。
  - 前日までの工場の稼働状況を常に把握でき、計画修正・意思決定が月末を待たずに毎日でもできるようになった。
  - 短時間で計算できることで、ユーザからの急な発注にも対応できるようになった。
  - 短時間で計算できることで、変動する各種コストに対応した最新の原価計算が可能になった。
- 情報利用の促進
  - 月次のデータ集計で状況を把握した時にはすでに手遅れという事態が発生していたのが、日々の迅速な対応ができるようになった。
  - より詳細な製品工程別原価把握により、より細かい対応・進化ができるになり、ビジネスが大幅進化できた。
- 開発期間と費用を非常に少なくできた

- 処理対象の実データを使ったインタラクティブな開発により、開発の期間・費用を削減できた。(Zap-In の自動プログラミングの機能による)
- レガシーシステムからオープン系システムへ移行でき、ハードウェア・ソフトウェアともに低コスト化できた。

---

### 3. 損害保険会社 C 社様

#### 顧客管理システム

- システム概要
  - 顧客情報の数が多いだけでなく変更も頻発するので、顧客マスターは機能していなかった。

このとき顧客との取引記録（入金、支払い、その他）を変更前と変更後を正しく記録してゆく簡便な方法は、トランザクションデータに顧客の最新情報を加えて挿入する方法がある。この場合、行数および項目数の多いテーブルが出現する。項目数が多いため、このテーブルの分析は易しくない。

Zap-In を用いると、多数の項目すべてに高速なインデックスが張られ様々な検索が高速に行え、かつ集計も高速なため、気軽にできなかった情報利用が簡単に行えるようになる。
- 情報利用の促進
  - 新しいデータ処理を依頼してから情報入手まで数日待つこともあったが、迅速で的確な情報提供ができるようになった。
  - これにより、顧客対応・意思決定がスピードアップできた。
- 開発期間と費用を非常に少なくできた
  - 処理対象の実データを使ったインタラクティブな開発により、開発期間と開発費用を削減できた。(Zap-In の自動プログラミングの機能による)
  - ユーザーが欲しい情報かどうかをその場で確認できるので、間違いのないデータ処理結果提供がやり直しなくできた。
  - オンラインバッチ処理をホストからオフロードし、新規投資を抑制できた。

---

#### 4. 総合技術商社 D社様

##### 財務管理システム

- システム概要
  - 膨大なトランザクションの中で、不整合の検出が重要なシステム。そのためには検索・集計・マッチング・カテゴリズなどのすべての機能で高速性が必要。
- 専用機の廃止によって情報処理コストを削減できた
  - 専用機からオープンシステムへ移行でき、コストを削減できた。（ハードウェア・ソフトウェア・システム開発・運用・保守など）
  - 定型的システム運用から、フレキシブルで迅速な情報利用が可能なシステムへ移行できた。
  - 経営ニーズに迅速に対応でき、意思決定のスピードアップができるようになった。
- 開発期間と費用を非常に少なくできた
  - 現行システムからノンプログラミングでシステム移行できた。
  - 実データを使ったインタラクティブな開発によって、間違いのないスムーズな移行ができた。

---

#### 5. 通信販売会社 E社様

##### 顧客管理システム

- システム概要
  - 通信販売会社にとっては、ダイレクトメールのヒット率が営業の成否を分ける。その分析には、集計だけではなく、ジョイン・マッチング・検索・カテゴリズなどの多岐にわたる機能を使用したシミュレーションが必要で、これらの機能すべてが高速でなければ十分な分析を行うことができない。
- ダイレクトメールのヒット率を向上、売上を拡大

- 担当者が自分の業務経験を業務スキルを活かして自分で新機能を追加するシステムができた。(Zap-In の自動プログラミングの機能による)
- データベースを情報利用の身近なツールとして業務に活用できるようになった。
- これにより、ダイレクトメールヒット率を3～5倍と大幅に改善できた。
- 開発期間と費用を非常に少なくできた
  - 従来に比べ、大幅に少ない予算と期間でシステム構築を実現できた。

---

## 6. 情報通信機器製造販売会社 F 社様

### 調達管理システム

- システム概要
  - 大手製造業では、多数の工場が個別に調達システムを構築していることが多い。これを本社が集中的に行うようにすれば、下記のように10%以上の調達コストの削減ができる。
    - 部品在庫が削減できる
    - 部品を整理統合して種類を大幅に削減できる
    - 必要な品質を満たしながらコスト的に最適な部品を選択できる
    - 情報利用コストの削減
  - ユーザー自身による新機能追加によって、非定型な情報利用を実現できた。(Zap-In の自動プログラミングの機能による)
  - オープンシステムに移行でき、ハードウェア・ソフトウェア・運用・外部委託のコストを削減できた。
  - 情報の鮮度向上によるペーパーレス促進ができた。(レポート出力80%削減)

- 調達部品変更・価格交渉等に調達明細データを活用し、迅速で的確な意思決定ができるようになった。

## 4. 製品紹介

### 4-1. ZAP-IN シリーズ AKTBLITZⅢ 製品概要

#### 動作環境

OS: Windows 7/8/10, Linux

ディスク: 100GB 以上推奨

---

#### 4-1-1. AKTBLITZⅢ スタンドアローン版

- 強力なグラフィカル・ユーザーインターフェースを備えています。

- タイプ S:

- OS: 32bit/64bit

- メインメモリ: 2GB まで (概ね 1000 万行まで)

- スレッド: 4 まで

- JOB 数: 16 まで

- タイプ L:

- OS: 64bit

- データ量制限: 最大 20 億行まで

- スレッド: 16 まで

- JOB 数: 999 まで

---

#### 4-1-2. AKTBLITZⅢ サーバー・クライアント版

- データベース処理を担当するサーバー機と、グラフィカル・ユーザーインターフェースを担当するクライアント機 (複数) とをネットワークで接続して運用します。

- タイプ S: (サーバー機側)

- OS: 32bit/64bit

- メインメモリ: 2GB まで (概ね 1000 万行まで)

スレッド： 4まで

JOB数： 16まで

- タイプL： (サーバー機側)
  - OS： 64bit
  - データ量制限： 最大20億行まで
  - スレッド： 16まで
  - JOB数： 999まで

---

#### 4-1-3. AKTBLITZⅢ クラスタ・サーバー版

- サーバー・クライアント版のサーバー機を最大16台まで並列動作させます。
- タイプL： (サーバー機16台の場合)
  - OS： 64bit
  - データ量制限： 最大20億行まで
  - スレッド： 16まで
  - JOB数： 999まで

---

#### 4-2-4. AKTBLITZⅢ ウェブサービス版

- サーバー・クライアント構成のサーバーを、ウェブサービス構成にしたものです。クライアントPC・スマートフォンのウェブブラウザを利用します。
- タイプS： (サーバー機側)
  - OS： 32bit/64bit
  - メインメモリ： 2GBまで(概ね1000万行まで)
  - スレッド： 4まで
  - JOB数： 16まで
- タイプL： (サーバー機側)
  - OS： 64bit
  - データ量制限： 最大20億行まで

スレッド : 16 まで

JOB 数 : 999 まで

- クライアント機側
  - Google Chrome, Microsoft IE などを搭載した PC、タブレットなど。

---

#### 4-1-5. AKTBLITZⅢ ENGINE 組み込み用モジュール版

- お客様側で開発するソフトウェアシステムに AktblitzⅢの機能を組み込むためのモジュールです。強力な API を備えています。
- タイプ S :
  - OS: 32bit/64bit
  - メインメモリ : 2GB まで (概ね 1000 万行まで)
  - スレッド : 4 まで
  - JOB 数 : 16 まで
- タイプ L :
  - OS: 64bit
  - データ量制限 : 最大 20 億行まで
  - スレッド : 16 まで
  - JOB 数 : 999 まで

#### 4-2. ZAP-IN シリーズ AKTBLITZⅣ (仮称)

- AktblitzⅢの上限、20 億行より大規模なデータを超高速に処理できる製品を計画中です。



## 5. 会社概要

- 事業方針
  - ミッション
    - データベースの画期的イノベーション技術を世界に普及させ、「ビッグデータを広範に利用して発展してゆく世界」を作り出す。
  - ビジョン
    - ビッグデータの画期的新技術を創出し続ける。
    - その画期的新技術を社会で役立つ製品として広く世界へ普及させる。
- 名称
  - 株式会社ターボデータラボラトリー  
Turbo Data Laboratories, Inc.
- ウェブページ
  - <http://turbo-data.co.jp>
- 本社
  - 231-0004 神奈川県横浜市中区元浜町 3-21-2 ヘリオス関内ビル  
電話：045-222-8826
- 流山拠点
  - 270-0152 千葉県流山市前平井 61  
電話：047-151-0398 FAX：047-115-8512
- 役員
  - 代表取締役社長 古庄 晋二  
開発担当取締役 ジャヒル フセイン  
社外取締役 小笠原 國義  
社外取締役 河上 達

- 資本金
  - 2億44万円
- 設立
  - 2000年8月28日

## 5-2. 販売パートナー様

- ITバリューアソシエイツ株式会社 <http://www.itvalue.jp/>
- 株式会社ウェブアイ <https://www.webi.co.jp/>
- 株式会社エスペラントシステム <https://www.ess-g.com/>
- 株式会社クロスキャット <http://www.xcat.co.jp/>
- 株式会社シイズ <http://www.kkseeds.com/>
- 株式会社セック <http://www.sec.co.jp/>
- 株式会社ニイズ <http://www.needs-inc.net/>
- 株式会社ニッセイコム <http://www.nisseicom.co.jp/>
- 株式会社日本コンピュータ開発 <http://www.nck-tky.co.jp/>
- 日本ソフト開発株式会社 <http://www.nihonsoft.co.jp>
- 株式会社ハイマックス <http://www.himacs.jp/>
- ベイソフト株式会社 <http://baysoft21.co.jp/>

