

Object Oriented Development & Object Oriented DataStore

オブジェクト指向システムと オブジェクト指向DBMS



株式会社 日本総合研究所
The Japan Research Institute, Limited

事業化技術センター 副所長
細川 努
hosokawa.t@jri.co.jp

オブジェクト指向開発の今

オブジェクト指向方法論

- オブジェクト指向の概念発生してから30年
- 開発方法論として急成長したのは7年前から

開発環境の充実

- Java(J2EE、モバイル)
- .NET

何を実現するのか

- ビジネスとITの一体ソリューション
- システム統合、ビジネス統合

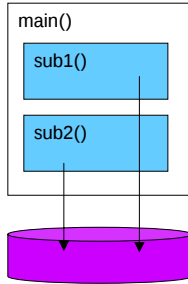
オブジェクト指向とは

● オブジェクト指向とは

「オブジェクト」という単位でシステム構築を考えるアプローチ

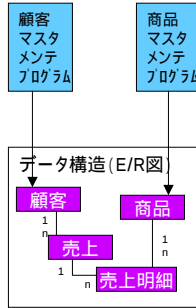
「構造化手法」

アルゴリズムを構造化・分割設計することによってシステムを構築するアプローチ



「データ中心アプローチ」

システムが処理するデータ構造を設計してから、処理機能を構築するアプローチ



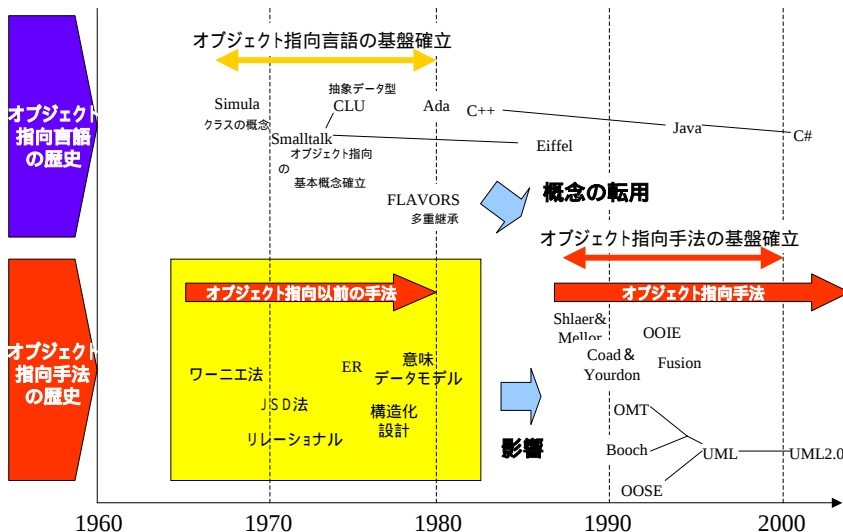
「オブジェクト指向」

オブジェクトを単位としてシステムを構築
→ オブジェクト同士が協調して動作するシステムとなる



Object Oriented Development
of Object Oriented DataBase

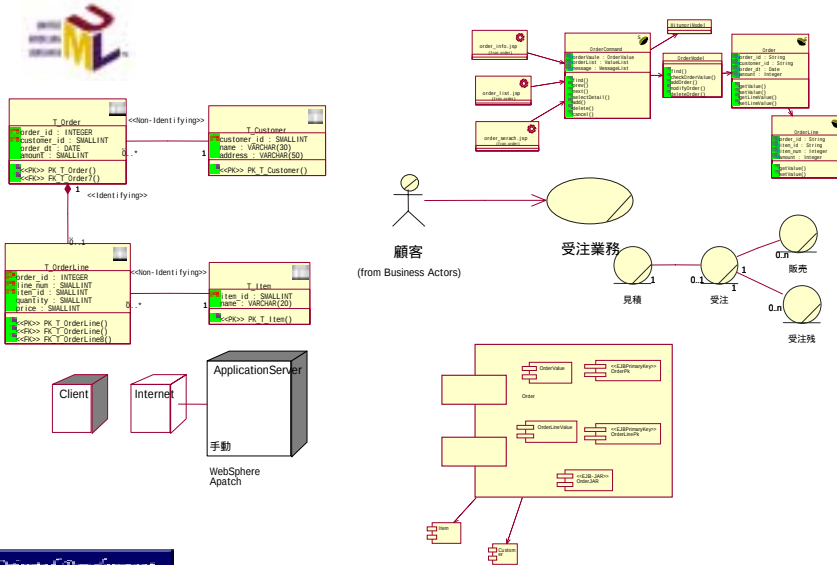
オブジェクト指向の歴史



Object Oriented Development
of Object Oriented DataBase

UML(Unified Modeling Language)

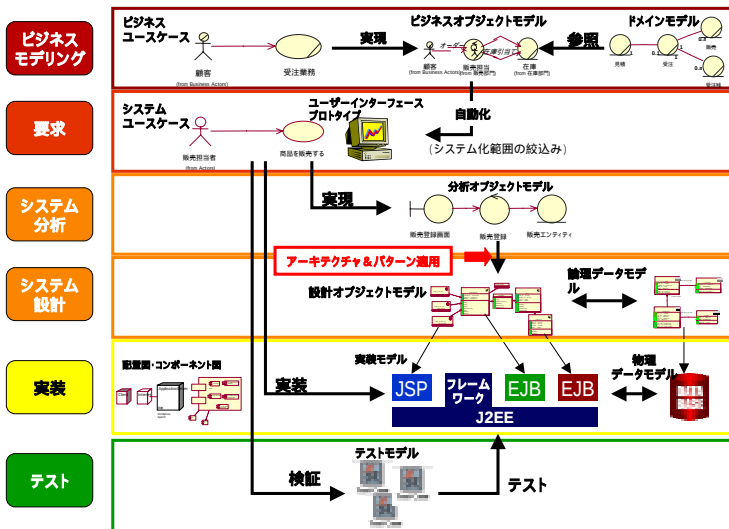
● オブジェクト指向における統一モデリング表記法



Object Oriented Development
of Object Oriented DataStore

オブジェクト指向開発の全体像

● 業務分析から実装・テストまでを一貫して分析・設計



Object Oriented Development
of Object Oriented DataStore

オブジェクト指向開発環境

● 統合開発環境の進化

- 従来、オブジェクト指向CASE、プログラミング、ビルド管理、自動テスト、構成管理等の各種ツールは別々の環境
- 最近、オープンソースのIDE(統合開発環境)に各種ツールをプラグイン化する傾向に

プラグイン

Rational
XDE

Together
CC

Ant ,etc

jUnit ,etc

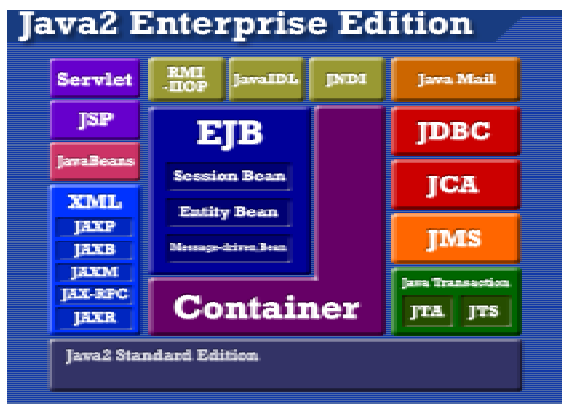
Eclipse WorkBench



Object Oriented Development
of Object Oriented DataStore

オブジェクト指向開発プラットフォーム

- J2EE (Java2 Platform Enterprise Edition)
ミッションクリティカルなシステムを構築するための
サーバーサイドJava環境



- **Servlet・JSP**
-JavaによるWebアプリを構築する機能
- **EJB**
-業務機能(ビジネスロジック・データ)を部品化するコンポーネント技術
- **RMI-IIOP, JavaIDL, JNDI**
-分散連携に関する技術
- **JDBC**
-リレーショナルDBへの接続
- **JCA**
-EAIアダプタへの接続
- **JMS**
-非同期メッセージング
- **JTA, JTS**
-トランザクションサービス
- **JAXP, JAXB, JAXM, JAX-RPC, JAXR**
-XMLおよびWebサービス関連機能

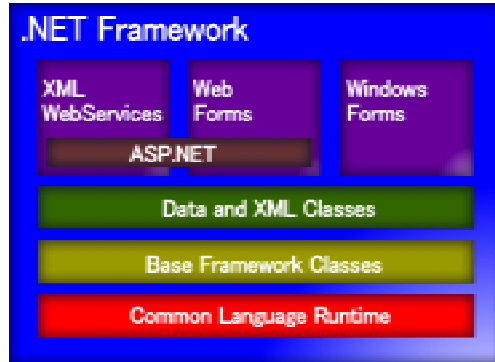
Object Oriented Development
of Object Oriented DataStore

オブジェクト指向開発プラットフォーム

● Microsoft .NET

新しいオブジェクト指向プログラミング環境()や
XML技術、WebServiceを積極的に活用

複数の言語(C#, J#, etc)を中間言語に変換し
共通言語実行環境(CLR)で実行



Object Oriented Development
of Object Oriented DataStore

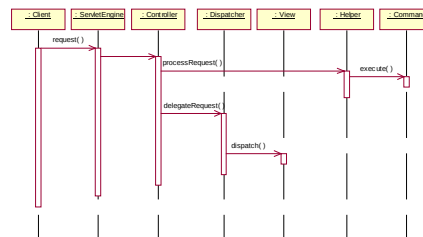
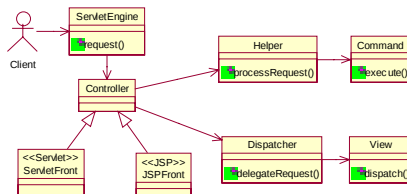
オブジェクト指向を目指すもの

● デザインパターン

オブジェクト指向における良い設計のお手本

- GOFのデザインパターン
- J2EEパターン

J2EEパターン(Service To Walker)

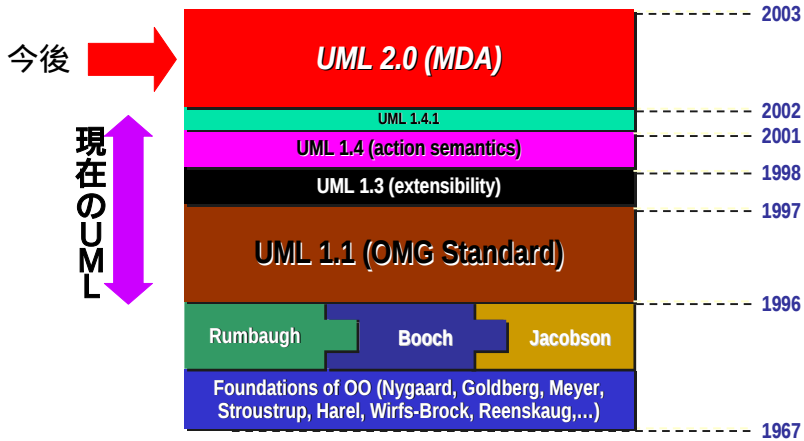


Object Oriented Development
of Object Oriented DataStore

オブジェクト指向を目指すもの

● UMLの未来

- より複雑なモデル記述への対応
- MDA(Model Driven Architecture)



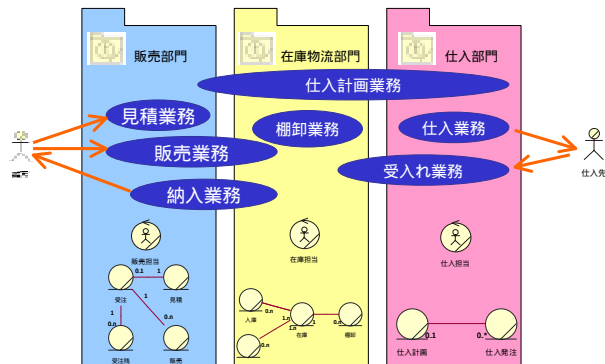
Object Oriented Development
of Object Oriented DataStore

オブジェクト指向を目指すもの

● ビジネスモデリング

オブジェクト指向は、ビジネスプロセスやビジネス構造を分析する際にも大きな威力を発揮

企業ビジネスプロセス・構造をオブジェクト指向でモデル化



Object Oriented Development
of Object Oriented DataStore

現代のデータマネジメント

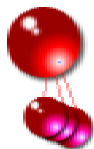
● 情報の多様化

現代においては、ビジネスや社会生活で取り扱うデータが
複雑化 + 高度化 + 大量化しつつある

ゲノムデータ



オブジェクト？



Business Intelligence



部品データ



XMLデータ

XML

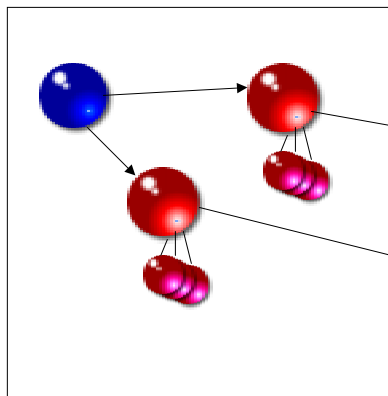
Object Oriented Development
of Object Oriented DataStore

オブジェクト指向におけるデータマネジメント

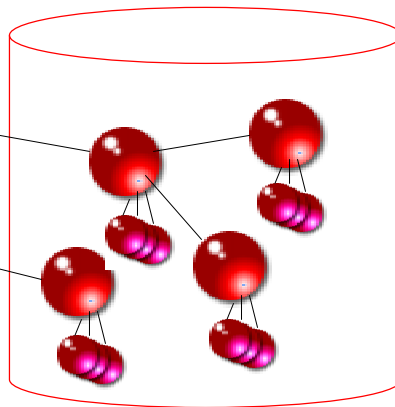
● オブジェクトデータの格納

オブジェクトモデルをそのまま、関係、依存性を維持したまま
データ格納するのが理想形(オブジェクト指向データベース)

オブジェクト実行環境



オブジェクトデータストア

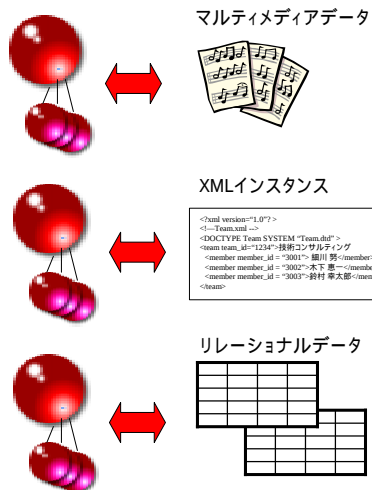


Object Oriented Development
of Object Oriented DataStore

なぜオブジェクトデータか？

● 情報管理の柔軟性

- ・オブジェクトデータは他のデータモデルにも柔軟に対応
- ・データ抽象化によってオブジェクトの振る舞いを隠蔽



Object Oriented Development
of Object Oriented DataStore

オブジェクトデータベース

● メリット

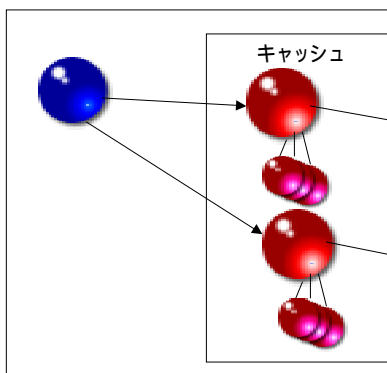
- オブジェクトデータ管理として理想的
- 検索は効率的

● デメリット

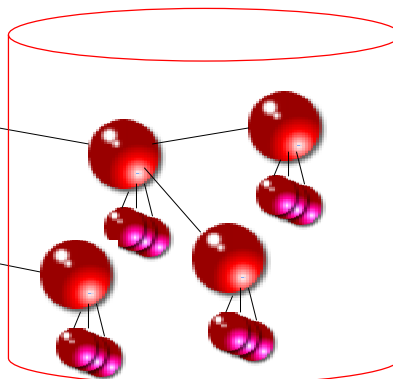
- 大量一括処理に弱い
- 更新に関するミッションクリティカル性が低い

➡ (残念ながら)リレーショナルデータベースほど普及せず

オブジェクト実行環境



オブジェクトデータベース

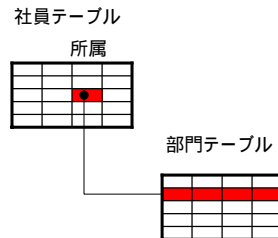


Object Oriented Development
of Object Oriented DataStore

リレーショナルデータベースのオブジェクト拡張

● オブジェクトリレーショナル

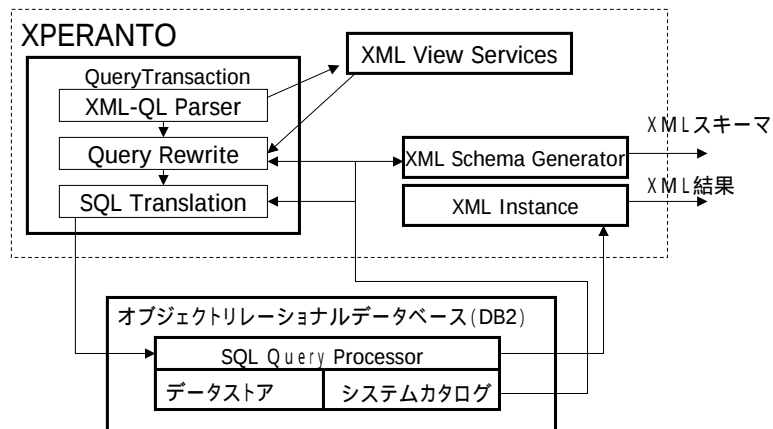
- リレーショナルデータベースをオブジェクト拡張
- 他テーブルの行インスタンスに対する参照
- テーブルTYPEの継承/拡張
- 複雑なデータは扱えるようになったがオブジェクトの振る舞いをデータベースに実装することには限界がある。



リレーショナルデータベースのオブジェクト拡張

● オブジェクトリレーショナルに対して各種エクステンションで拡張

XPERANTO: オブジェクトリレーショナルに対してXML格納・アクセスを実現

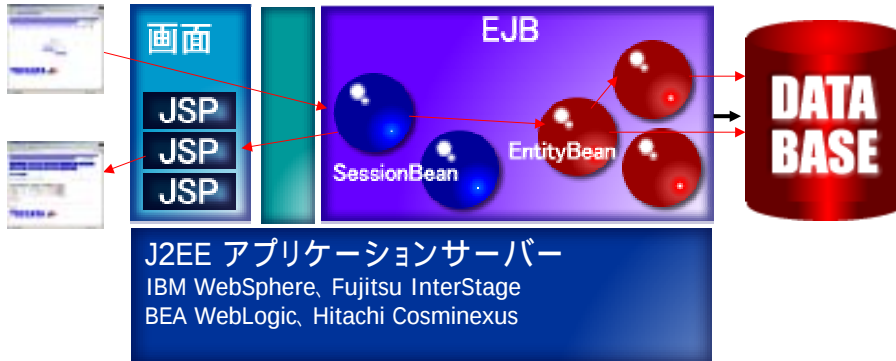


EJBコンポーネント

● EJB(企業システム向けのJavaコンポーネントモデル)

リレーショナルデータベースに対して、オブジェクトをマッピング

→複雑な振舞いはコンポーネント側に実装

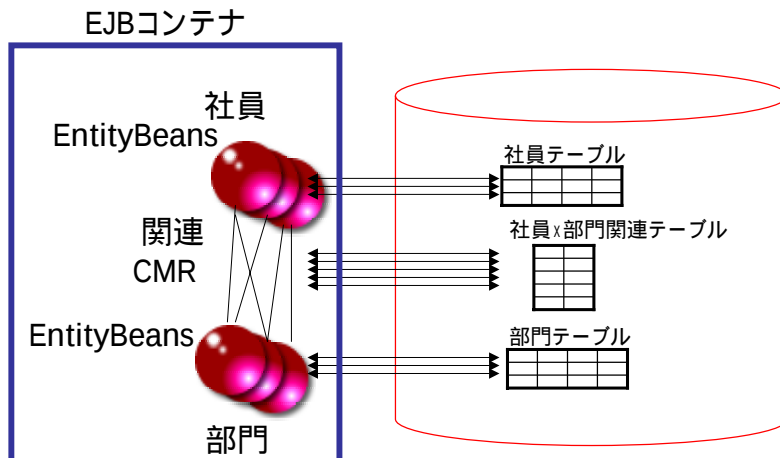


Object Oriented Development
of Object Oriented DataStore

EJB 2.0でのオブジェクト-リレーションマッピング

● EJB (J2EEの分散コンポーネントモデル)

例) 社員が複数部門への兼任がある組織データの実装

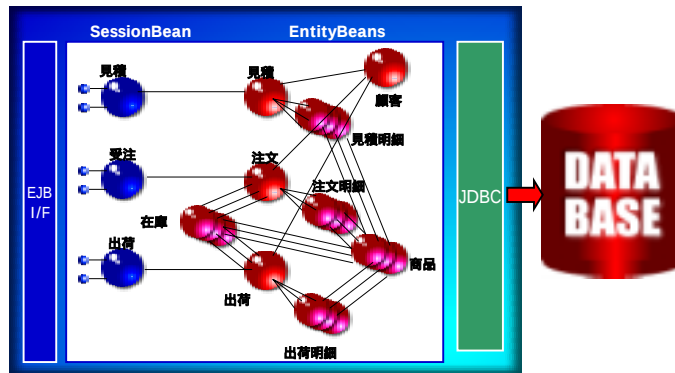


Object Oriented Development
of Object Oriented DataStore

EJBによる企業システム構築

● EJBによる業務機能オブジェクトの部品化

- 一般的な企業システムも、オブジェクトの構造としては複雑
データベース側を複雑にするのではなく、EJBコンポーネント
側でオブジェクト指向システムとして構築

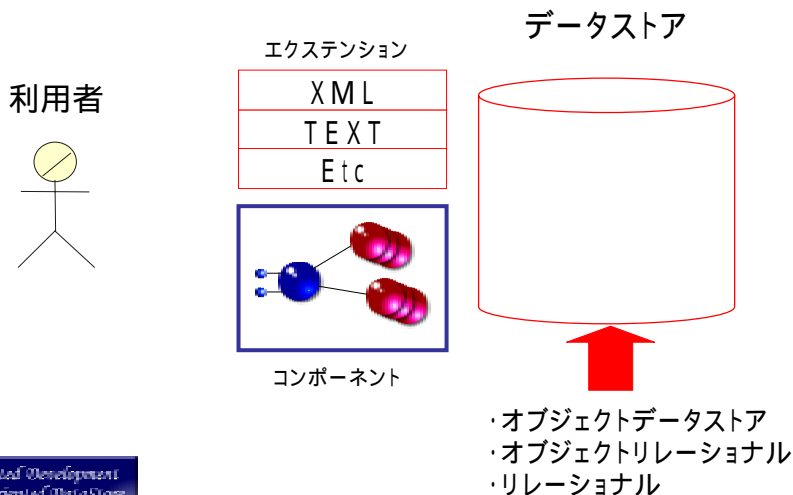


Object Oriented Development
of Object Oriented DataStore

オブジェクト指向データ管理のまとめ

● オブジェクト指向データ管理の多様化

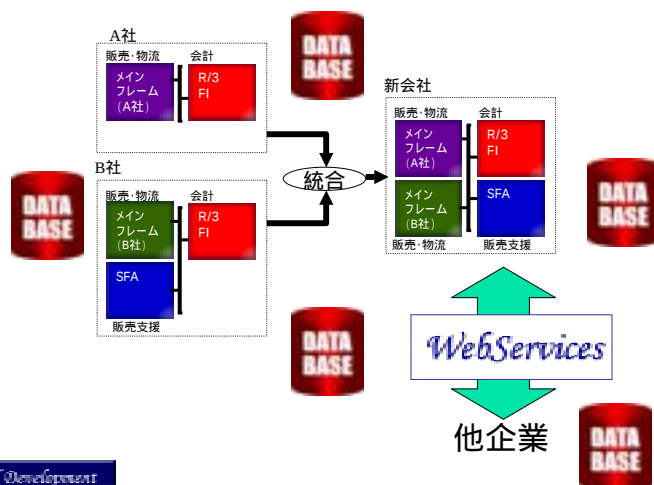
データストアだけのオブジェクト化でなく、
周辺環境(エクステンション、コンポーネント)
のオブジェクト指向化も進む



Object Oriented Development
of Object Oriented DataStore

さらにグローバルなオブジェクトデータ管理へ

- 今後、もっと大きな単位でのデータ管理が必要
 - システム間統合
 - 企業統合



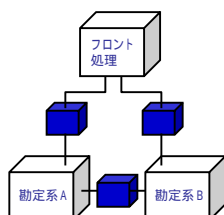
Object Oriented Development
of Object Oriented DataStore

統合技術の進化

● システム間の機能的連携

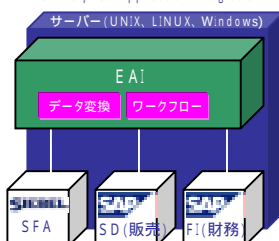
従来のシステム統合は、システム間を単に連携させることを目的としていたが
これからは、業務機能間での機能的連携が必要に。

第一世代 独自技術統合

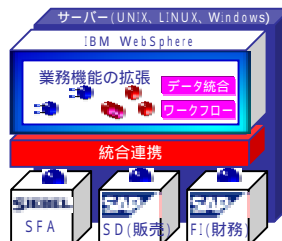


第二世代 ミドルウェア中心統合

EAI: Enterprise Application Integration



第三世代 コンポーネント型統合

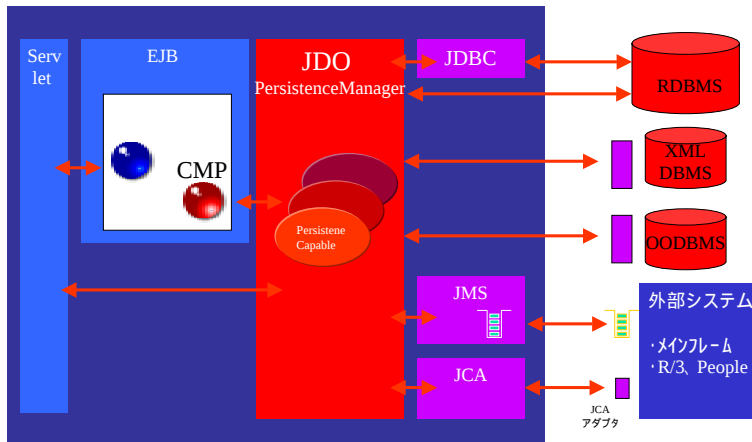


Object Oriented Development
of Object Oriented DataStore

Javaにおける統合技術

● JDO (JavaDataObject)

データベースだけでなく、外部システム自体をオブジェクトとしてアクセス可能

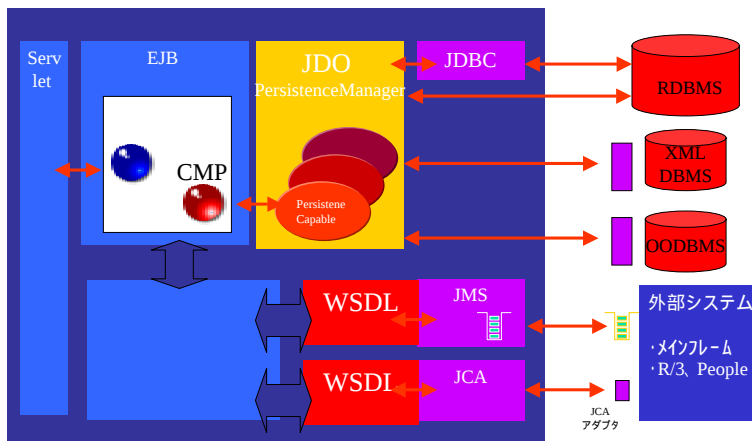


Object Oriented Development
of Object Oriented DataStore

Javaにおける統合技術

● Webサービス (WSDL) による統合

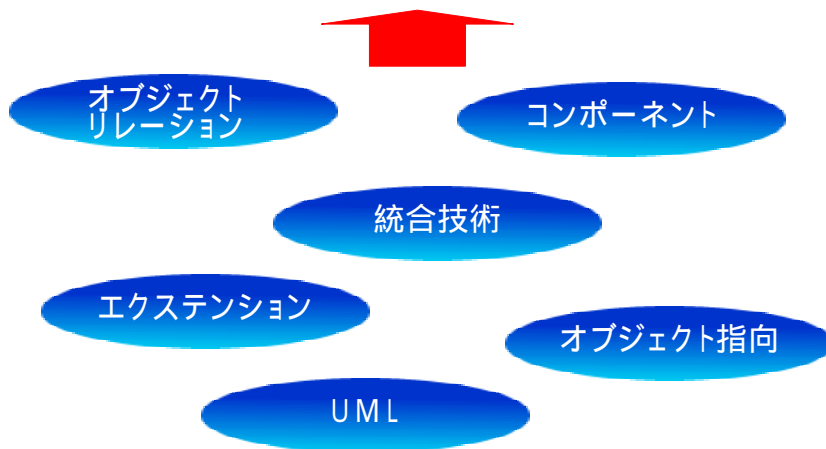
柔軟性の高い統合を実現



Object Oriented Development
of Object Oriented DataStore

これからのシステム開発

もっと拡張性があり、インテリジェンスなシステム構築



Object Oriented Development
of Object Oriented DataStore