

XMLDB入門

～XMLDBの特性とシステム作りのポイント、適用分野～

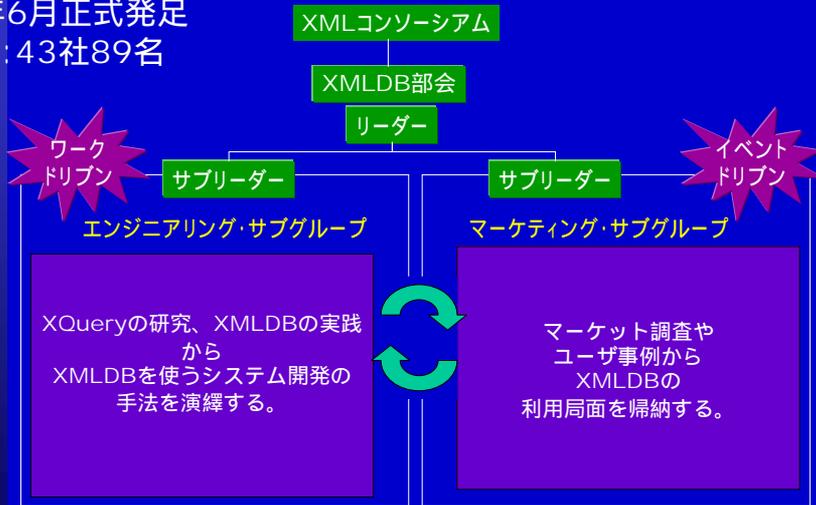
XMLコンソーシアム
XMLデータベース部会
リーダー 加藤哲義 (ジャストシステム)
tetsuyoshi.kato@justsystems.com



1. XMLデータベース部会の紹介

XMLDB部会の構成と活動指針

2006年8月から活動開始
2007年6月正式発足
メンバー：43社89名



XMLデータベース (XMLDB) 部会の背景

2006～7年にかけてXMLデータ処理の共通基盤が整う。

XML Query1.0 正式勧告 (2007年1月)

各社からXML DB製品リリース
(ネイティブXMLDB & RDBハイブリッド型)

XMLをB2Bの中間フォーマットというより、
蓄積して、活用するためのデータとしてとらえる潮流

XMLDBを、何に、どうやって使うの？

XML DB部会の目的

XMLDBを 何に どうやって 使うの？

マーケットスタディ

技術的実践

マーケティング・サブグループ

エンジニアリグ・サブグループ

XMLDBの市場形成とXMLDB技術者の拡大

5

2. XMLDBの特性

6

XMLはデータとドキュメントの掛け橋

XMLはデータとドキュメント、
両方の特性を持っている。

データ処理できるドキュメントとも
ドキュメントの自由度をもった
データともいえる。



XMLは構造と値が一体化されたテキストデータ

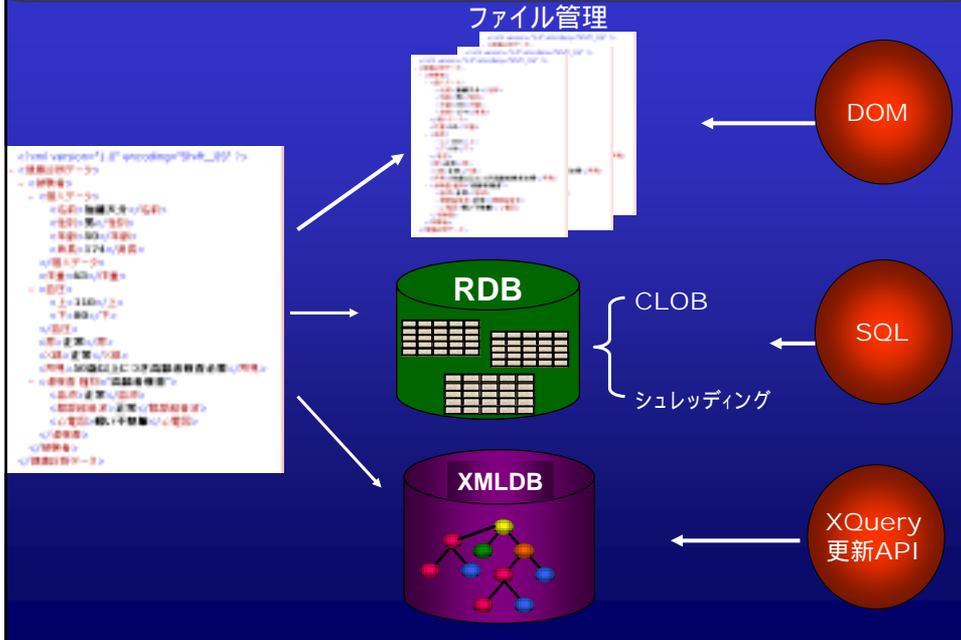
```
<?xml version="1.0" encoding="Shift_JIS" ?>
<健康診断データ>
  - <健康者>
    - <個人データ>
      <名前>加藤大介</名前>
      <性別>男</性別>
      <年齢>50</年齢>
      <身長>174</身長>
      <個人データ>
        <体重>63</体重>
    - <血圧>
      <上>110</上>
      <下>80</下>
      <血糖>
        <非空腹前>
          <値>正常</値>
        <空腹前>
          <値>59歳以上かつ高血糖を報告する</報告>
    - <健康診断項目>高血糖検査</検査>
      <血糖>正常</血糖>
      <脂質検査>正常</脂質検査>
      <心電図>軽い不整脈</心電図>
    </健康者>
  </健康診断データ>
```

- ・記述性
階層化による複雑なデータ表現が可能
- ・可塑性
いつでもデータ構造の改造、加工が容易
- ・再利用性
必要な箇所を利用、様々なビューで利用
- ・永続性
データの原本はDBやアプリケーションでなくこのテキストファイル

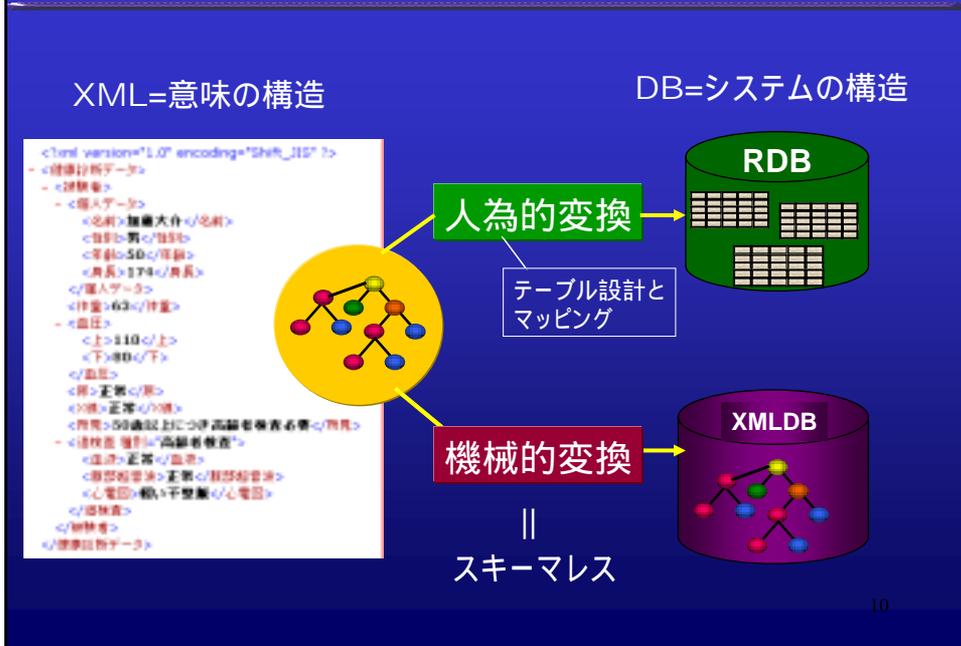
対象をリアルにデータ表現できる
まずはDB設計を意識しなくてもよい

現場感のあるデータ
システムのプロでなくてもよい

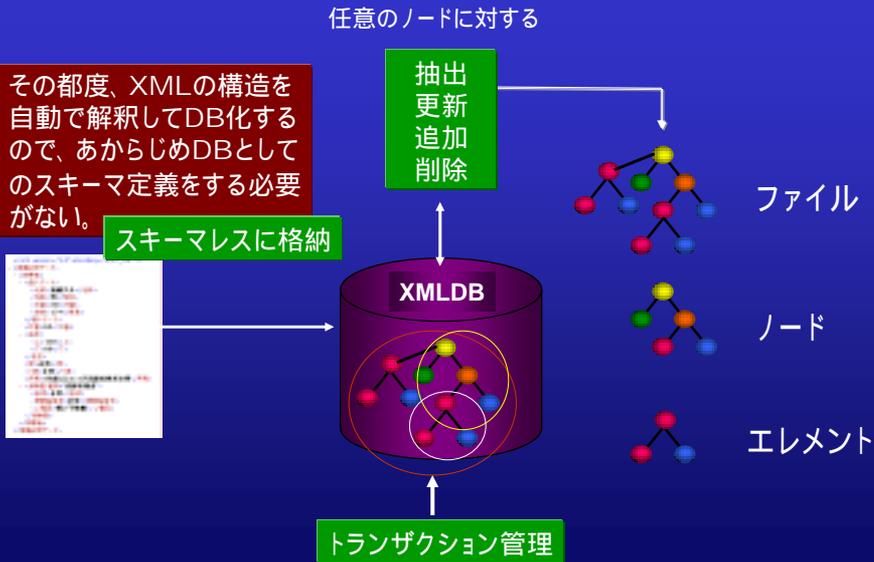
XMLデータの格納先とAPI



意味の構造からシステムの構造への変換



XMLDBの特性



任意の粒度での処理

```
<?xml version="1.0" encoding="Shift_JIS" ?>
<健康診断データ>
  <被験者>
    <個人データ>
      <名前>加藤大介</名前>
      <性別>男</性別>
      <年齢>50</年齢>
      <身長>174</身長>
    </個人データ>
    <体重>63</体重>
  </被験者>
  <血压>
    <上>110</上>
    <下>60</下>
  </血压>
  <尿>正常</尿>
  <血糖>正常</血糖>
  <所見>50歳以上につき高齢者検査必要</所見>
  <遠検査 種別>高齢者検査
    <血液>正常</血液>
    <胸部超音波>正常</胸部超音波>
    <心電図>軽い不整脈</心電図>
  </遠検査 種別>
</被験者>
</健康診断データ>
```

ファイル

ノード

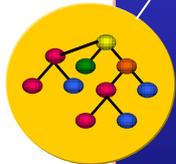
エレメント

/健康診断データ/被験者[個人データ/年齢 >= 40 and 血压/上 >= 100]

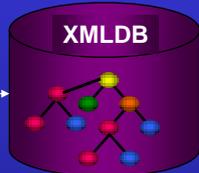
機械的変換はユーザが楽な分、製品が汗をかく

```

<?xml version="1.0" encoding="Shift_JIS" ?>
<健康診断データ>
  <健康者>
    <個人データ>
      <名前>加藤大介</名前>
      <性別>男</性別>
      <年齢>50</年齢>
      <身長>174</身長>
    </個人データ>
    <体重>63</体重>
  </健康者>
  <上>110</上>
  <下>80</下>
</血圧>
<骨>正常</骨>
<胆>正常</胆>
<腎臓>50歳以上かつ血圧を収める者</腎臓>
<尿検査 糖質>高血糖者検査</尿検査>
  <尿糖>正常</尿糖>
  <尿蛋白尿量>正常</尿蛋白尿量>
  <心電図>軽い不整脈</心電図>
</尿検査>
</健康診断データ>
  
```



機械的変換



XMLDB各社は、この方法にしのぎを削る。



XMLDBのカテゴリ

RDB ハイブリッド型



ネイティブ型

DBMS型

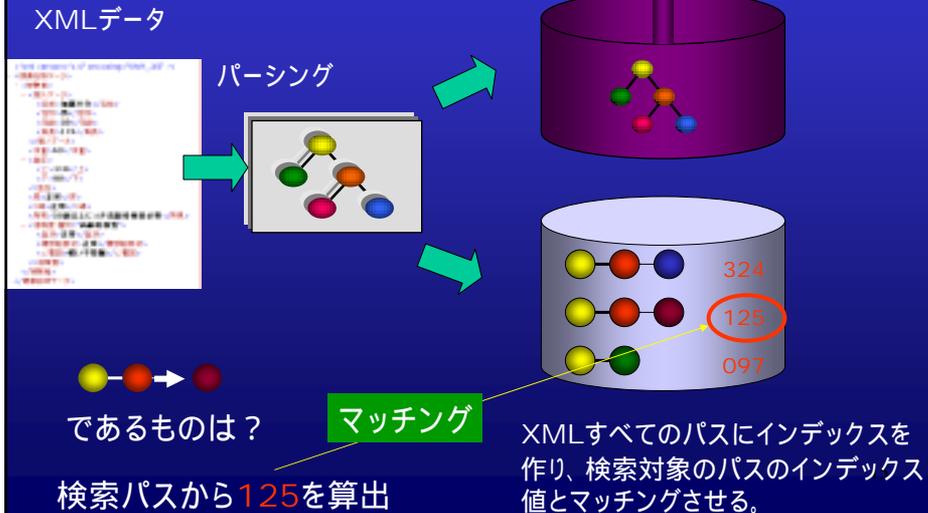


ファイル・インデキシング型



どうやってスキーマレスを実現するか

XMLをオブジェクト化して既存のオブジェクト型DBにそのまま容れる。

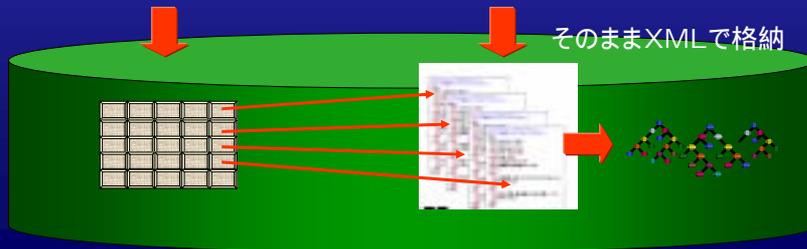


カテゴリごとの特性 (トレードオフ)

	検索性能	格納・更新	性能アップのチューニング
DBMS型	工夫しないと速くない (メモリ量、インデックス、データ配置などの最適化)	負荷軽い	必要 余地有り プロ志向
データのインサート、更新が多いケースで有利			
ファイル インデキシング型	何もしなくても高速 (完全一致でないと速くない)	負荷重い	不要 余地少ない 組込要件
大容量のXML文書を検索するケースで有利			

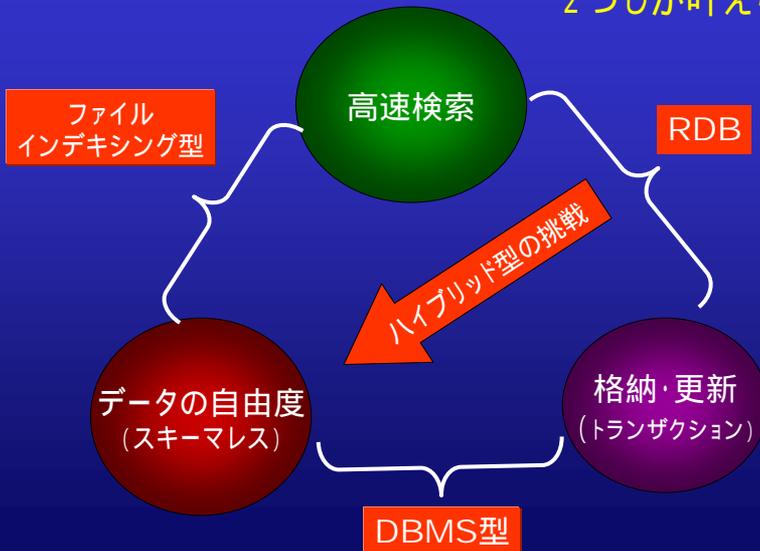
ハイブリッド型 テーブルとXMLネイティブストアの混成

設備製品	共通項目			個別項目			
	項目A	項目B	項目C	項目D	項目E	項目F	項目?
あ							
い							
う							
え							
お							
か							
き							
く							
け							



大容量データに対するDB機能の三すくみ

2つしか叶えられない

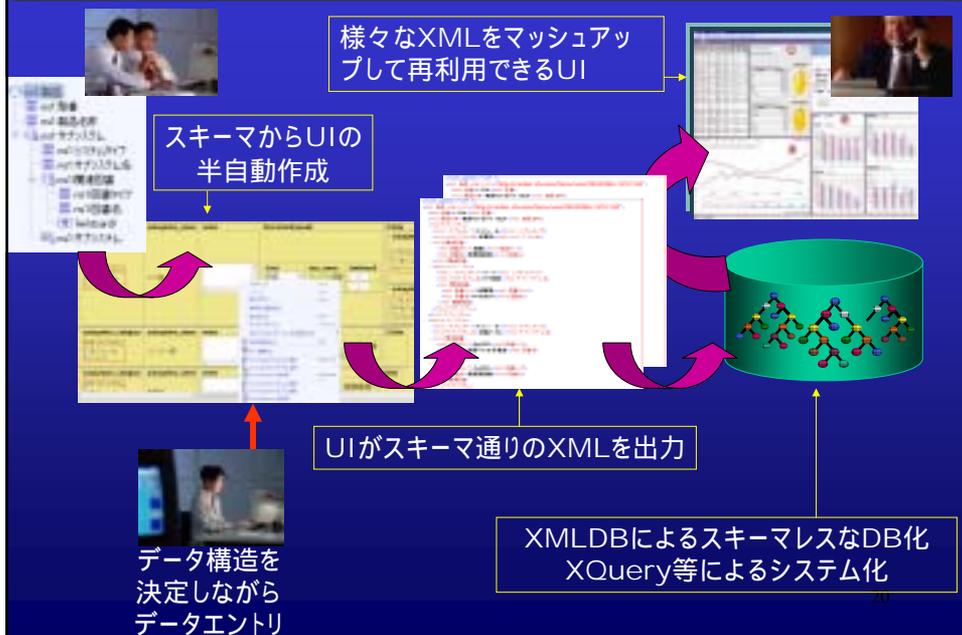


XMLDBを使うポイント

1. XMLデータを扱うからといって、必ずしもXMLDBが必要ともいえない。
(処理の粒度によっては、ファイル処理やRDBでもいいのでは)
2. 各製品の逆鱗を見極めること
3. データ処理要件がその逆鱗に触れない製品を選ぶこと
4. RDBの常識を一度忘れて、白紙で望むこと
5. XML周辺製品を活用すること (UI製品、データ連携製品、等)
6. 不確定モデルはXMLDBで臨め

19

XML周辺製品と一体でシステム化 例 xfy(UI製品)



20

3. XMLDBの適用分野

21

XMLDBの適用分野

1. データタイプとして

- ・ 任意の繰り返し項目があるデータ
- ・ レコード毎に項目のばらつきが激しいデータ
- ・ 階層が深いデータ

22

XMLDBの適用分野

2. システム要件として

- ・ 運用後、データ仕様が変わりやすいシステム
- ・ 文書データだが任意の粒度で処理したいシステム
- ・ ユーザがデータ構造を決定するシステム

23

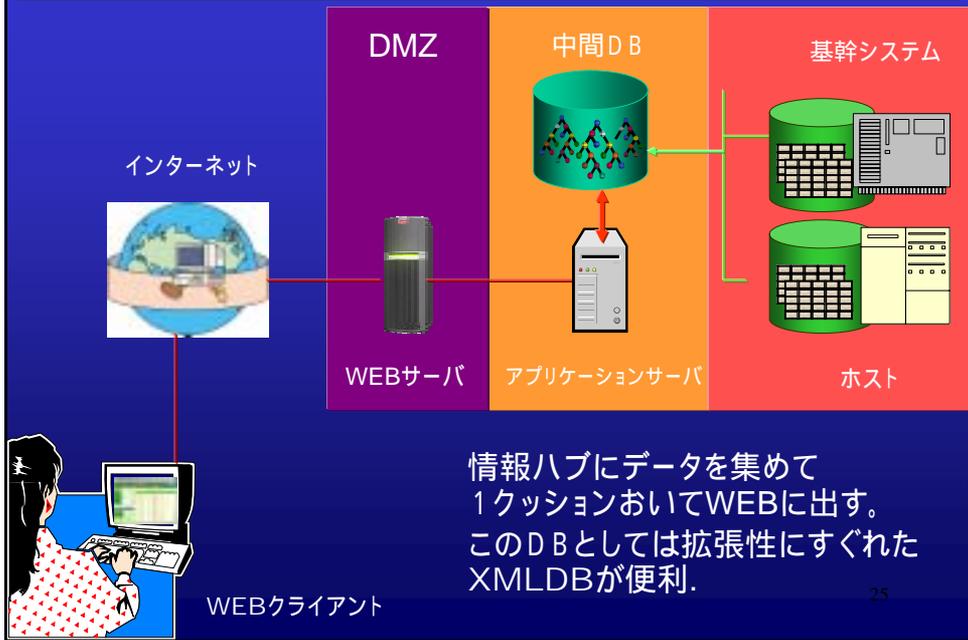
XMLDBの適用分野

3. DB要件として

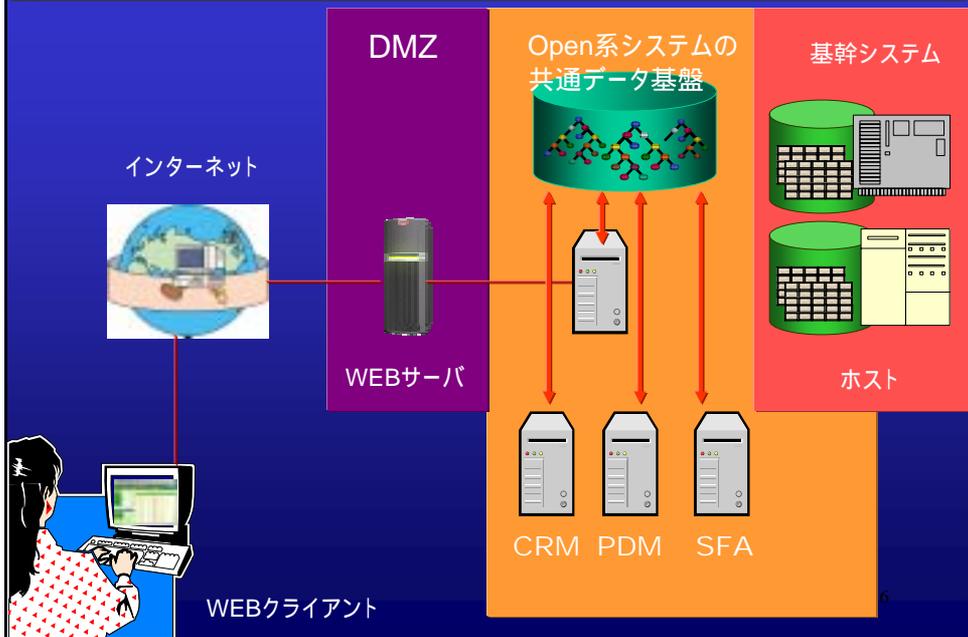
- ・ 基幹系データをWEBシステムに晒す際の間接DB
- ・ 社内データを統合し、再利用するための情報ハブ
- ・ データ仕様が確定するまでの初期開発用DB
- ・ いつか構造的に再利用するだろうファイルシステム
(用途は後から出る)

24

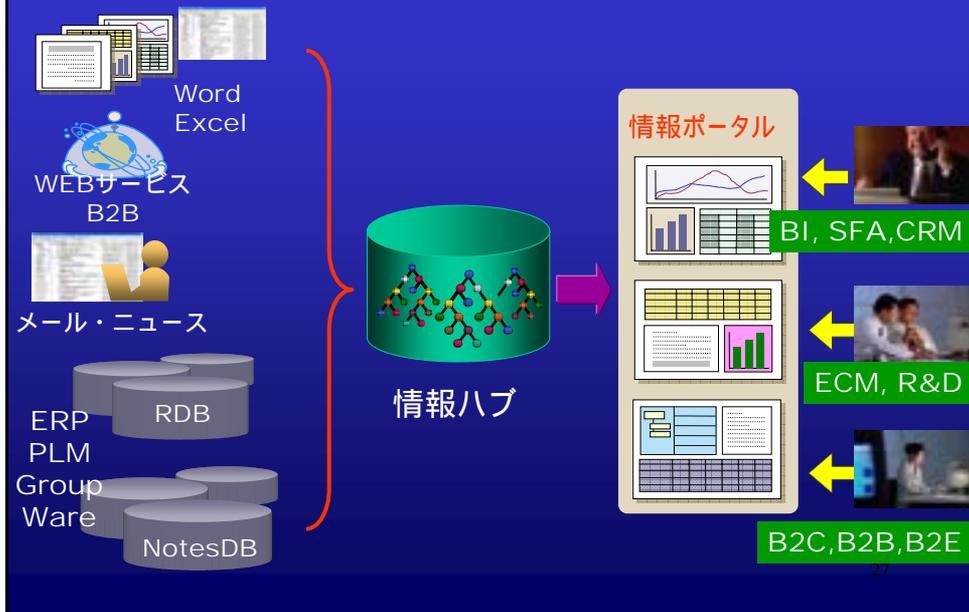
基幹系データをWEBシステムに晒す際の中間DB



社内データを統合し、再利用するための情報ハブ



社内データを統合し、再利用するための情報ハブ



XMLDB勉強会 日本IBM社 訪問

2006.12.2.



次からの、7スライドは、当日の日本IBM様からの資料を引用しています。
その著作権は同社が保有しています。

DB2 9 pureXMLの活用事例

日本アイ・ピー・エム株式会社
ソフトウェア事業
Info.Mgmt.テクニカル・セールス

Information Management software

活用事例1.電子カルテシステム/健診システムのデータ

Why XML

- データの属性がレコードによってまばら
- データの構造変更が多い
- 将来的に他システムと情報連携したい

医療システムのデータも以下の特性を持ち、XML向きのデータと言える

1. 患者、検査ごとに文書の構造(属性)が異なる
2. 患者、検査、年度ごとに構造の変更の可能性がある
3. 将来的に他病院との連携も考えられる

医療データ/純リレーショナル形式の表現

- 属性がまばらなケース(レ=使われている列)
 - 住所を登録している患者としていない患者がいる
 - 患者によってアレルギーのありなしが異なる。一人の患者が複数アレルギーを持つ場合もある
- 構造変更の多いデータ
 - 血液検査で調査する項目は、検査技術の進歩によって変わる可能性がある

患者ID	氏名	性別	生年月日	住所	電話番号	血液型	感染症	アレルギー1	アレルギー2	検査結果1	検査結果2
A	レ	レ	レ	レ	レ	レ				レ	
B	レ	レ	レ	レ			レ	レ	レ	レ	レ
C	レ	レ	レ		レ	レ		レ			
D	レ	レ	レ	レ		レ	レ			レ	レ
E	レ	レ	レ	レ			レ	レ			
F	レ	レ	レ		レ			レ		レ	
...											

医療データ/XMLによる表現

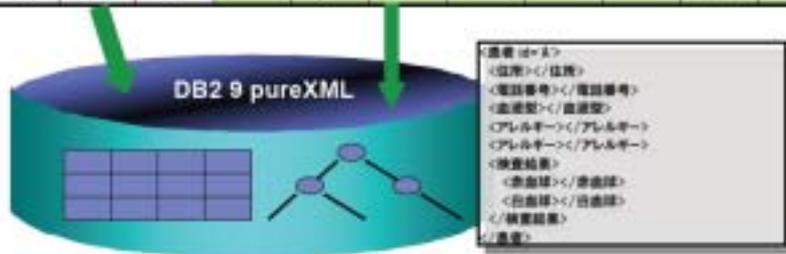
- 属性がまばらなデータをXMLで表現
 - 値の入っている属性のみXMLのタグとして表現
 - 1表でも実現可能
 - システム化が困難だった分野へも適用が現実化
- 柔軟なデータ変更にも対応
 - 変更が他のデータに影響を与えない



医療データ/RDBとXMLのハイブリッド・デザイン

- キーや必須属性はリレーショナル列へ、まばらで変更の多い属性はXML列へ

患者ID	氏名	性別	生年月日	住所	電話番号	血液型	診療科	アレルギー1	アレルギー2	検査結果1	検査結果2
A	レ	レ	レ	レ	レ	レ		レ	レ	レ	レ
B	レ	レ	レ	レ			レ	レ	レ	レ	レ
C	レ	レ	レ		レ	レ					
D	レ	レ	レ	レ	レ	レ	レ			レ	レ
E	レ	レ	レ	レ	レ		レ				
F	レ	レ	レ		レ					レ	



例1) ハイブリッド形式のデザイン

DB2 Viper
Star Festival

• 2表で自然な従属を反映

- 誰が何々を購入
- 顧客情報や言語の拡張性大

注文 - 戦略的施策を早期に実現可能

注文番号	顧客番号	注文
987564331	A6789	
...	...	

商品

商品番号	価格	課税	説明
A54	200	Y	
985	115	Y	
...

```

<注文 ponum="987564331">
  <顧客 custid="A6789">
    <name>.....</name>
    <address>.....</address>
  </顧客>
  <itemlist>
    <item>
      <商品番号>A54</商品番号>
      <数量>12</数量>
    </item>
    <item>
      <商品番号>985</商品番号>
      <数量>1</数量>
    </item>
  </itemlist>
</注文>

<商品 pathno="A54">
  <言語 id="1">
    <説明>鉛筆</説明>
  </言語>
  <言語 id="2">
    <説明>Pencils</説明>
  </言語>
</商品>
    
```

例1) 純リレーショナル形式のデザイン

DB2 Viper
Star Festival

・注文伝票: 4表のジョイン

注文

注文番号	顧客番号
987564331	A6789
...	...

商品

商品番号	価格	課税
A54	200	Y
985	115	Y
...

注文明細

注文番号	商品番号	数量
987564331	A54	12
987564331	985	1
...

商品カタログ

商品番号	言語	説明
A54	1	鉛筆
A54	2	Pencils
...

IBM

XMLDB勉強会

日本オラクル社 訪問

2007.2.5.

次からの2スライドは当日の日本オラクル様からの資料を引用しています。
その著作権は同社が保有しています。

某医療薬品研究所 (1)

- 会社概要
 - 遺伝医学研究、人ゲノム解析、医薬品開発
- 遺伝調査サーバ
 - 大規模研究データを格納
 - 紙媒体から電子データへ変換
 - データ収集とWeb サポート
 - XML DB を採用による、システムのパフォーマンス向上、コストの削減化を実現
- リレーショナルシステムからの移行
 - 利用テーブル数を 1/8 に削減
 - アプリケーションコード 1/4 に削減

ORACLE

Copyright © 2007, Oracle. All rights reserved.

31

某医療薬品研究所 (2)

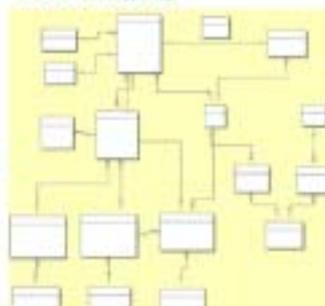
XML DB 導入によるデータモデルのシンプル化

RDBMS構造



導入前

XML DB構造

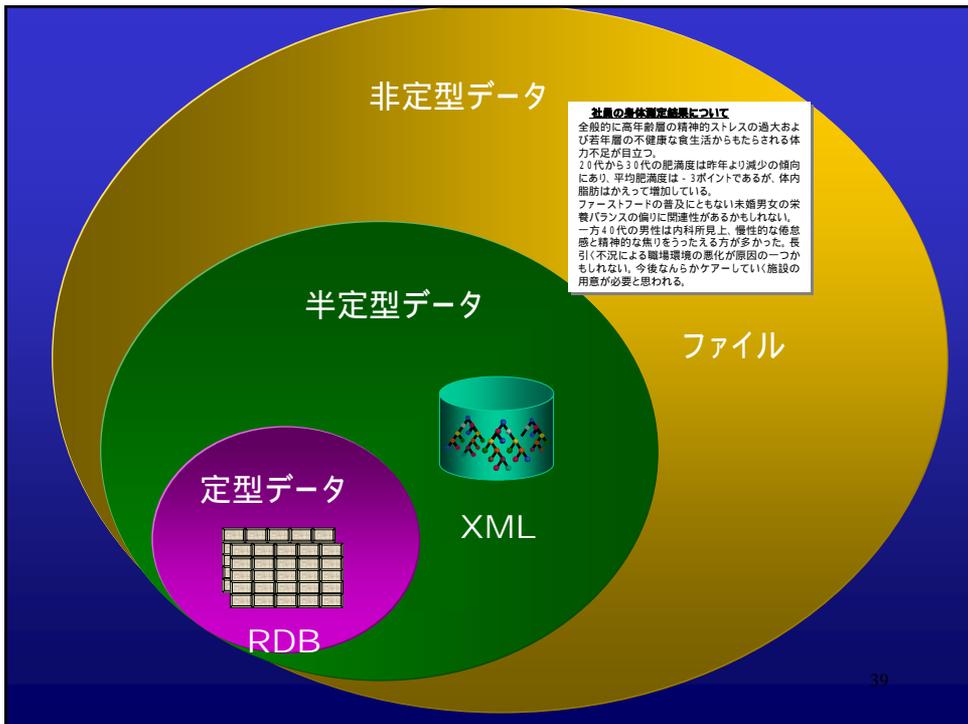


導入後

ORACLE

Copyright © 2007, Oracle. All rights reserved.

32



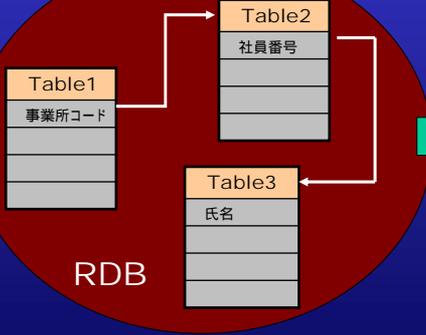
半定型データ って何？

どこまでが定型でどこからが不定型なのか

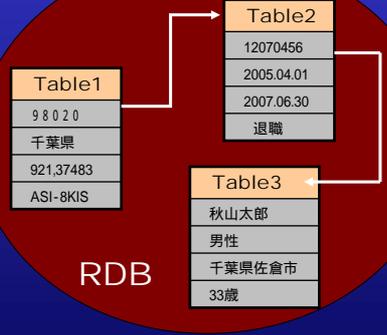
40

RDBは一貫して定型

SEによるDB設計

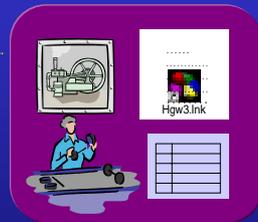


ユーザによるデータエントリ



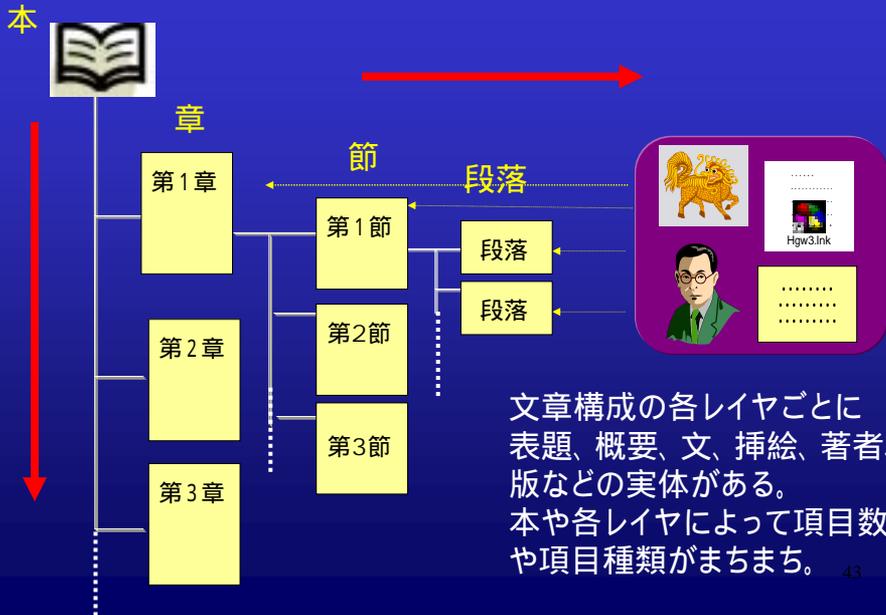
DB定義(設計) = データ構造(形)の決定

例えば製品構成と業務項目がリンクしているモデル



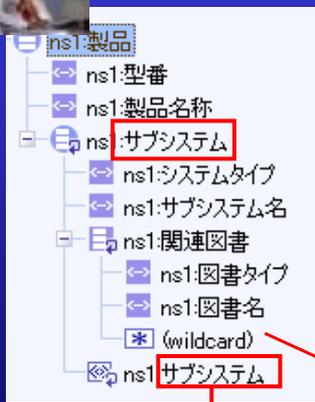
製品構成の各レイヤごとに
関連文書、担当者、業務内容
等の項目がある。
製品や各レイヤによって項目数
や項目種類がまちまち。

文章作成も不確定モデルである



XMLは不確定モデル

スキーマの定義はすれど、構造(形)の確定はユーザ次第



何回要素を繰り返す?
どの要素を選ぶ?
どんな要素を作る?
どこまで下位階層を作る?

任意の繰り返し
任意の繰り返し
要素の選択
任意の要素

階層的繰り返し(再帰)



データ構造 (縦と横の広がり) はユーザ (現場) が決める

```

<?xml version="1.0" ?>
- <rs1:製品 xmlns:rs1="http://xmlns.afx.com/form/user/20070906175712735">
  <rs1:型番>112A</rs1:型番>
  <rs1:製品名称>新型モジュールFT-70pt</rs1:製品名称>
- <rs1:サブシステム>
  <rs1:システムタイプ>サブシステム</rs1:システムタイプ>
  <rs1:サブシステム名>制御基盤系</rs1:サブシステム名>
- <rs1:関連図書>
  <rs1:図書タイプ>図書</rs1:図書タイプ>
  <rs1:図書名>基礎制御図</rs1:図書名>
  </rs1:関連図書>
- <rs1:サブシステム>
  <rs1:システムタイプ>モジュール</rs1:システムタイプ>
  <rs1:サブシステム名>HV制御基盤</rs1:サブシステム名>
- <rs1:関連図書>
  <rs1:図書タイプ>基礎図書</rs1:図書タイプ>
  <rs1:図書名>PO制御97</rs1:図書名>
  </rs1:関連図書>
- <rs1:サブシステム>
  <rs1:システムタイプ>ロータ</rs1:システムタイプ>
  <rs1:サブシステム名>1Qモータ制御</rs1:サブシステム名>
- <rs1:関連図書>
  <rs1:図書タイプ>社務書</rs1:図書タイプ>
  <rs1:図書名>SH4221</rs1:図書名>
  </rs1:関連図書>
  </rs1:サブシステム>
</rs1:サブシステム>
</rs1:製品>
  
```

階層的繰り返し

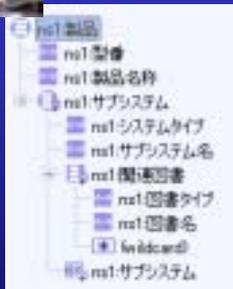
同列的繰り返し、選択



半定型データ = 定義 + 未定型

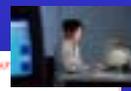
SEによるルール(義)の決定

ユーザによる形の決定



```

<?xml version="1.0" ?>
<rs1:製品 xmlns:rs1="http://xmlns.afx.com/form/user/20070906175712735">
  <rs1:型番>112A</rs1:型番>
  <rs1:製品名称>新型モジュールFT-70pt</rs1:製品名称>
  <rs1:サブシステム>
    <rs1:システムタイプ>サブシステム</rs1:システムタイプ>
    <rs1:サブシステム名>制御基盤系</rs1:サブシステム名>
    <rs1:関連図書>
      <rs1:図書タイプ>図書</rs1:図書タイプ>
      <rs1:図書名>基礎制御図</rs1:図書名>
    </rs1:関連図書>
  </rs1:サブシステム>
  <rs1:システムタイプ>モジュール</rs1:システムタイプ>
  <rs1:サブシステム名>HV制御基盤</rs1:サブシステム名>
  <rs1:関連図書>
    <rs1:図書タイプ>基礎図書</rs1:図書タイプ>
    <rs1:図書名>PO制御97</rs1:図書名>
  </rs1:関連図書>
  <rs1:サブシステム>
    <rs1:システムタイプ>ロータ</rs1:システムタイプ>
    <rs1:サブシステム名>1Qモータ制御</rs1:サブシステム名>
    <rs1:関連図書>
      <rs1:図書タイプ>社務書</rs1:図書タイプ>
      <rs1:図書名>SH4221</rs1:図書名>
    </rs1:関連図書>
  </rs1:サブシステム>
</rs1:製品>
  
```



XMLSchema定義 データ構造 (形) の決定
最終形はユーザに委ねられる

XMLの場合

SEによるスキーマ設計時点

ユーザによるデータエントリ時点



データ項目の出現ルールを定義する。
(スキーマ定義)

だけどデータの最終形は？

ルールの範囲で、自由にデータ項目を出現させる。
(データ構造の確定)

DB定義(設計)

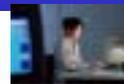
データ構造(形)の決定

47

半定型データ = 定義 + 未定形

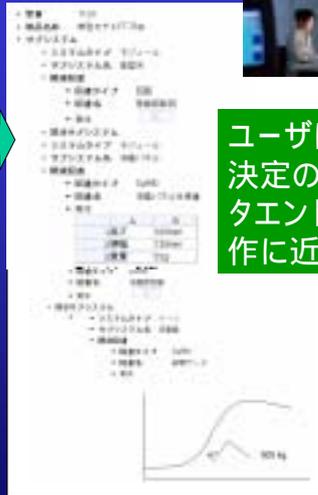
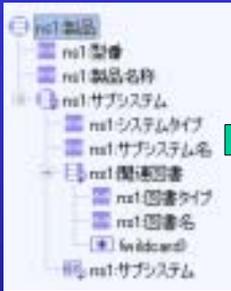
SEによるルール(義)の決定

ユーザによる形の決定



XMLSchema定義 データ構造(形)の決定
最終形はユーザに委ねられる

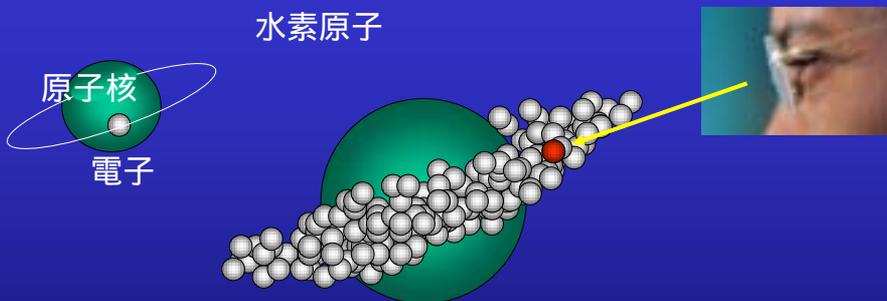
半定型データの定型化は創造的



ユーザにデータ構造とデータ値決定の裁量権があるので、データエントリというよりデータの著作に近い。



XML定義は電子雲のようなもの



観測者が見るまで電子の位置が確定できない。
電子雲は確率の軌跡である。

XMLのスキーマ定義もユーザがタグ・パターンを
確定するまでは、不確定な定義である。

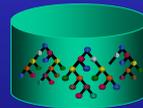
不確定モデルに耐える技術

3つのeXtensible
(拡張性が前提)

eXtensibleなデータ (XML)



スキーマレスなDB (XML DB)



eXtensibleなユーザインターフェース

動的に、階層的にも
フィールドを拡張可能、
他のXML (UI) もマッシュ
アップ (混成) 可能。



最後に

新しい酒は新しい革袋に容れよ (新約聖書)



XMLDBはシステムアイディアの起爆剤
ユーザーに、新たな利用領域を独創する
ことを求めている。





XMLコンソーシアム・XMLDB部会関連

<http://www.xmlconsortium.org/>
wg-xml-db-ldr@xmlconsortium.org

xfy・XMLDBシステム関連

<http://www.xfy.com/jp/>
tetsuyoshi.kato@justsystems.com

株式会社ジャストシステム コンサルティング部
加藤 哲義 03-5412-3895

本書に記載されている商品名、会社名などの固有名詞は、各社の商標及び登録商標です。
本書の内容の一部またはすべてを、当著作者の事前の許可なく複製や再利用することを禁じます。