

DB2 V8 XML機能

- **SQL/XML**
 - ▶ 表に格納された2次元データをXMLとして出力する標準規格
 - XMLドキュメントを操作するためのUDF群
- **XMLExtender**
 - ▶ XMLドキュメントをデータベース・ストレージに格納する手段を提供
 - XML型のサポート(XMLCollection, XMLColumn)
 - XML UDFのサポート
- **Information Integrator V8.1**
 - ▶ 既存のXMLドキュメントをDBに格納することなく表形式に変換して出力
 - XML Wrapper in Information Integrator
- **Web Services**
 - ▶ ノンプログラミングで実現するDB2 Web Services機能
 - Web Services Provider機能
 - Web Services Consumer機能

SQL/XML

■ SQL/XML ISO国際標準ISO/IEC 9075-14:2003

IBMを含めたSQLX Informal Group of Companiesで開発されANSIに移管

▶ XMLドキュメント作成・変換のためのUDF群を用意

- XMLELEMENT

- XMLの要素を作成。引数として要素名、属性のコレクションを取る。XMLデータタイプをリターンする。

- XMLATTRIBUTE

- XMLの属性を作成。引数として属性として使用する列を指定。

- XMLAGG

- 引数として指定されているXMLデータタイプを連結。

- XML2CLOB

- 引数で指定されている内容をCLOBとしてリターン

- REC2XML

- XML タグを使って形式設定された、列名および列データを含むSTRINGをリターン

SQL/XML

■ SQL/XML 例 1)

Ex.)

```
SELECT XML2CLOB(
  XMLELEMENT(NAME "Employees",
    XMLAGG( XMLELEMENT(NAME "Employee",
      XMLATTRIBUTES(EMPNO, FIRSTNME, LASTNAME, HIREDATE, EDLEVEL))))))
FROM employee
```

```
<Employees>
  <Employee EMPNO="000010" FIRSTNME="CHRISTINE" LASTNAME="HAAS" HIREDATE="1965-01-01" EDLEVEL="18" />
  <Employee EMPNO="000020" FIRSTNME="MICHAEL" LASTNAME="THOMPSON" HIREDATE="1973-10-10" EDLEVEL="18" />
  <Employee EMPNO="000030" FIRSTNME="SALLY" LASTNAME="KWAN" HIREDATE="1975-04-05" EDLEVEL="20" />
  <Employee EMPNO="000050" FIRSTNME="JOHN" LASTNAME="GEYER" HIREDATE="1949-08-17" EDLEVEL="16" />
  <Employee EMPNO="000060" FIRSTNME="IRVING" LASTNAME="STERN" HIREDATE="1973-09-14" EDLEVEL="16" />
  <Employee EMPNO="000070" FIRSTNME="EVA" LASTNAME="PULASKI" HIREDATE="1980-09-30" EDLEVEL="16" />
  <Employee EMPNO="000090" FIRSTNME="EILEEN" LASTNAME="HENDERSON" HIREDATE="1970-08-15" EDLEVEL="16" />
  <Employee EMPNO="000100" FIRSTNME="THEODORE" LASTNAME="SPENSER" HIREDATE="1980-06-19" EDLEVEL="14" />
  <Employee EMPNO="000110" FIRSTNME="VINCENZO" LASTNAME="LUCCHESSI" HIREDATE="1958-05-16" EDLEVEL="19" />
  <Employee EMPNO="000120" FIRSTNME="SEAN" LASTNAME="O'CONNELL" HIREDATE="1963-12-05" EDLEVEL="14" />
  <Employee EMPNO="000130" FIRSTNME="DOLORES" LASTNAME="QUINTANA" HIREDATE="1971-07-28" EDLEVEL="16" />
  .....
</Employees>
```

SQL/XML

■ SQL/XML 例 2)

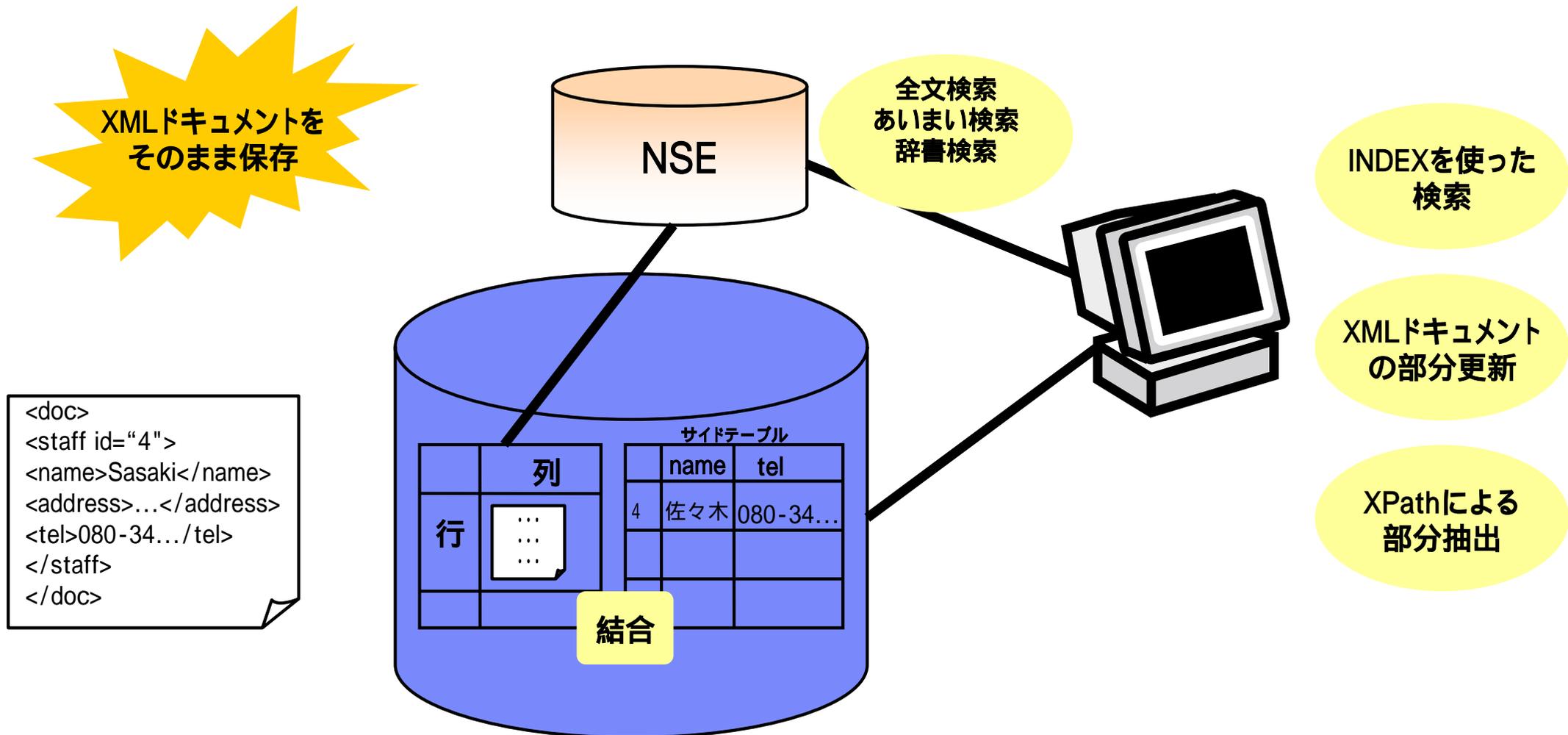
Ex.)
SELECT REC2XML (1.0, 'COLATTVAL', "",
EMPNO, FIRSTNME, LASTNAME, HIREDATE, EDLEVEL)
FROM employee

```
<row>  
  <column name="EMPNO">000010</column>  
  <column name="FIRSTNME">CHRISTINE</column>  
  <column name="LASTNAME">HAAS</column>  
  <column name="HIREDATE">1965-01-01</column>  
  <column name="EDLEVEL">18</column>  
</row>  
  
<row>  
  <column name="EMPNO">000020</column>  
  <column name="FIRSTNME">MICHAEL</column>  
  <column name="LASTNAME">THOMPSON</column>  
  <column name="HIREDATE">1973-10-10</column>  
  <column name="EDLEVEL">18</column>  
</row>  
.....  
.....
```

XML Extender

▶ XMLColumn

- XMLドキュメントをそのままの形でCLOB型のカラムに挿入するイメージ
- 検索の効率を良くするためにインデックスを格納するためのサイドテーブルを作成
- **Net Search Extender**と併用することで、ハイスピードな全文検索を実現できる。



XML Extender

▶ XMLColumn作成手順

1. DADファイルの定義

- DTDによるValidationの有無
- サイドテーブルの構成

- `<table name = "side_XXX">`

- `<column name = "XXX" type="VARCHAR(20)" path = "/root/xxx/@xxx" />`

2. XMLエクステンダーの有効化

3. テーブルの作成

- `CREATE TABLE TABXML (ID INT, XMLDOC DB2XML.XMLCLOB)`

4. DTDの登録(Optional)

5. DADファイルの登録、XML Columnの有効化

- `dxxadm enable_column DB名 TABXML XMLDOC 1`で作成したDADのパス

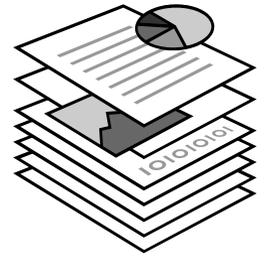
6. インデックスの作成

- 1で作成したサイドテーブルの列に対してインデックスを張る
 - `CREATE INDEX idx_xxx ON side_XXX(XXX)`

7. XMLドキュメントの挿入

- `INSERT INTO TABXML(ID, XMLDOC) VALUES("1", "DB2XML.XMLCLOBFROMFILE("パス名"))`

DB2 Net Search ExtenderによるXColumn検索



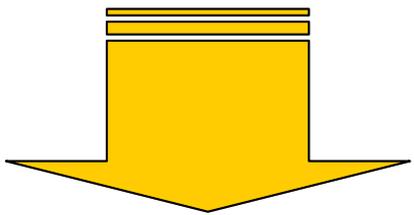
Net Search Extender



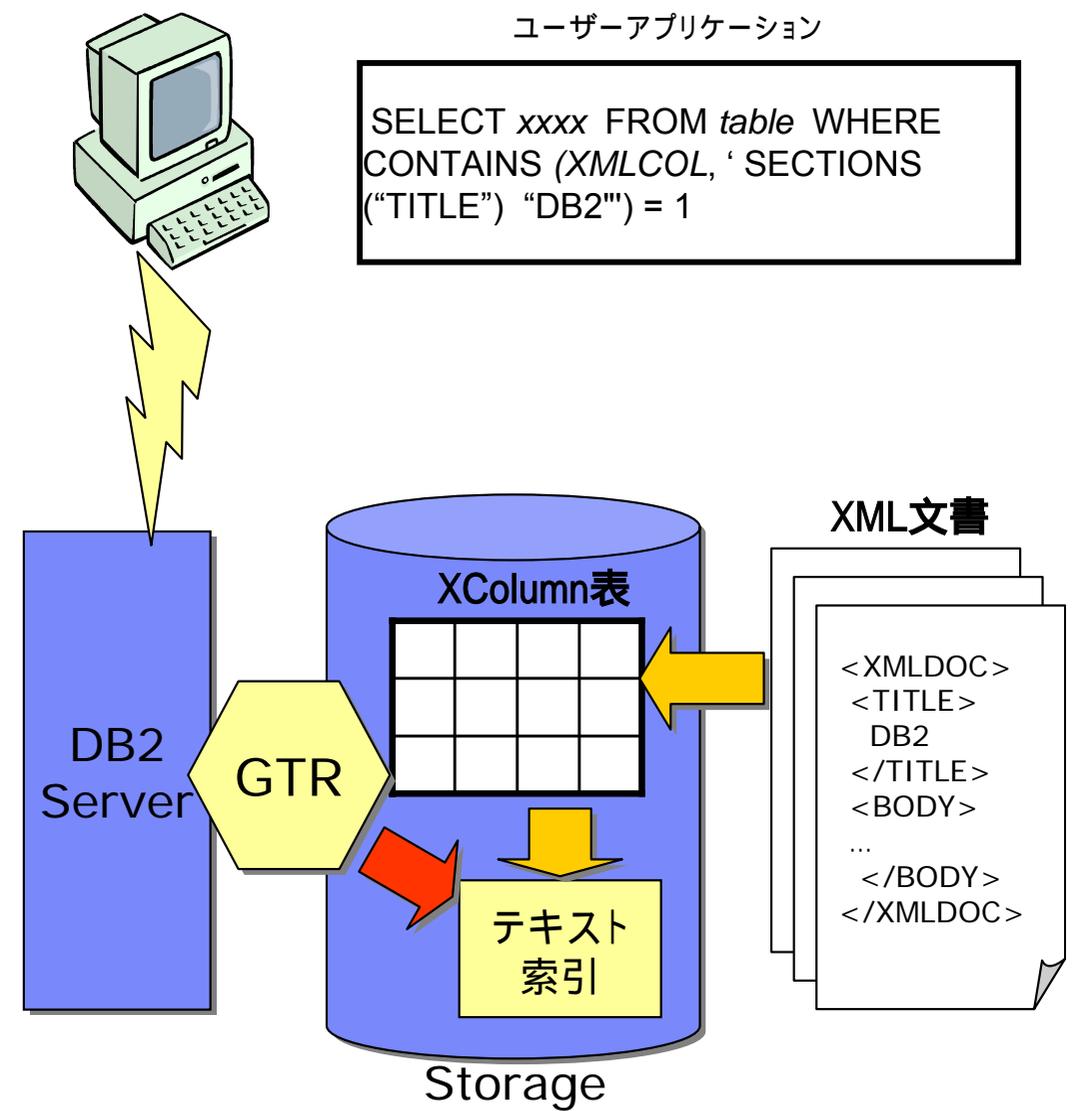
高速で対象範囲が広く、インテリジェントな全文検索ができるツールはないのか？

NSEの特色

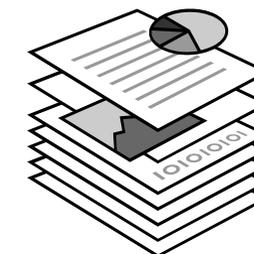
- インテリジェントな自然語検索
- 高速なテキスト検索ソリューション
- SQL インターフェースによる検索
- 運用の容易性



DB2 Net Search Extender は、SQL 照会を使用して DB2 やその他のデータベース、およびファイル・システムに保管されているフルテキスト文書を検索するための、高速で多様なインテリジェント方式をユーザーとアプリケーション・プログラマーに提供します。



DB2 Net Search ExtenderによるXColumn検索



Net Search Extender

SQL 検索との機能比較

	データベース標準 SQL	Net Search Extender
スキャン方法	●表スキャン or 索引スキャン	●索引スキャン
考慮点	<ul style="list-style-type: none"> ●データ量が大きくなるとパフォーマンスが劣化する (表スキャン) ●索引用のディスクスペースが必要(索引スキャン) ●索引のメンテナンスが必要 (索引スキャン) 	<ul style="list-style-type: none"> ●索引用のディスクスペースが必要 ●索引のメンテナンスが必要
完全一致検索	(LIKE述部の使用)	
大文字小文字を区別しない検索	×	
あいまい検索	×	
シソーラス検索	×	
構造化文書の検索		

XML Extender

▶ XMLCollection

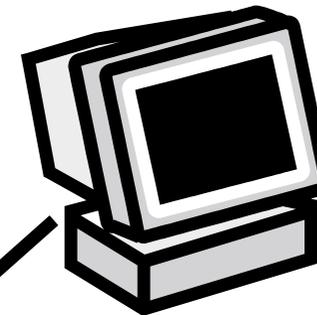
- XMLドキュメントの要素を分解し、要素ごとに列に格納する。
- 列・行に分解されてリレーショナル表に格納されるので検索/挿入/更新のパフォーマンスがよい

XMLドキュメントを
分解して保存

```
<doc>  
<staff id="4">  
<name>Sasaki</name>  
</staff>  
</doc>
```

4
SASAKI

	ID	NAME
	4	SASAKI



INDEXを使った
検索

SQLによる
挿入/更新/削除

XMLとして
出力

```
<doc>  
<staff id="4">  
<name>Sasaki</name>  
</staff>  
</doc>
```

XMLドキュメントに
合成して出力

XML Extender

▶ XML Collection作成手順

1. DADファイルの定義 -リレーショナル<->XMLデータマッピング

- DTDによるValidationの有無
- XML要素・属性とRDBMS表・列のマッピング
 - ```
<element_node name="EMPLOYEE">
 <attribute_node name="empid">
 <RDB_node> <table name="employee"/> <column name="empid"/>
 </RDB_node>
 </attribute_node>
 <element_node name="firstname">
 <RDB_node> <table name="employee"/> <column name="firstname"/>
 </RDB_node>
 </element_node>
```

### 2. XMLエクステンダーの有効化

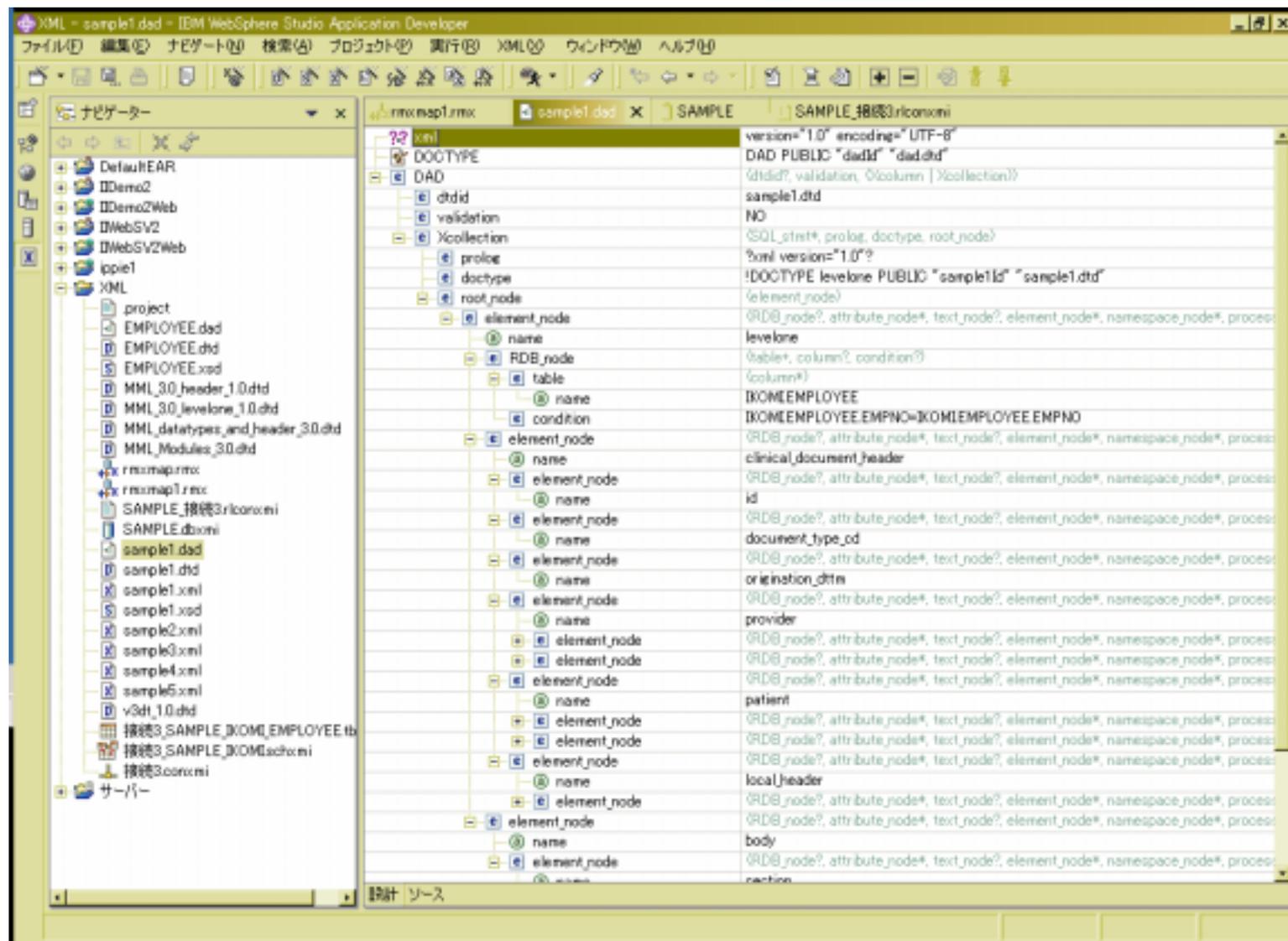
### 3. DTDの登録

### 4. DADファイルの登録、XML Collectionの有効化

- `dxxadm enable_collection DB名 コレクション名 DADのパス`

# XML Extender

WSAD (WebSphere Studio Application Developer)により、既存のDB2表からマッピングスキーマDADファイルを自動生成



# XML Extender

## ■ XMLCollection vs. XML Column

### ▶ どちらの格納タイプを使用するか

#### • XML Column

- ドキュメントが既に存在してて、そのままの形で保存したいとき
- 参照されるのが主で更新はほとんど無い
- 主に参照される要素や属性が決定しているとき
- DB2以外で外部ファイルとして扱いたいとき

#### • XML Collection

- 既存のリレーショナルデータからXMLを作成したいとき
- 更新が頻繁に行われる
- 参照される要素や属性がいろいろあるとき

# XML Extender

## ▶ XMLを格納、検索、更新、検査するためのユーザー定義関数(UDF)群

### • 格納

- XMLVarcharFromFile(), XMLCLOBFromFile()  
外部のXMLファイルを読み込み、DB2のXML型に変換して格納。

ファイルからXML  
カラムに格納

### • 検索

- extractInteger(), extractVarchar(),extractTime() etc.  
XPathにより指定したエレメントやアトリビュートの値を抽出

XPathによる  
部分検索

### • 更新

- Update()  
指定したエレメントやアトリビュートの値を、XML文書で更新

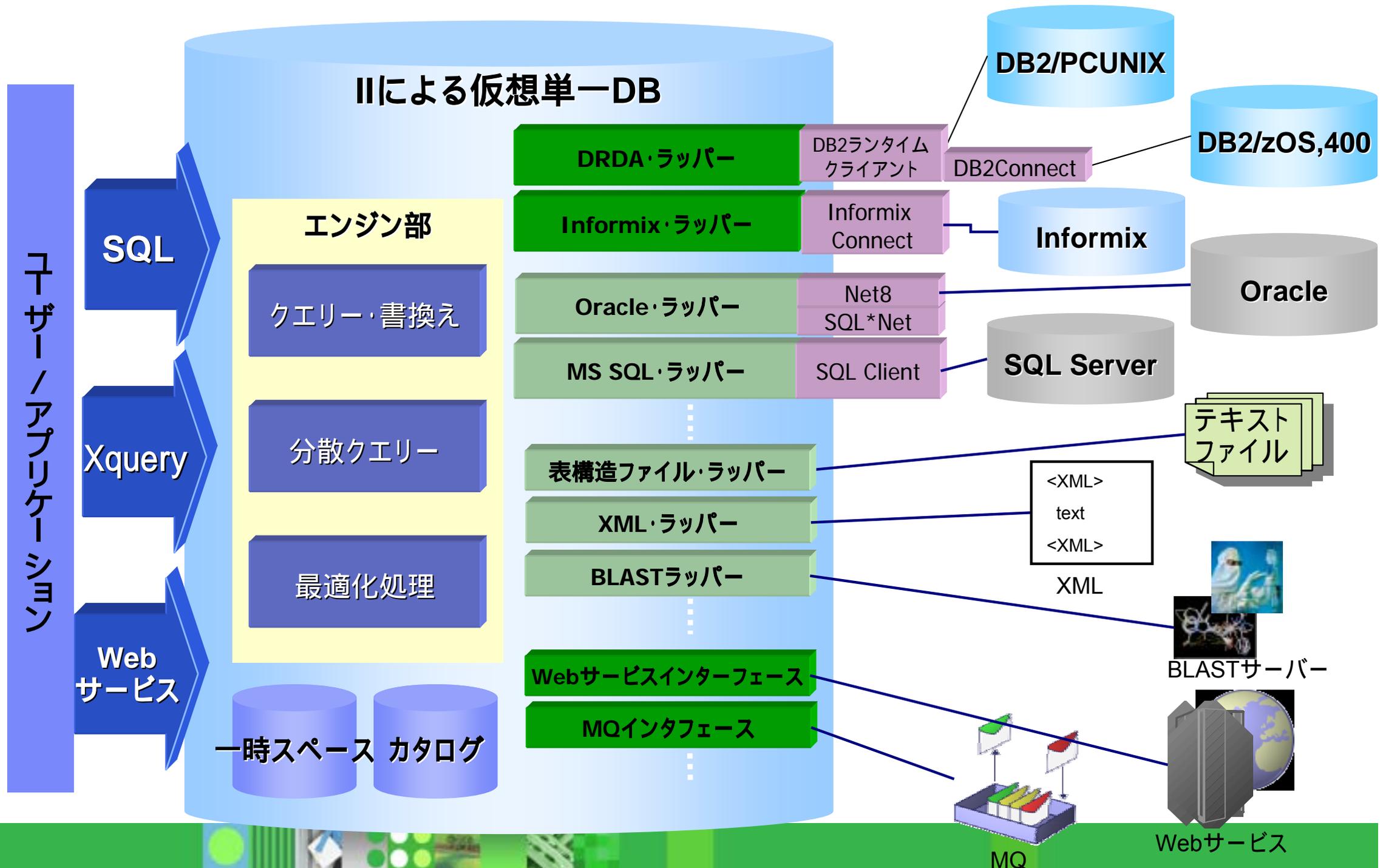
XMLドキュメント  
を使用した更新

### • 検証

- SVALIDATE()  
XMLSchemaを使ったXMLドキュメントの妥当性検査
- DVALIDATE()  
DTDを使ったXMLドキュメントの妥当性検査

XMLドキュメント  
の検証

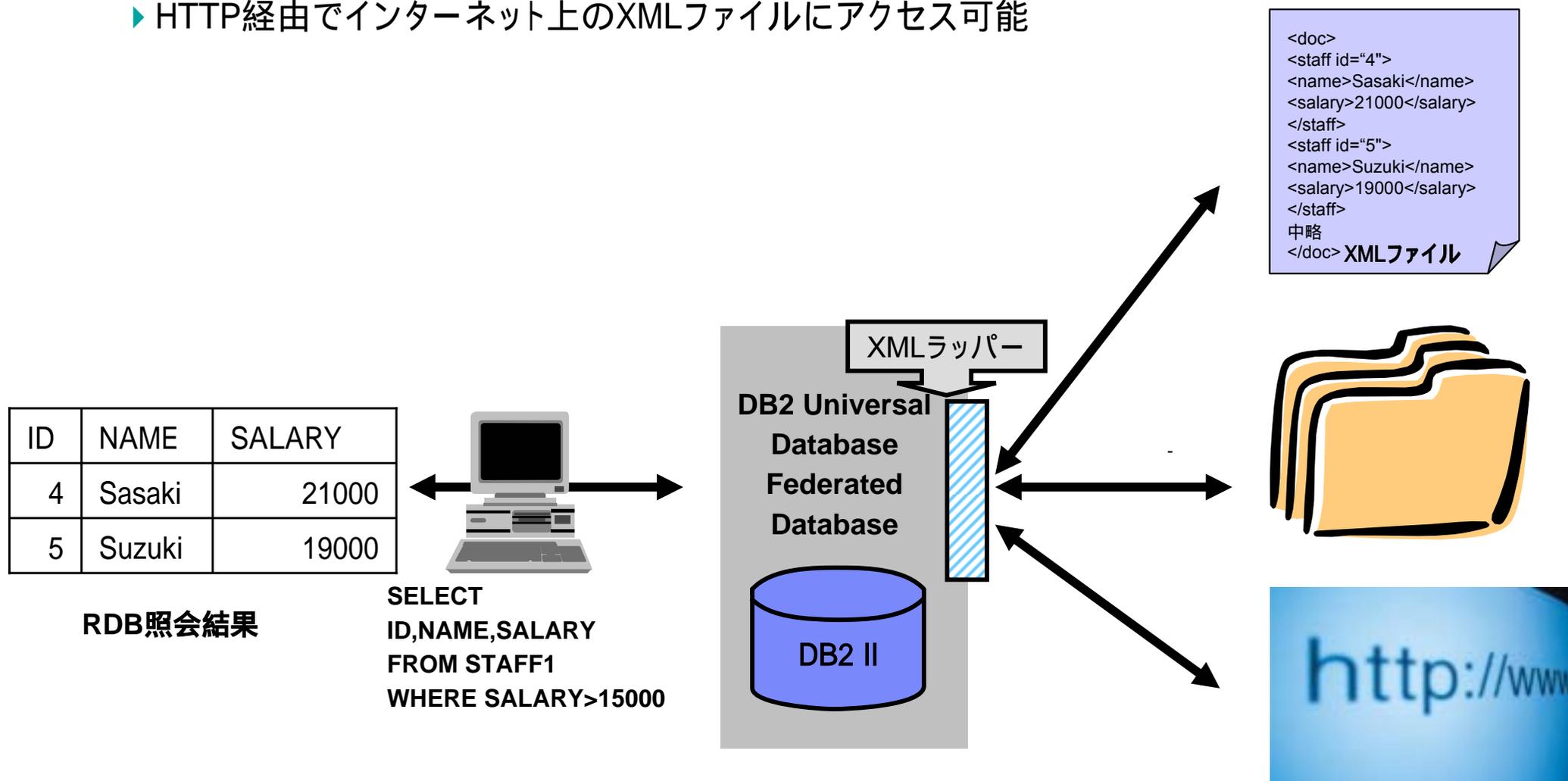
# DB2 Information Integrator V8.1



# DB2 Information Integrator V8.1

## ■ XMLラッパー

- ▶ 外部にあるXMLドキュメントを行と列に分解し、SQLによってXMLドキュメントにアクセス可能
  - 要素・属性と列のマッピングはウィザードを使用し自動的にマッピング
- ▶ HTTP経由でインターネット上のXMLファイルにアクセス可能



## DB2 Information Integrator V8.1

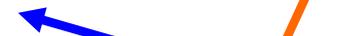
部門表(ORG\_NN)

| DEPTNUMB | DEPTNAME    | ... |
|----------|-------------|-----|
| 10       | Head Office |     |
| 15       | New England |     |
| ...      | ...         |     |

従業員表(STAFF\_NN)

| ID  | NAME     | ... | DEPTNUMB |
|-----|----------|-----|----------|
| 160 | Molinare |     | 10       |
| 210 | Lu       |     | 10       |
| ... | ...      |     |          |
| 50  | Hanes    |     | 15       |
| ... | ...      |     |          |

|                                                                                                                                                       |
|-------------------------------------------------------------------------------------------------------------------------------------------------------|
| <doc>                                                                                                                                                 |
| <org deptnumb="10"><br><deptname>Head Office</deptname><br><manager>160</manager><br><division>Corporate</division><br><location>New York</location>  |
| <staff id="160"><br><name>Molinare</name><br><job>Mgr</job><br><years>7</years><br><salary>22959.20</salary><br><comm /><br></staff>                  |
| <staff id="210"><br><name>Lu</name><br><job>Mgr</job><br><years>10</years><br><salary>20010.00</salary><br><comm /><br></staff>                       |
| <staff id="240"><br>中略<br></staff><br><staff id="260"><br>中略<br></staff><br></org>                                                                    |
| <org deptnumb="15"><br><deptname>New England</deptname><br><manager>50</manager><br><division>Eastern</division><br><location>Boston</location><br>後略 |
| </doc>                                                                                                                                                |



- 1つのXMLは、複数のRDBテーブルのデータ構造になることもある。
- RDBでは、「従業員がある部門に所属している」という関係を表すために、従業員表の中に、部門表との紐付けを行う外部キーフィールドを定める。

## XMLとRDBのマッピング

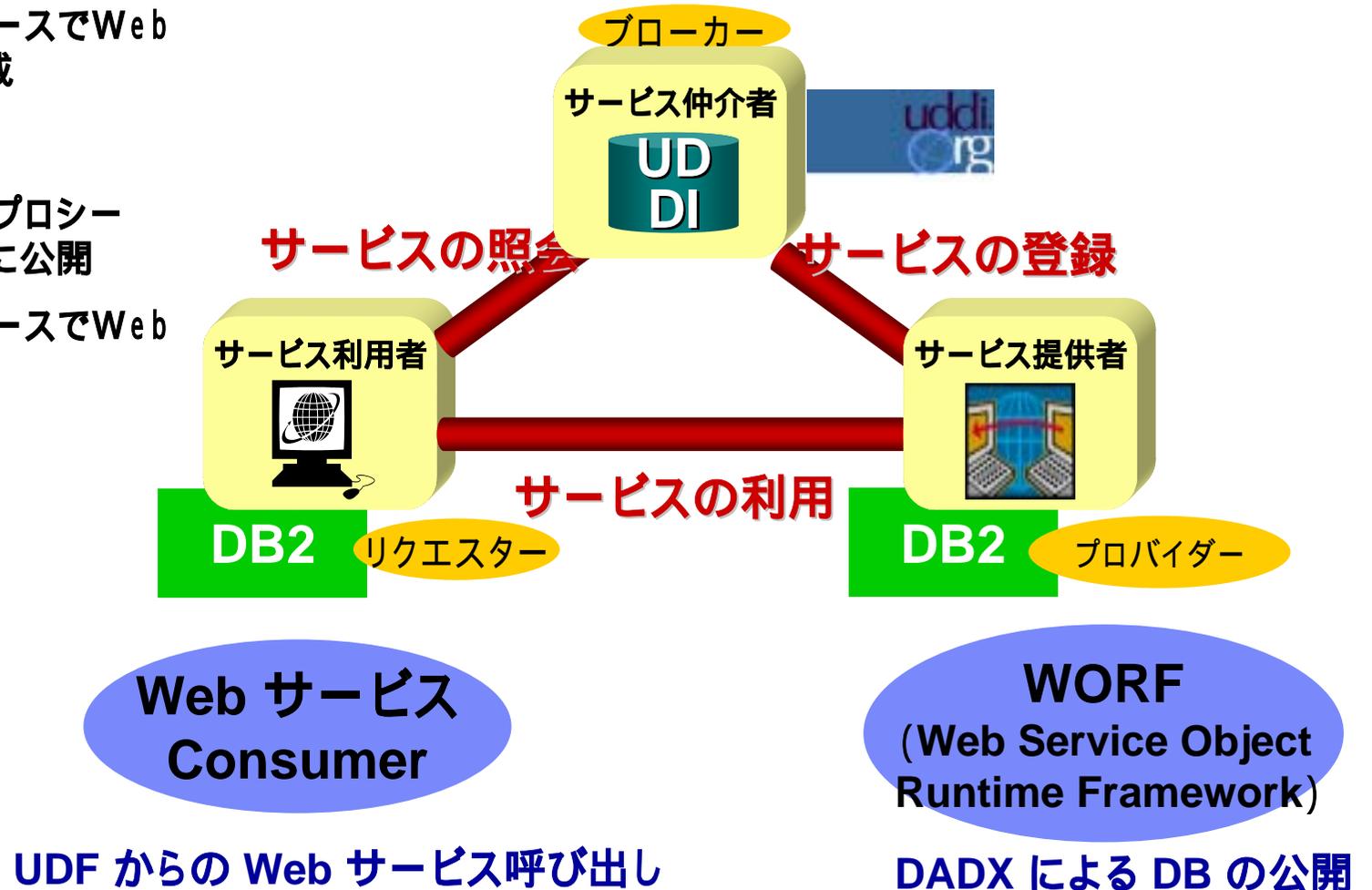
# Web Services

## Web Services Consumer機能

- SQLからWebサービス呼び出し、動的な値を取得できる
- WSADと連携して、ウィザードベースでWebサービスユーザー定義関数を作成

## Web Services Provider機能

- DB2内の表のデータやストアドプロシージャをWeb Servicesとして外部に公開
- WSADと連携して、ウィザードベースでWebサービス・サーブレットを作成



# Web Services

## Consumer機能 WEBサービスをSQL文から呼び出せる

```
CREATE FUNCTION getStock (SYMBOL VARCHAR(4)) RETURN DECIMAL(7,2)
```

```
.....
db2xml.soaphttpv(
 'http://localhost:9080/stock1/DADX/stock1.dadx/SOAP',
 'http://tempuri.org/DADX/stock1.dadx',
 <m:getStock xmlns:m="http://tempuri.org/DADX/emp1.dadx"
 SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
 <symbol xsi:type="xsd:string"> '000010' </symbol>
 </m:getLastname>
);
```

Webサービス呼び出し

```
...
END
```

| 会社名 | 住所  | 電話番号   | SYMBOL | ... |
|-----|-----|--------|--------|-----|
| IBM | 日本  | 090XXX | IBM    | ... |
| ... | ... | ...    | ...    | ... |



株価取得  
Webサービス

ex.) SELECT 会社名、住所、getStock(?) as 株価 FROM 会社表 WHERE SYMBOL = ?

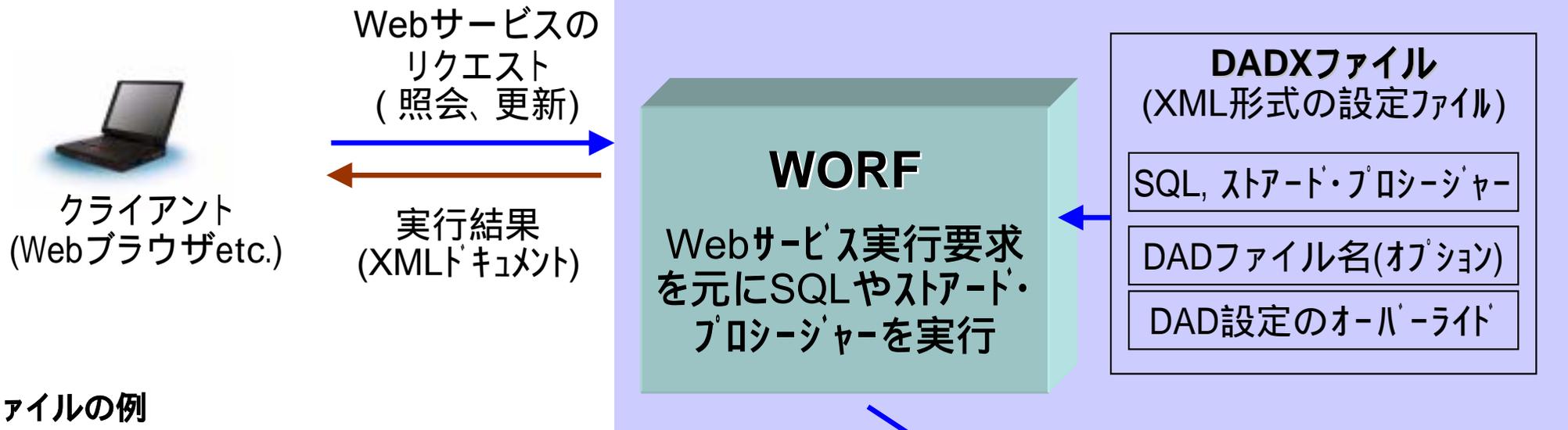
| 会社名 | 住所 | 株価   |
|-----|----|------|
| IBM | 日本 | 81.5 |

会社テーブル経由の静的なデータ

WEBサービス経由の動的なデータ

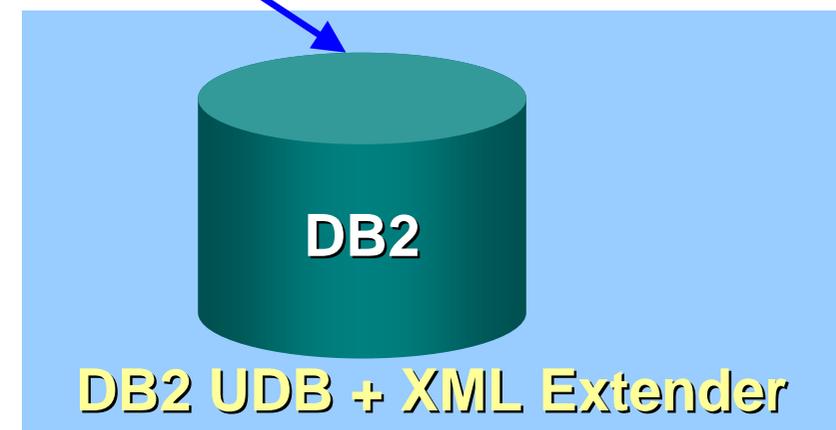
# Web Services

Provider機能 (WORF) DB2のデータをWebサービスとして公開することが出来るノンプログラミ  
ンクアーキテクチャー



## DADXファイルの例

```
<?xml version="1.0" encoding="UTF-8"?>
<DADX xmlns="http://schemas.ibm.com/db2/dxx/dadx"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" >
 <operation name="getWeather">
 <query>
 <SQL_query>
 select wtext from weather
 where wdate = :wDate and wpref= :prefName
 </SQL_query>
 <parameter name="wDate" type="xsd:date" />
 <parameter name="prefName" type="xsd:string" />
 </query>
 </operation>
</DADX>
```



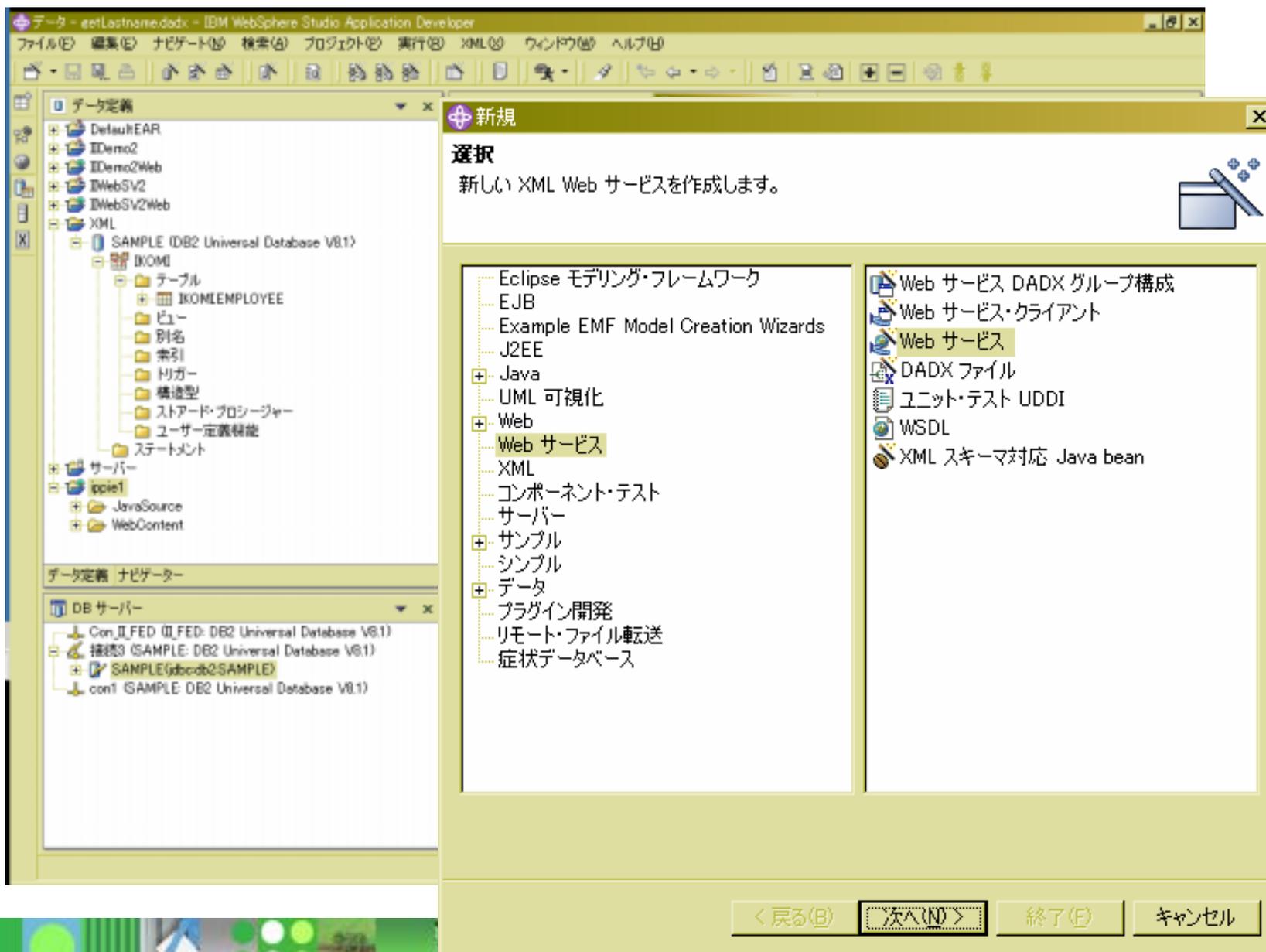
The screenshot shows a Microsoft Internet Explorer window with the following details:

- Address Bar:** `http://localhost:9080/ippie1/DADX/getLastname1.dadx/TEST`
- Page Title:** WOLF Test Page
- Navigator:** Shows a tree view with 'ID' selected, containing links for 'Operations', 'Description', 'WSDL', 'WSDL Binding', 'WSDL Service', 'XML Schema', and 'Documentation'. The 'Documentation' section notes: "There is no documentation available service."
- Actions:** Displays the 'getLastname Operation' with a text input field for 'string empno:' containing the value '000010'. There are 'Invoke' and 'Clear' buttons.
- Results:** Shows the XML response:

```
<?xml version="1.0" ?>
- <xsd1:getLastnameResponse
 xmlns:xsd1="http://tempuri.org/ippie1/DADX/getLastname1.dadx/XSD" xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
 xmlns:xsd="http://www.w3.org/2001/XMLSchema"
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
- <return>
 - <xsd1:getLastnameResult
 xmlns:xsd1="http://tempuri.org/ippie1/DADX/getLastname1.dadx/XSD"
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
 - <getLastnameRow>
 <LASTNAME>HAAS</LASTNAME>
 </getLastnameRow>
</getLastnameResponse>
```
- Status Bar:** Displays the message "ページが表示されました" (Page displayed) and "イントラネット" (Intranet).

# Web Services

WSAD (WebSphere Studio Application Developer)に付属のWORFプラグインにより、Wizardベースで簡単にWebサービスの構築ができるようになります。



# Backup

- **SQL/XML**
  - ▶ **SQL Reference Vol.1**
    - [http://www-306.ibm.com/cgi-bin/db2www/data/db2/udb/winos2unix/support/v8pubs.d2w/en\\_main](http://www-306.ibm.com/cgi-bin/db2www/data/db2/udb/winos2unix/support/v8pubs.d2w/en_main)
  - ▶ **Tutorial**
    - [https://www6.software.ibm.com/reg/devworks/dw-db2sqlx-i?S\\_TACT=102B7W84&S\\_CMP=DB2DD](https://www6.software.ibm.com/reg/devworks/dw-db2sqlx-i?S_TACT=102B7W84&S_CMP=DB2DD)
  - ▶ **IBM System Journals Vol.41 XML programming with SQL/XML and XQuery**
    - <http://www.research.ibm.com/journal/sj/414/reinwald.pdf>
- **XMLExtender**
  - ▶ **Homepage**
    - <http://www-306.ibm.com/software/data/db2/extenders/xmlxt/>
  - ▶ **Manual**
    - <http://publib.boulder.ibm.com/series/v5r2/ic2924/books/sm04/c2711722.pdf>
  - ▶ **Redbook**
    - <http://www.redbooks.ibm.com/redpapers/pdfs/redp0135.pdf>
  - ▶ **カンタン！DB2テクテク第一歩**
    - <http://www-6.ibm.com/jp/software/data/developer/column/kantanextend/05xmlxtender/01.html>
- **Information Integrator V8.1**
  - ▶ **Home Page**
    - <http://www-6.ibm.com/jp/software/data/ii/>
  - ▶ **Developer Domainインフォメーション・インテグレーション**
    - <http://www-6.ibm.com/jp/software/data/developer/features/ii/01.html>
- **Web Services**
  - ▶ **IBM System Journal Vol.41 DB2 and Web services**
    - <http://www.research.ibm.com/journal/sj/414/malaika.pdf>
  - ▶ **Developer Works**
    - <http://www-106.ibm.com/developerworks/db2/zones/webservices/worf/>

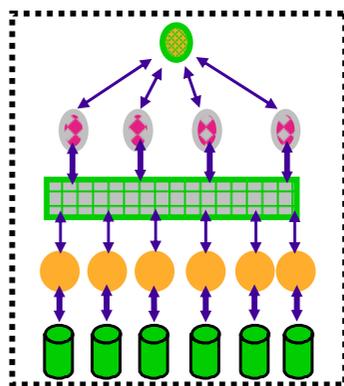
# 參考資料

# DB2 UDB の並列データベース・システム

- DB2 UDB ESE with Database Partitioning Feature による並列データベース・システム
  - ▶ シェアード・ナッシング・アーキテクチャによるリニアなスケーラビリティ
    - パーティション内並列処理と、パーティション間並列処理の同時使用
  - ▶ パーティション・データベースにアクセスするアプリケーションやユーザーは、データが物理的にどこにあるかを認識する必要なし

パーティション内並列処理  
(PE / WSE / ESE)

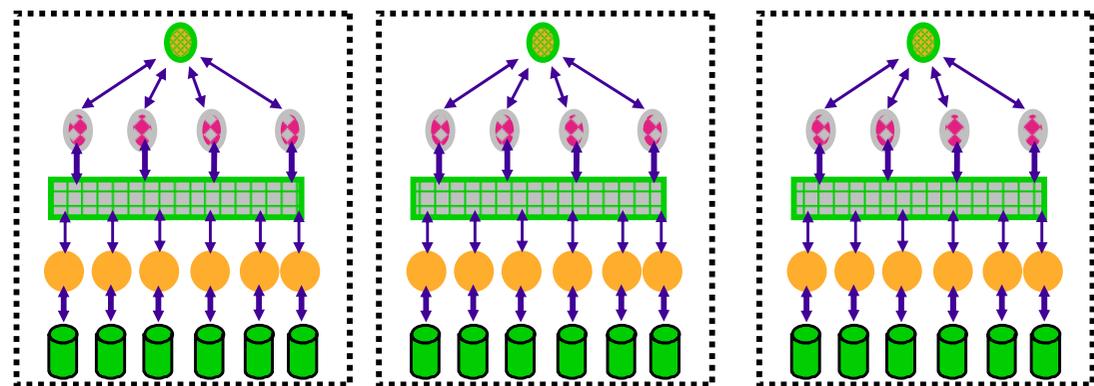
SMP



- 入出力
- 照会
- ユーティリティ

パーティション内&パーティション間並列処理

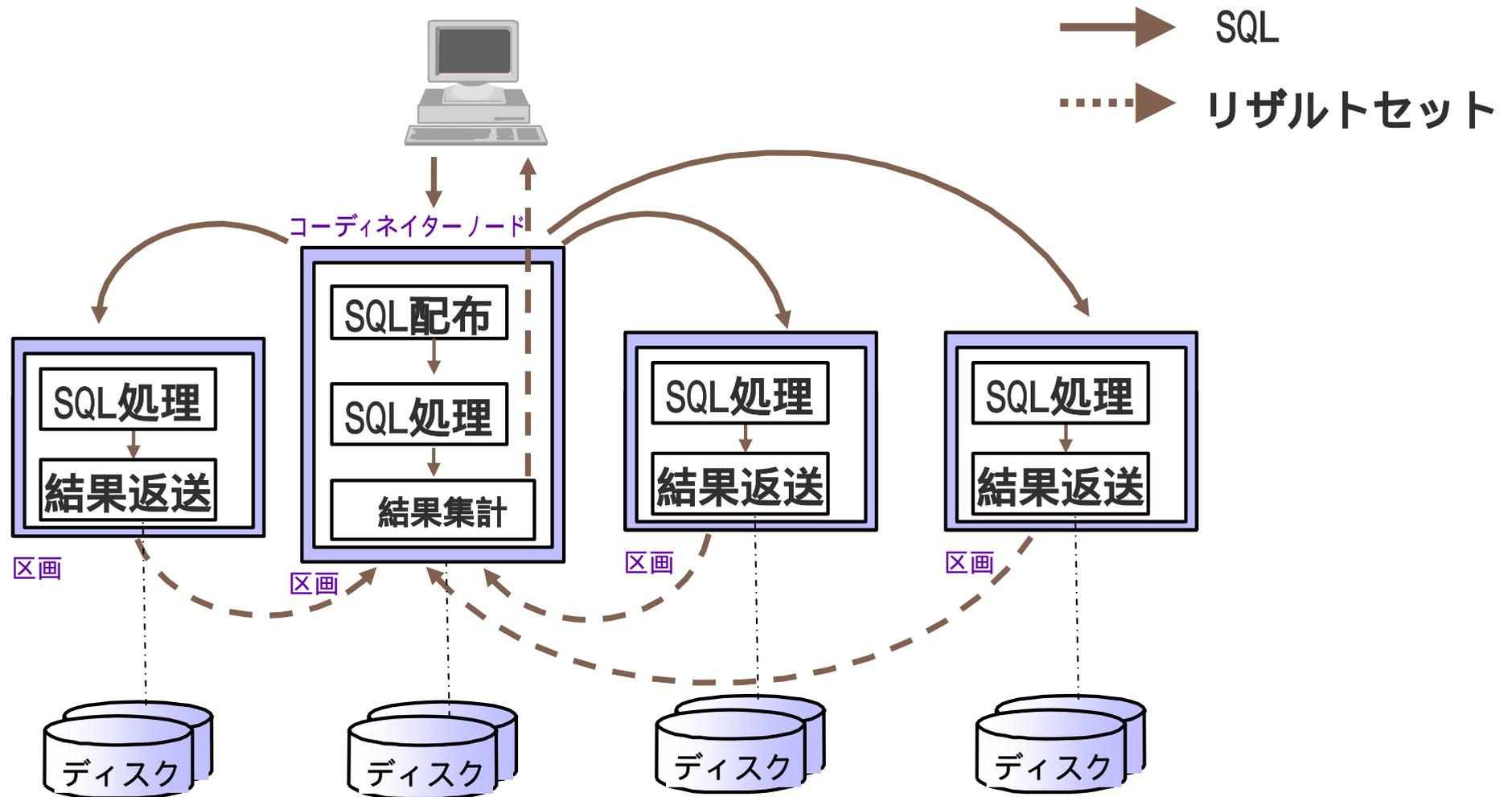
(ESE with DPF)  
SMP Cluster  
論理パーティション



- 入出力
- 照会
- ユーティリティ

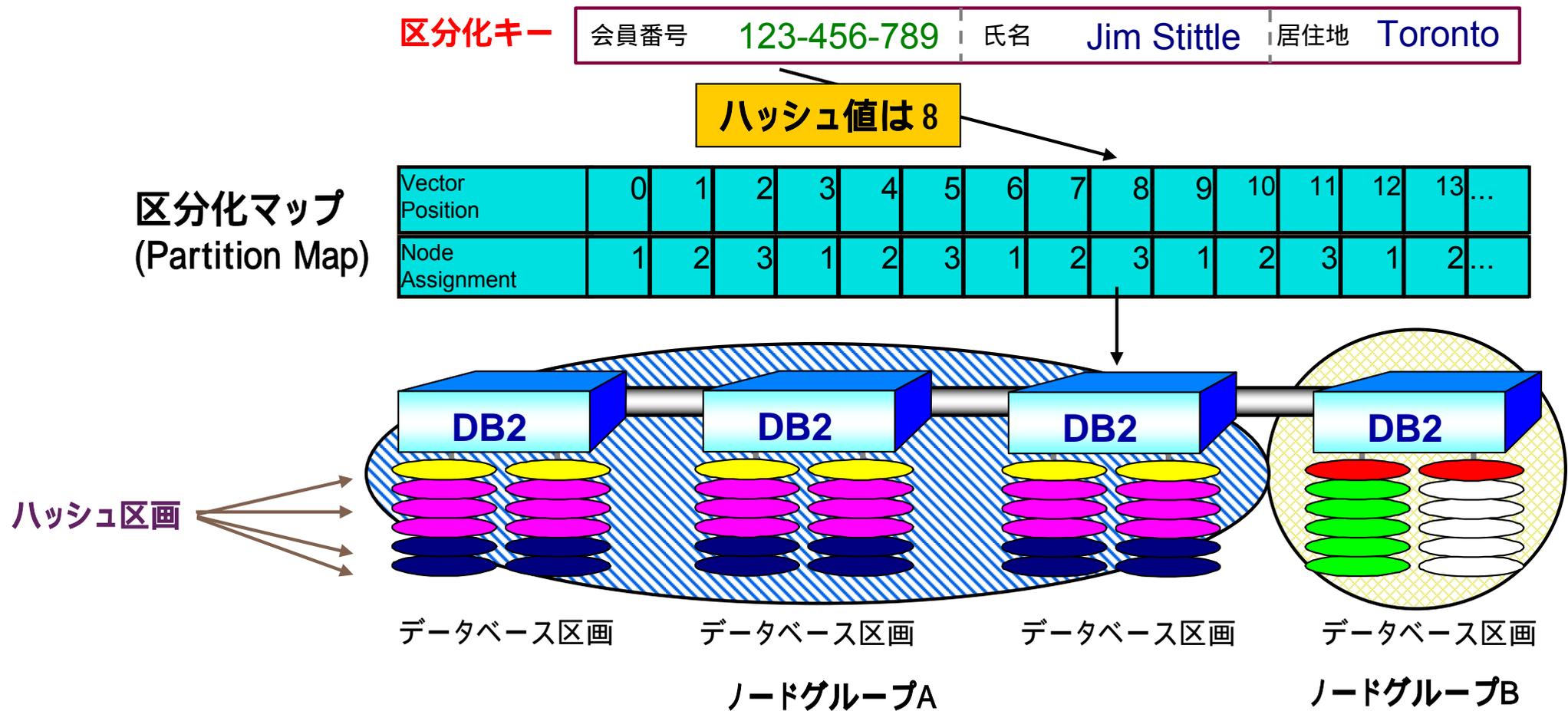
# 区画間並列処理

- コーディネーターノードが SQL を配布し、結果を集計



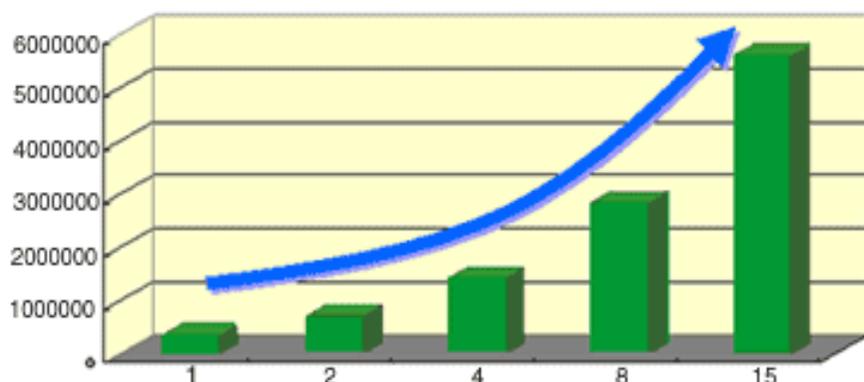
# ハッシュ・パーティショニング

- IBM 独自のハッシュ関数と区分化マップによるデータ分割
- 区分化マップをデータ内容・管理要求に応じて調整可能



## 国内で実施されたスケーラビリティ検証(DB2 ESE DPF)

- ▶ 検証にはIBM DB2 Integrated Cluster Environment for Linux(DB2 ICE)を用いました
- ▶ DB2 ICEは、ソフト、ハード、サービスを最適に統合した新しいクラスター・ソリューション。2ノードから1000ノードまでの拡張性を、リーズナブルな料金で構築できます

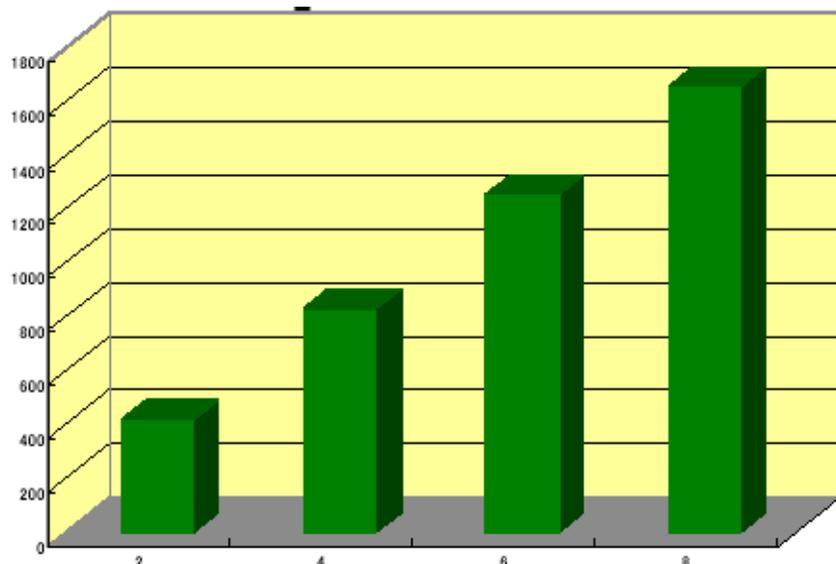


ノード数	スケーラビリティ
4	1.00
8	1.99
16	3.92

トランザクションはTPC-H like

IBM eServer XSeries 335  
CPU:Xeon 2.4GHz×2  
Memory:1GB  
HDD:36.4GB×2

Red Hat Linux 7.3  
DB2 UDBV8.1 ESE(FP3)



ノード数	スケーラビリティ
2	1.00
4	1.98
6	2.98
8	3.93

トランザクションはTPC-C like

IBM eServer XSeries 335  
CPU:Xeon 3.06GHz×2  
Memory:4GB  
HDD:36.4GB×1  
FASTT 900(73.4GB×4)

Red Hat Enterprise Linux 2.1AS  
DB2 UDBV8.1 ESE(FP3)

アプリ修正せずにスケール!