



IBM Software Group

DB2 & Web 2.0 Demos

(Course code: C2)

DB2 Information Management Software



© 2005 IBM Corporation

Agenda

- Introduction
- Two approaches for app development
- pureXML™ demos
 - ▶ db2vps application demo (pureXML comparison)
- Java
- Ruby on Rails
 - ▶ Ruby on Rails basics
 - ▶ Starter toolkit for DB2 on Rails
 - ▶ Demo
- PHP
 - ▶ PHP basics
 - ▶ ZendCore for IBM



Introduction

Introduction

■ IBM Toronto Laboratory

- ▶ Largest software facility in Canada
- ▶ 2,500 developers, engineers, architects
- ▶ 3rd largest in IBM Software Group
- ▶ Founder of CAS – CASCON
- ▶ Products:
 - DB2 (Distributed platforms)
 - Websphere tools
 - Rational Developer for Web/Java
 - Tivoli Intelligent Orchestrator

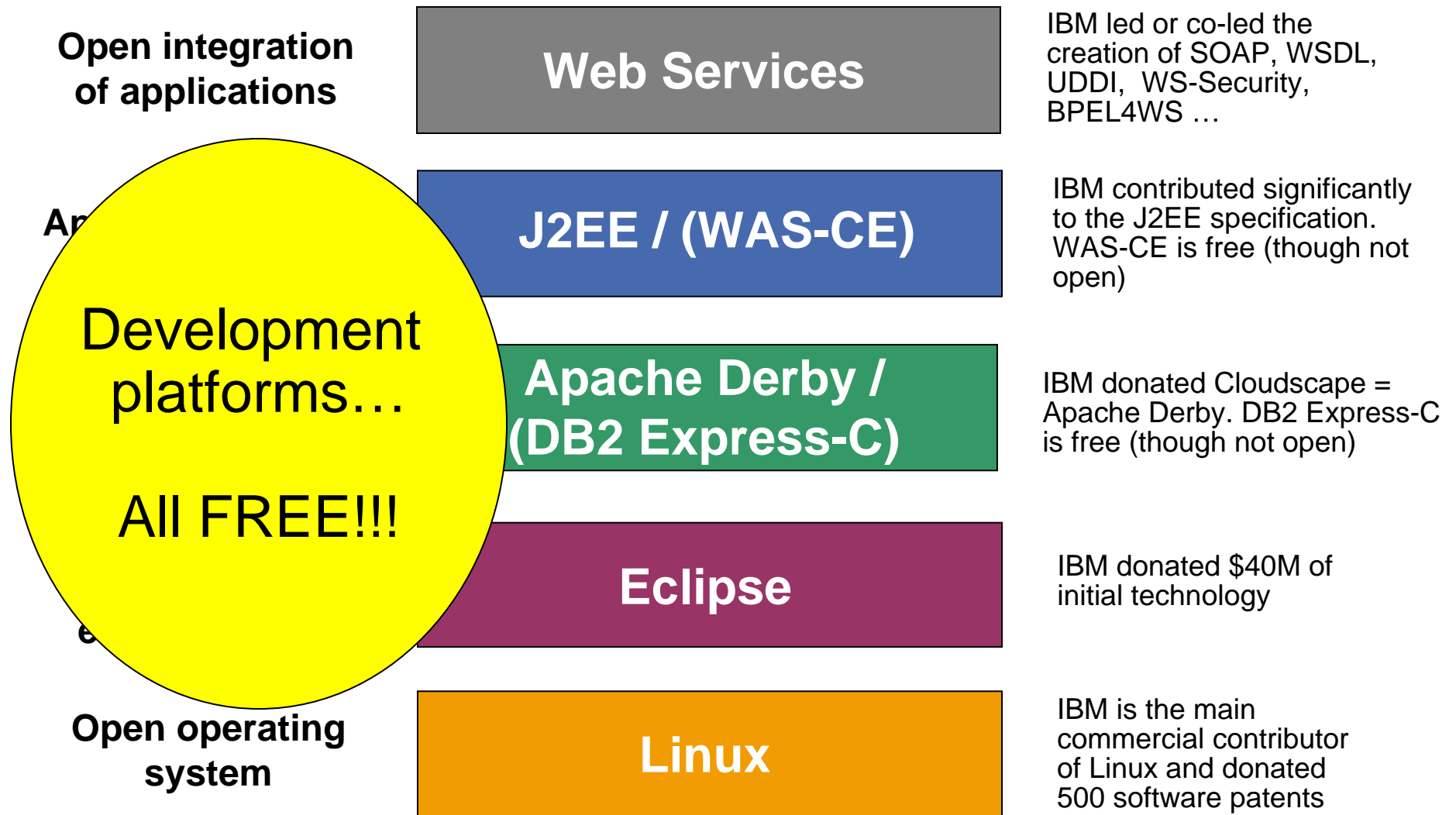


IBM Toronto Lab - From 55 people in 1967 to 2500 professionals with international mandates today

Two approaches



Free development platforms



Web 2.0 Languages: PHP & Ruby on Rails

Zendcore for IBM

- ▶ Environment ready for developing with PHP and DB2 Express-C: FREE!
- ▶ <http://www.ibm.com/software/data/info/zendcore>

■ **Startup Toolkit for DB2 on Rails**

- ▶ Environment ready for developing with Ruby, Rails and DB2 Express-C: FREE!
- ▶ <http://www.alphaworks.ibm.com/tech/db2onrails>

Application Development Freedom

- **Ruby on Rails**
- **C/C++ (ODBC and Static SQL)**
- **JDBC** and **SQLJ**
- **Borland**
- **PHP**
- **Perl**
- **.Net languages**
- **OLE-DB**
- **ADO**
- **Web Services**
- **SQL**
- **MS Office: Excel, Access, Word**

WebSphere



Zend The *php* Company

Rational[®]
the software development company

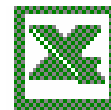


Microsoft
Visual Studio.net

Microsoft
Visual Studio

Kylix[™]
Rapid e-business development for Linux[®]

Borland[®] USA



AppForge

metrowerks[®]

IBM[®] DB2 Universal Database[™]
SQL Reference **IBM**
for Cross-Platform Development

<http://www7b.software.ibm.com/dmdd/library/techarticle/0206sqlref/0206sqlref.html>

pureXML™ Demos



Why use XML with Databases?

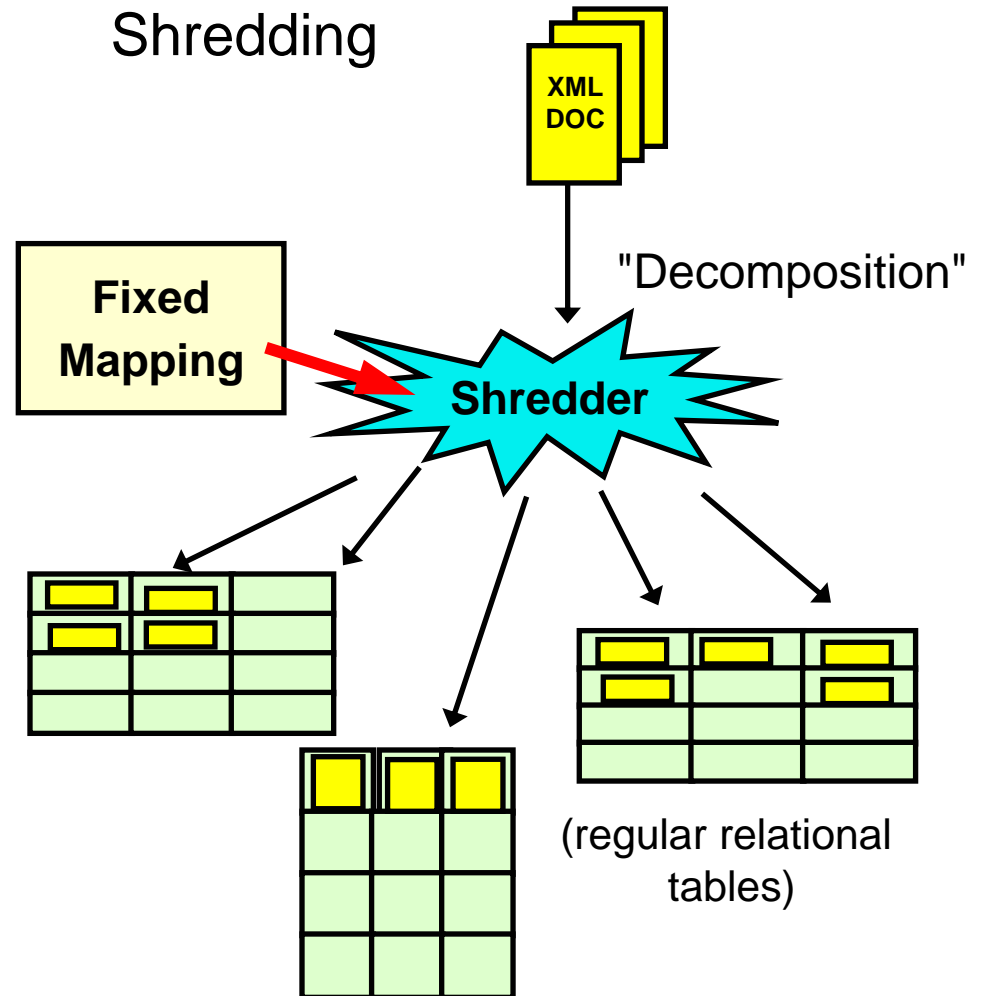
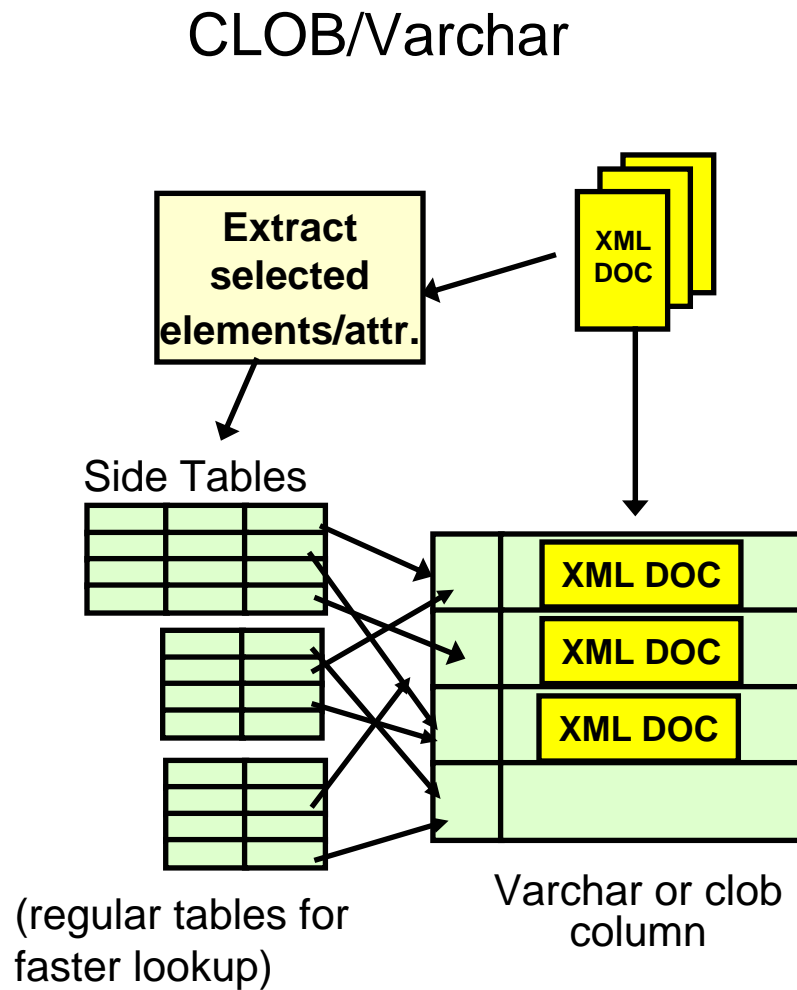
- Managing large volumes of XML data is a DB problem!
 - ▶ Efficient Search & Retrieval of XML
 - ▶ Persistency, Recovery, Transactions, ACID
 - ▶ Performance, Scalability
 - ▶ ...all the same reasons as for relational data!

- Integration
 - ▶ Integrate new XML data with existing relational data
 - ▶ Publish (relational) data as XML
 - ▶ Database support for web applications, SOA, web services (SOAP)

XML Databases

- XML-enabled Databases
 - ▶ The core data model is not XML (but e.g. relational)
 - ▶ Mapping between XML data model and DB's data model is required, or XML is stored as text
 - ▶ E.g.: DB2 XML Extender (V7, V8)
- Native XML Databases
 - ▶ Use the hierarchical XML data model to store and process XML internally
 - ▶ No mapping, no storage as text
 - ▶ Storage format = processing format
 - ▶ E.g.: DB2 9

XML-Enabled Databases: Two Main Options

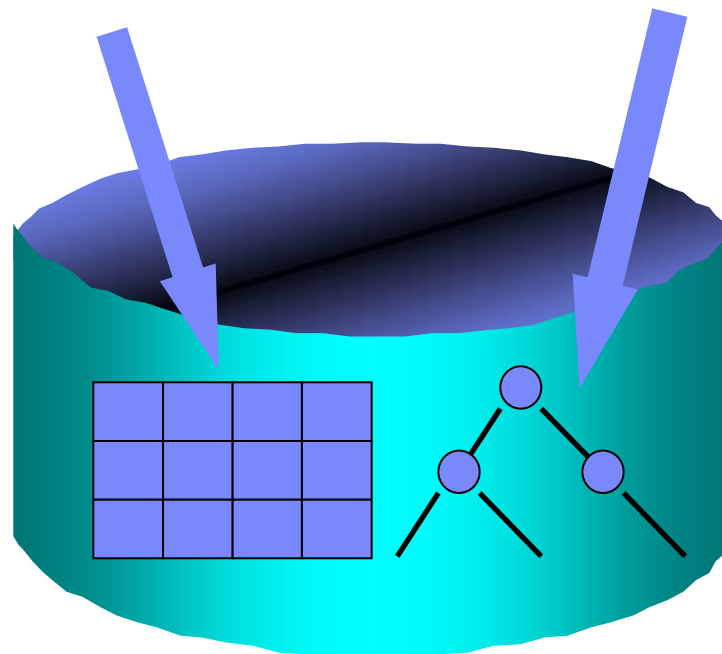


Native XML Storage

- Must store XML in parsed hierarchical format (similar to the DOM representation of the XML infoset)

```
create table dept (deptID char(8), ..., deptdoc xml);
```

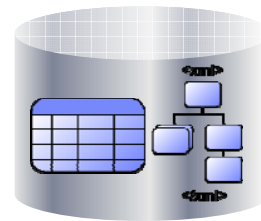
- Relational columns are stored in relational format (tables)
- XML columns are stored natively
- XML stored in UTF8



XML in DB2 9



SQL Person... "I see a world class RDBMS that also supports XML"



DB2 with XML Support



XML Person... "I see a world class XML repository that also supports SQL"

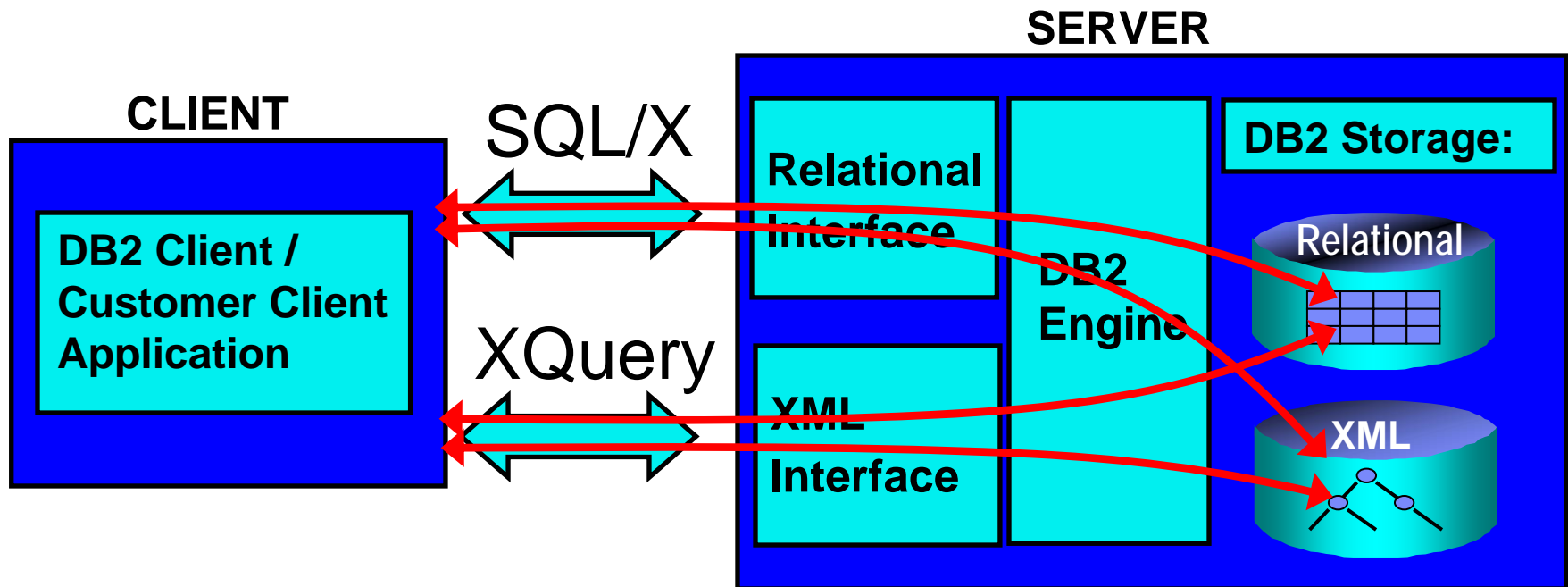
XML integrated in all facets of DB2!

New XML applications benefit from:

- Ability to seamlessly leverage relational investment
- Proven Infrastructure that provides enterprise-class capabilities

XML in DB2

- "Feels" relational and/or XML
- Both SQL flavor and fully XML flavor
- XML *is* DB2 internals - XML Extender becomes one with the data engine



Reduce Code Complexity with DB2 9

```
<?php
$conn = db2_connect($dbname, $dbuser, $dbpass);

/* Insert Customer Documents */

$stmt = db2_prepare($conn, "VALUES (NEXT VALUE FOR
Cid)");
db2_execute($stmt);
list($Cid) = db2_fetch_array($stmt);
```

```
/* Insert Product Documents */

$fileContents = file_get_contents
("products/p1.xml");
$dom = simplexml_load_string($fileContents);

$prodID = (string) $dom["pid"];

$stmt = db2_prepare($conn, "INSERT INTO xmlproduct
(Pid, Description) VALUES (?, ?)");
if(!db2_execute($stmt, array($prodID,
```

```
$Cid) );
db2_execute($stmt);
list($Cid) = db2_fetch_array($stmt);
```

```
/* Insert Product Documents */

$fileContents = file_get_contents
("products/p1.xml");
$dom = simplexml_load_string($fileContents);

$prodID = (string) $dom["pid"];
```


Make Changes Easily with DB2 9

```

<DEPARTMENT deptid="15" deptname="Sales">
  <EMPLOYEE>
    <EMPNO>10</EMPNO>
    <FIRSTNAME>CHRISTINE</FIRSTNAME>
    <LASTNAME>SMITH</LASTNAME>
    <PHONE>408-463-4963</PHONE>
    <PHONE>415-010-1234</PHONE>
    <SALARY>52750.00</SALARY>
  </EMPLOYEE>
  <EMPLOYEE>
    <EMPNO>27</EMPNO>
    <FIRSTNAME>MICHAEL</FIRSTNAME>
    <LASTNAME>THOMPSON</LASTNAME>
    <PHONE>406-463-1234</PHONE>
    <SALARY>41250.00</SALARY>
  </EMPLOYEE>
</DEPARTMENT>

```

Requires:

- Normalization of existing data !
- Modification of the mapping
- Change of applications

Phone

EMPNO	PHONE
27	406-463-1234
10	415-010-1234
10	408-463-4963

Department

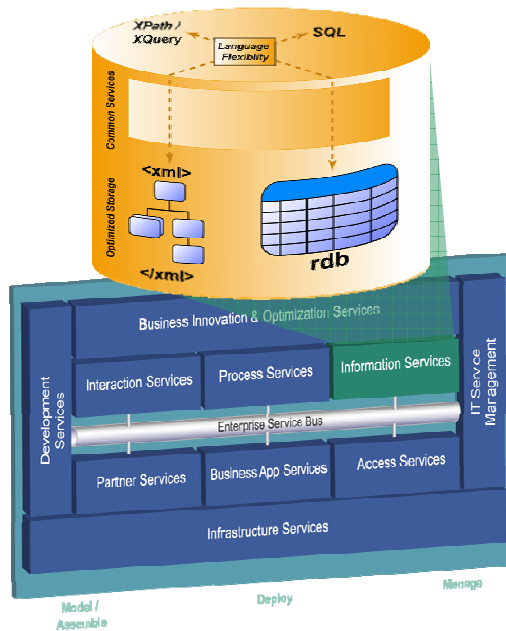
DEPTID	DEPTNAME
15	Sales

Costly!

Employee

DEPTID	EMPNO	FIRSTNAME	LASTNAME	PHONE	SALARY
15	27	MICHAEL	THOMPSON	406-463-1234	41250
15	10	CHRISTINE	SMITH	408-463-4963	52750

DB2 9: Early adopter feedback



Proto-type results using DB2 9 based SOA solution

Task	With relational DB	With DB2 9
Development of search & retrieval business processes	CLOB: 8 hrs Shred: 2 hrs	30 min.
Relative lines of I/O code (65% reduction)	100	35
Add field to schema	1 week	5 min.
Queries	24 - 36 hrs	20 sec - 10 min

What industry experts are saying about DB2 9 XML Technology?

- ***"...this leaves Oracle and Sybase ...well behind the curve, with Microsoft and the others more or less out of sight."***
 - ▶ Philip Howard, Bloor Research, The Register

- ***"You want to be able to take those data-centric things in XML and put them into a database without a loss of fidelity, and this is one area where IBM is going further than Oracle and Microsoft."***
 - ▶ Peter O'Kelley Burton Group, Internet News

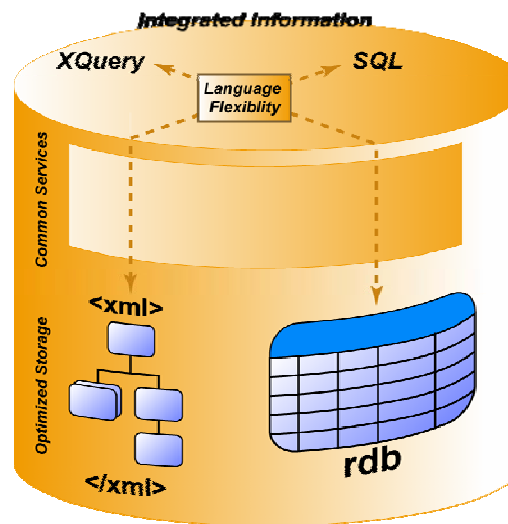
- ***"... enable users to work with both types of data via SQL or Xquery requests. It could also open up the the world of database applications to developers and ISVs with experience in other data types and applications."***
 - ▶ Barbara Darrow, CRN



Selected Partners enabling XML based solutions for DB2 9



JustSystem

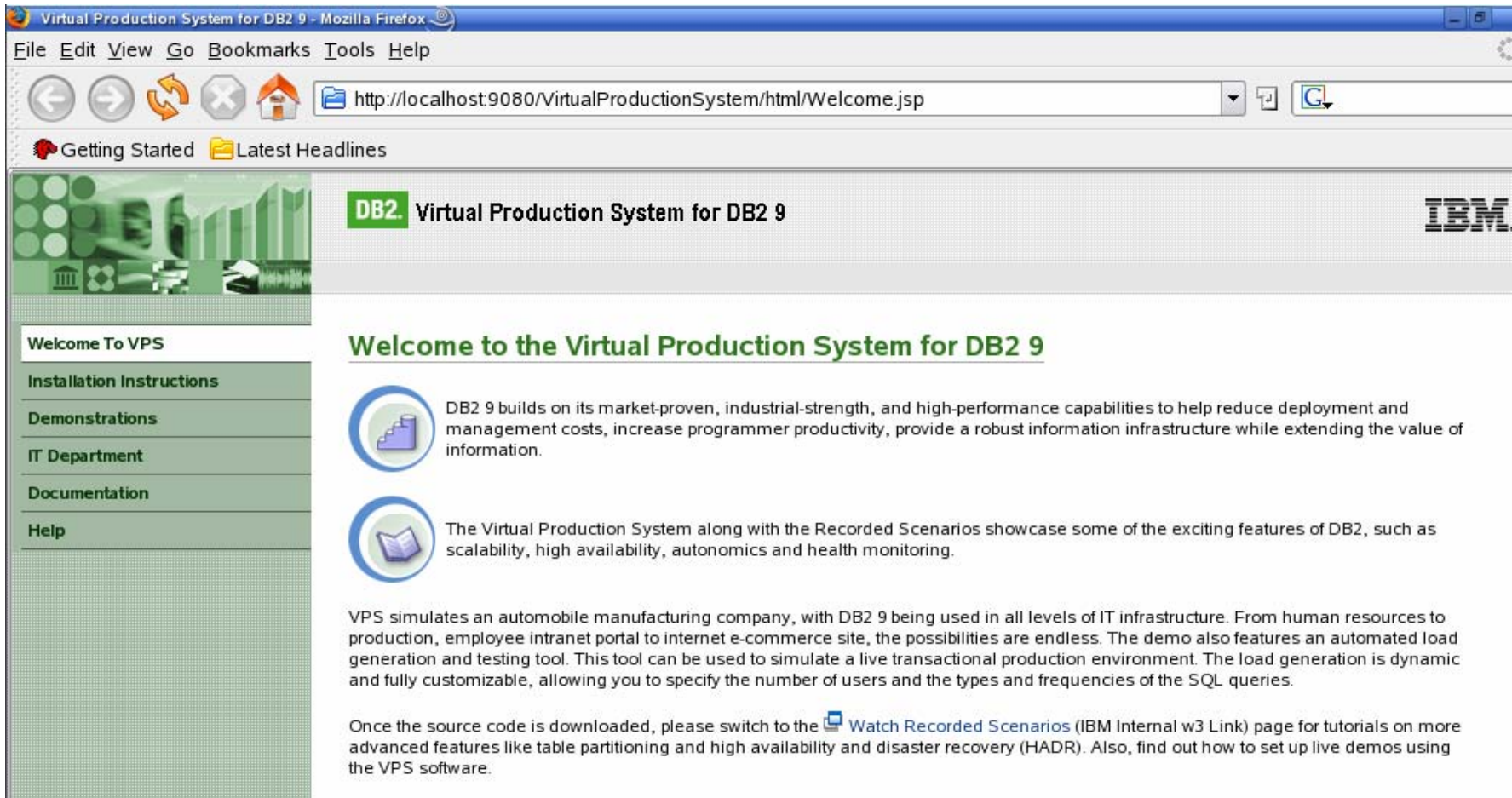


db2vps Demo

- Automobile manufacturing company system
 - ▶ Intranet employee portal
 - ▶ Internet e-commerce
 - ▶ Administration for different departments of the company)
 - ▶ Used for OLTP demos with DB2, and pureXML demos
- Built using:
 - ▶ J2EE, WAS 6.0, DB2 Express-C 9, SLES 10



db2vps Demo



The screenshot shows a web browser window titled "Virtual Production System for DB2 9 - Mozilla Firefox". The address bar displays "http://localhost:9080/VirtualProductionSystem/html/Welcome.jsp". The page features a navigation menu on the left with links: "Welcome To VPS", "Installation Instructions", "Demonstrations", "IT Department", "Documentation", and "Help". The main content area is titled "DB2. Virtual Production System for DB2 9" and includes the IBM logo. Below the title, there is a section titled "Welcome to the Virtual Production System for DB2 9" with two bullet points: "DB2 9 builds on its market-proven, industrial-strength, and high-performance capabilities to help reduce deployment and management costs, increase programmer productivity, provide a robust information infrastructure while extending the value of information." and "The Virtual Production System along with the Recorded Scenarios showcase some of the exciting features of DB2, such as scalability, high availability, autonomies and health monitoring." A paragraph follows, stating: "VPS simulates an automobile manufacturing company, with DB2 9 being used in all levels of IT infrastructure. From human resources to production, employee intranet portal to internet e-commerce site, the possibilities are endless. The demo also features an automated load generation and testing tool. This tool can be used to simulate a live transactional production environment. The load generation is dynamic and fully customizable, allowing you to specify the number of users and the types and frequencies of the SQL queries." Finally, a note at the bottom says: "Once the source code is downloaded, please switch to the [Watch Recorded Scenarios](#) (IBM Internal w3 Link) page for tutorials on more advanced features like table partitioning and high availability and disaster recovery (HADR). Also, find out how to set up live demos using the VPS software."

db2vps Demo

Virtual Production System for DB2 9 - Mozilla Firefox

File Edit View Go Bookmarks Tools Help

http://localhost:9080/VirtualProductionSystem/html/FindCar.jsp

Getting Started Latest Headlines

DB2. Virtual Production System for DB2 9

Welcome To VPS
Installation Instructions
Demonstrations
IT Department
Documentation
Help

AUTO SEARCH

FIND YOUR VEHICLE FIND YOUR PARTS REGISTER


Make: Select a make
Model: Select a model
Year: Any to Any
Price: 0 to >10000
Search

INSURANCE CENTRE
Save your money, get a free insurance quote before you purchase a vehicle.

LATEST VEHICLE REVIEWS
For the best selection and price, shop Auto Search.

SHOPPING ADVICE
Tips to auto shopping. All the information you need to purchase your car.

db2vps Demo



DB2. Virtual Production System for DB2 9

IBM

Welcome To VPS


Installation Instructions

Demonstrations

IT Department




Documentation

Help



 **AUTO SEARCH**

FIND YOUR VEHICLE | FIND YOUR PARTS | REGISTER

SEARCH RESULTS :

DESCRIPTION	YEAR	PRICE
 Minivan Primia Sold as-is.	1952	56389.0
 Minivan Primia Minor engine knocking.	1957	46102.0
 Minivan Primia Major engine knocking.	1961	40157.0


db2vps Demo

**DB2. Virtual Production System for DB2 9**

Welcome To VPS
Installation Instructions
Demonstrations
XML Scenario
OLTP Scenario
IT Department
Documentation
Help

Application Configuration

To begin using VPS please enter the connection information of the VPS database that you have created. If you have not yet created the database, information on how to do so can be found by clicking the [Installation Instructions](#) tab in the left menu.

Server: Status:  Connected
Port:
Instance:
DBName:
User:
Pass:
Driver:

XML Scenario

DB2 9 pureXML provides native XML storage and retrieval. This demo shows performance benefits of using the native XML support compared to storing and accessing XML data in a LOB data type. The easy to use interface provides you with the ability to configure and run the demo with a variable set of users and transaction types.

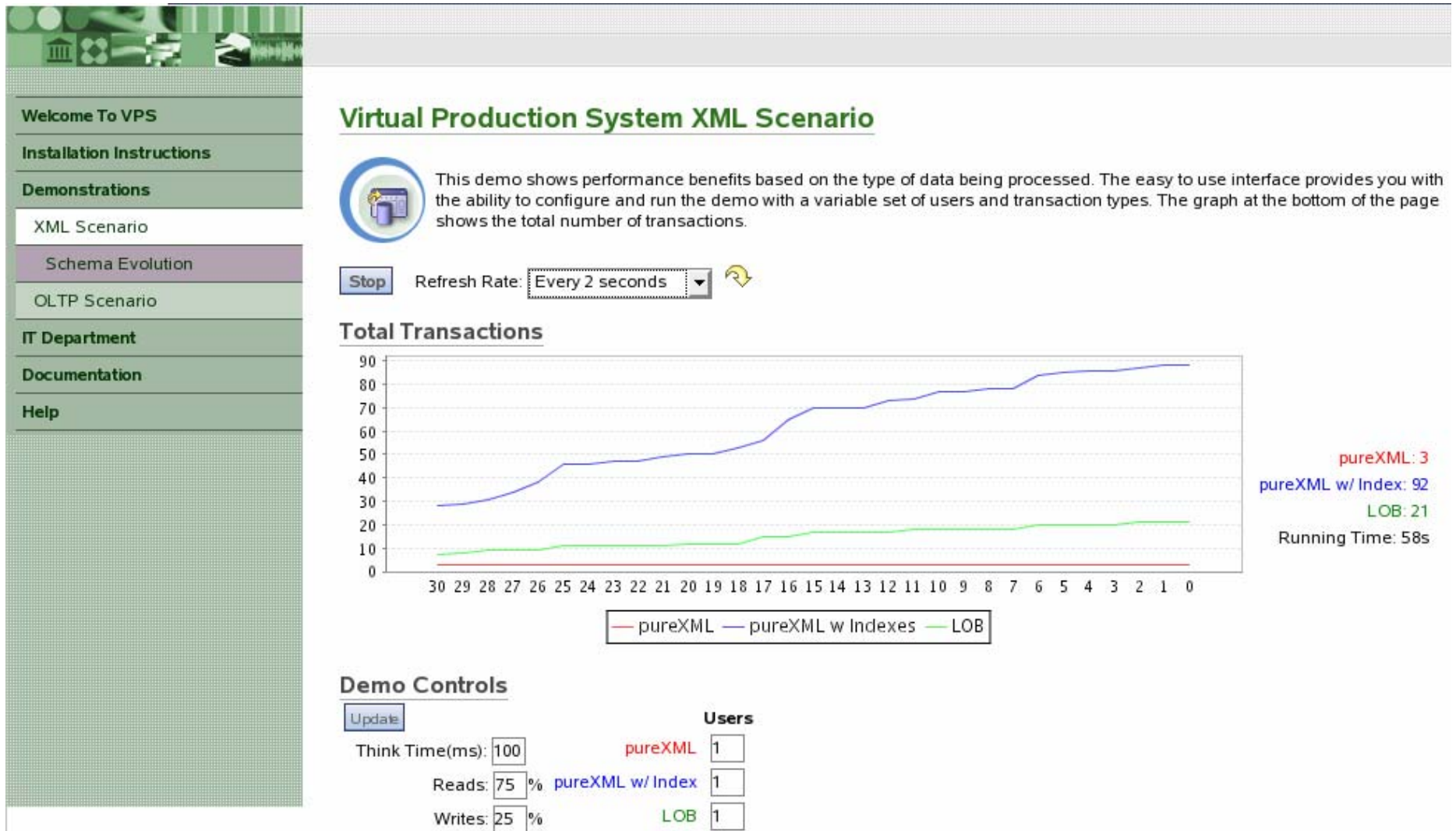
[Click here to start the XML Scenario](#)

OLTP Scenario


This scenario simulates a production level OLTP environment. The easy to use interface provides you with the ability to configure and run the demo with a variable set of users and transaction weightings. This environment can be used to conduct database efficiency tests and to showcase the health monitoring facilities.

[Click here to start the OLTP Scenario](#)

db2vps Demo



db2vps Demo



DB2 Virtual Production System for DB2 9

Welcome To VPS

Installation Instructions

Demonstrations

- XML Scenario
- OLTP Scenario
- Average Transaction Time
- Locking Scenario
- STMM Scenario


IT Department

Documentation

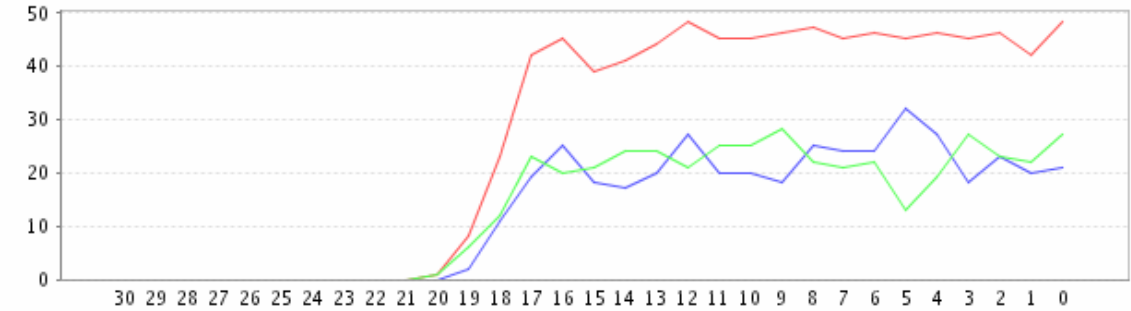
Help

Virtual Production System OLTP Scenario

This demo simulates a production level OLTP environment. The easy to use interface provides you with the ability to configure and run the demo with a variable set of users and transaction weightings. The graph at the bottom of the page shows the transaction throughput.

Refresh Rate: 

Transactions/Second



— Total Transactions — Read Transactions — Write Transactions

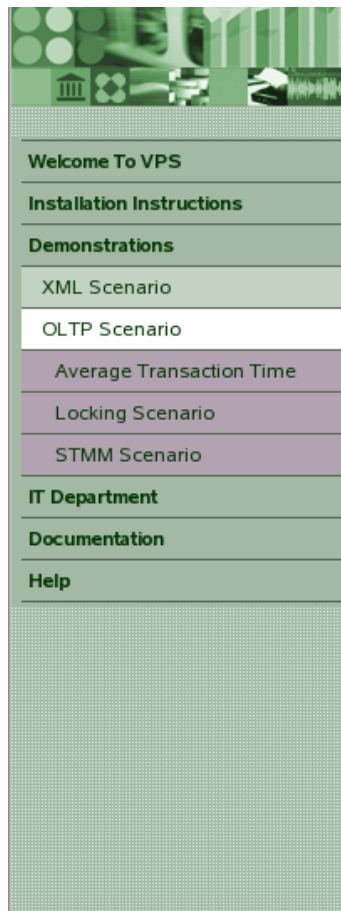
Total Transactions: 840
Reads: 412
Writes: 428
Running Time: 22s

Demo Controls

Workload:

Users: Think Time (ms):

db2vps Demo



DB2. Virtual Product

Virtual Product

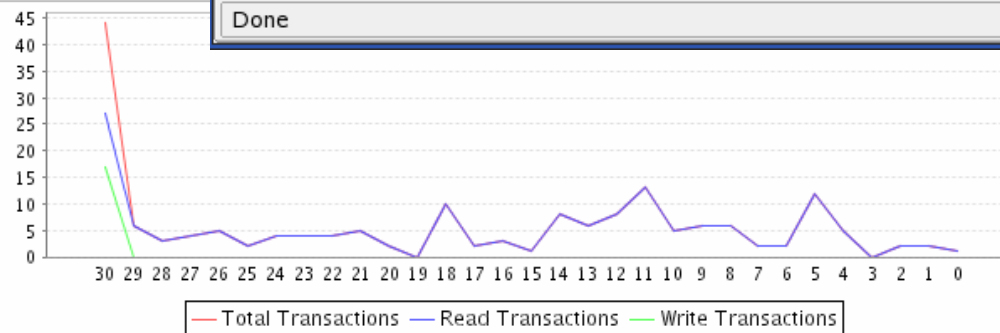


This demo
and run the
transaction

Stop

Refresh Rate:

Transactions/Sec



Demo Controls

Update

Workload: Mixed

Users: 5

Think Time (ms): 100

File Edit View Go Bookmarks Tools Help

http://localhost:9080/VirtualProduct

Getting Started Latest Headlines

Parameter	Current Value	Target Value	Status	Value	Action
lock list	512	100	✓	100	Obtain Locks
max locks	98 %	98 %	✓	98	
package cache size	1200	1200	✓	1200	Tune Cache
sort heap threshold	5000	5000	✓	500	Execute Sorts
sort heap	257	256	✓	256	
buffer pool size	1000	2	✗	2	Update Values

Reset Baseline

In order to force the DB2 Self Tuning Memory Management feature to update memory assignments in any of the listed parameters, we must set the current threshold in that field.

Done

Total Transactions:
 10510
 Reads: 5287
 Writes: 5223
 Running Time: 261s



IBM Software Group

Inserts of XML Data

DB2 Information Management Software

A horizontal bar containing a series of small, colorful icons including a green square, yellow square, red square, purple square, cyan square, a camera lens, a green cross, a blue sphere, a green arrow, a grey grid, a green cube, a green square, a blue and green striped rectangle, a green square, a green square, a green square, and a green square.

ON DEMAND BUSINESS™



© 2005 IBM Corporation













Table definitions with XML columns

```
create table items (  
  id          int primary key not null,  
  brandname   varchar(30),  
  itemname    varchar(30),  
  sku         int,  
  srp         decimal(7,2),  
  comments    xml  
);
```

```
create table clients(  
  id          int primary key not null,  
  name        varchar(50),  
  status      varchar(10),  
  contact     xml  
);
```

Contents of D:\Raul directory with XML documents

D:\Raul   G

Name ▲	Size	Type
 Client3227.xml	1 KB	XML Document
 Client4309.xml	1 KB	XML Document
 Client5681.xml	1 KB	XML Document
 Client8877.xml	1 KB	XML Document
 Client9077.xml	1 KB	XML Document
 Client9177.xml	1 KB	XML Document
 ClientInfo.xsd	2 KB	XML Schema
 clients.del	1 KB	DEL File
 Comment3926.xml	1 KB	XML Document
 Comment4023.xml	1 KB	XML Document
 Comment4272.xml	1 KB	XML Document
 items.del	1 KB	DEL File

Insert & Import of XML data

```
INSERT INTO clients VALUES (77, 'John Smith', 'Gold',  
    '<addr>111 Main St., Dallas, TX, 00112</addr>');
```

```
IMPORT FROM "D:\Raul\clients.del" of del xml from "D:\Raul"  
    INSERT INTO ARFCHONG.CLIENTS (ID, NAME, STATUS,  
    CONTACT);
```

```
IMPORT FROM "D:\Raul\items.del" of del xml from "D:\Raul"  
    INSERT INTO ARFCHONG.ITEMS (ID, BRANDNAME,  
    ITEMNAME, SKU, SRP, COMMENTS);
```




IBM Software Group

XPath

DB2 Information Management Software

A horizontal bar containing a series of small, colorful icons representing various business and technology concepts, such as a globe, a network, a bar chart, and a document.

ON DEMAND BUSINESS™

© 2005 IBM Corporation

XPath

- ❑ XML Query Language
- ❑ Subset of XQuery & SQL/XML

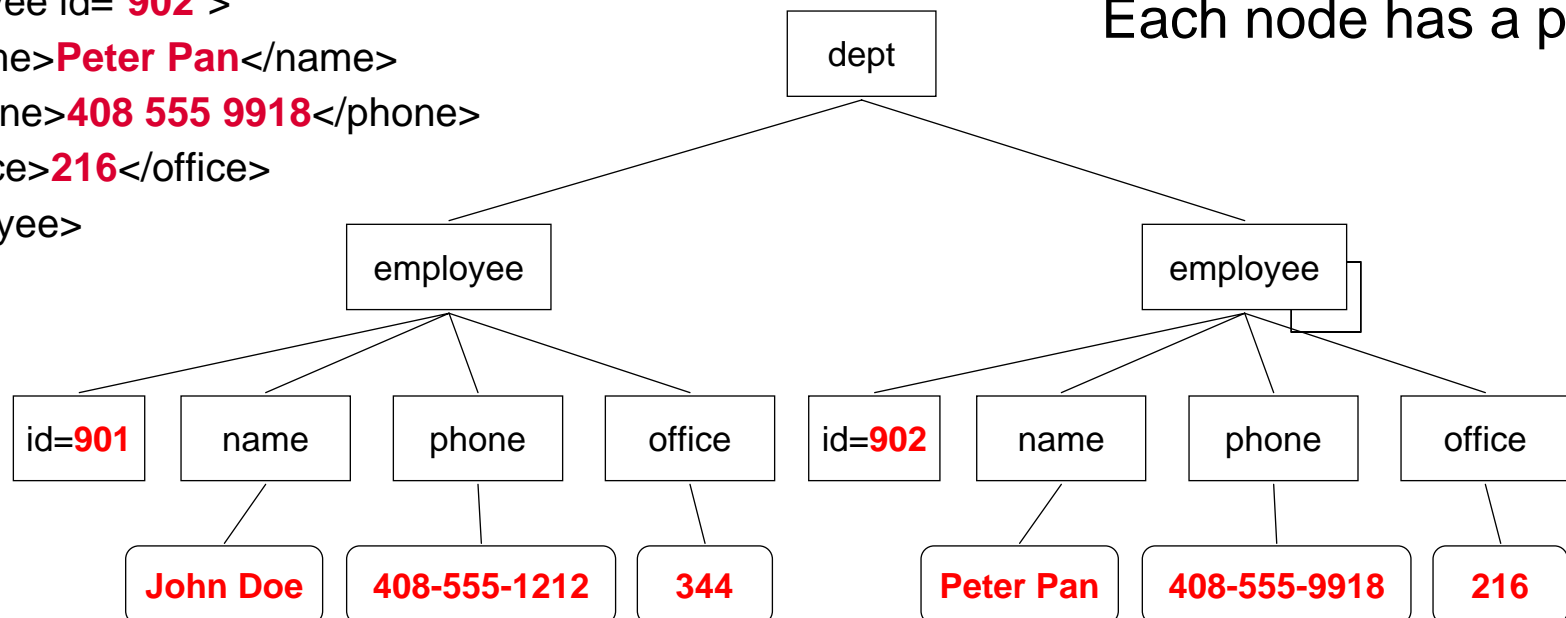
```

<dept bldg="101">
  <employee id="901">
    <name>John Doe</name>
    <phone>408 555 1212</phone>
    <office>344</office>
  </employee>
  <employee id="902">
    <name>Peter Pan</name>
    <phone>408 555 9918</phone>
    <office>216</office>
  </employee>
</dept>

```

/
/dept
/dept/employee
/dept/employee/@id
/dept/employee/name
/dept/employee/phone
/dept/employee/office
(...)

Each node has a path



XPath: Simple XPath Expressions

- Use fully qualified paths to specify elements/attributes
- “@” is used to specify an attribute
- use “text()” to specify the text node under an element

```
<dept bldg="101">
  <employee id="901">
    <name>John Doe</name>
    <phone>408 555 1212</phone>
    <office>344</office>
  </employee>
  <employee id="902">
    <name>Peter Pan</name>
    <phone>408 555 9918</phone>
    <office>216</office>
  </employee>
</dept>
```

XPath	Result
/dept/@bldg	101
/dept/employee/@id	901 902
/dept/employee/name	<name>Peter Pan</name> <name>John Doe</name>
/dept/employee/name/text()	Peter Pan John Doe

XPath: Wildcards

- * matches any tag name
- // is the “descendent-or-self” wildcard

```
<dept bldg="101">
  <employee id="901">
    <name>John Doe</name>
    <phone>408 555 1212</phone>
    <office>344</office>
  </employee>
  <employee id="902">
    <name>Peter Pan</name>
    <phone>408 555 9918</phone>
    <office>216</office>
  </employee>
</dept>
```

XPath	Result
/dept/employee/*/text()	John Doe 408 555 1212 344 Peter Pan 408 555 9918 216
/dept/*/@id	901 902
//name/text()	Peter Pan John Doe
/dept//phone	<phone>408 555 1212</phone> <phone>408 555 9918</phone>

XPath: Predicates

- Predicates are enclosed in square brackets [...]
- Can have multiple predicates in one Xpath
- Positional predicates: [n] selects the n-th child

```
<dept bldg="101">
  <employee id="901">
    <name>John Doe</name>
    <phone>408 555 1212</phone>
    <office>344</office>
  </employee>
  <employee id="902">
    <name>Peter Pan</name>
    <phone>408 555 9918</phone>
    <office>216</office>
  </employee>
</dept>
```

XPath	Result
/dept/employee[@id="902"]/name	<name>Peter Pan</name>
/dept[@bldg="101"]/employee[office > "300"]/name	<name>John Doe</name>
//employee[office="344" OR office="216"]/@id	901 902
/dept/employee[2]/@id	902

XPath: The Parent Axis

- Current context: “.”
- Parent context: “..”

```
<dept bldg="101">
  <employee id="901">
    <name>John Doe</name>
    <phone>408 555 1212</phone>
    <office>344</office>
  </employee>
  <employee id="902">
    <name>Peter Pan</name>
    <phone>408 555 9918</phone>
    <office>216</office>
  </employee>
</dept>
```

XPath	Result
/dept/employee/name[../@id="902"]	<name>Peter Pan</name>
/dept/employee/office[.>"300"]	<office>344</office>
/dept/employee[office > "300"]/office	<office>344</office>
/dept/employee[name="John Doe"]/../@bldg	101
/dept/employee/name[.="John Doe"]/../../@bldg	101



What is XQuery?

XML Schema

[www.w3.org/
XML/Schema](http://www.w3.org/XML/Schema)

XQuery

Expressions

www.w3.org/TR/XQuery & [SQL/XML](http://www.w3.org/TR/SQL/XML)

Functions & Operators

www.w3.org/TR/XQuery-operators/

XPath 2.0

[www.w3.org/
TR/xpath20/](http://www.w3.org/TR/xpath20/)

XQuery 1.0 & XPath 2.0 Data Model

www.w3.org/TR/query-datamodel/

FLWOR and PATH expressions

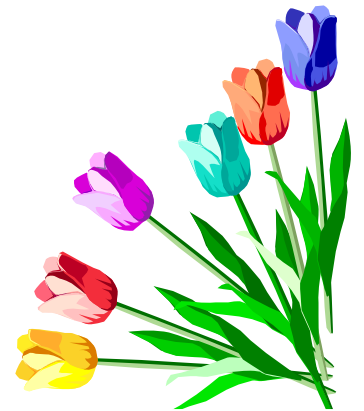
- Like SELECT-FROM-WHERE expression in SQL an XQuery FLWOR expression may contain several clauses that begin with certain keywords.
 - for
 - let
 - where
 - order
 - Return
- A Path expression in XQuery consists of a series of “steps” separated by slash characters

XQuery: The FLWOR Expression

- **FOR**: iterates through a sequence, bind variable to items
- **LET**: binds a variable to a sequence
- **WHERE**: eliminates items of the iteration
- **ORDER**: reorders items of the iteration
- **RETURN**: constructs query results

```
create table dept(deptID char(8),deptdoc xml);
```

```
xquery  
for $d in db2-fn:xmlcolumn('dept.deptdoc')/dept  
let $emp := $d//employee/name  
where $d/@bldg > 95  
order by $d/@bldg  
return  
  <EmpList>  
    {$d/@bldg, $emp}  
  </EmpList>
```



```
<dept bldg=101>  
  <employee id=901>  
    <name>John Doe</name>  
    <phone>408 555 1212</phone>  
    <office>344</office>  
  </employee>  
  <employee id=902>  
    <name>Peter Pan</name>  
    <phone>408 555 9918</phone>  
    <office>216</office>  
  </employee>  
</dept>
```



IBM Software Group

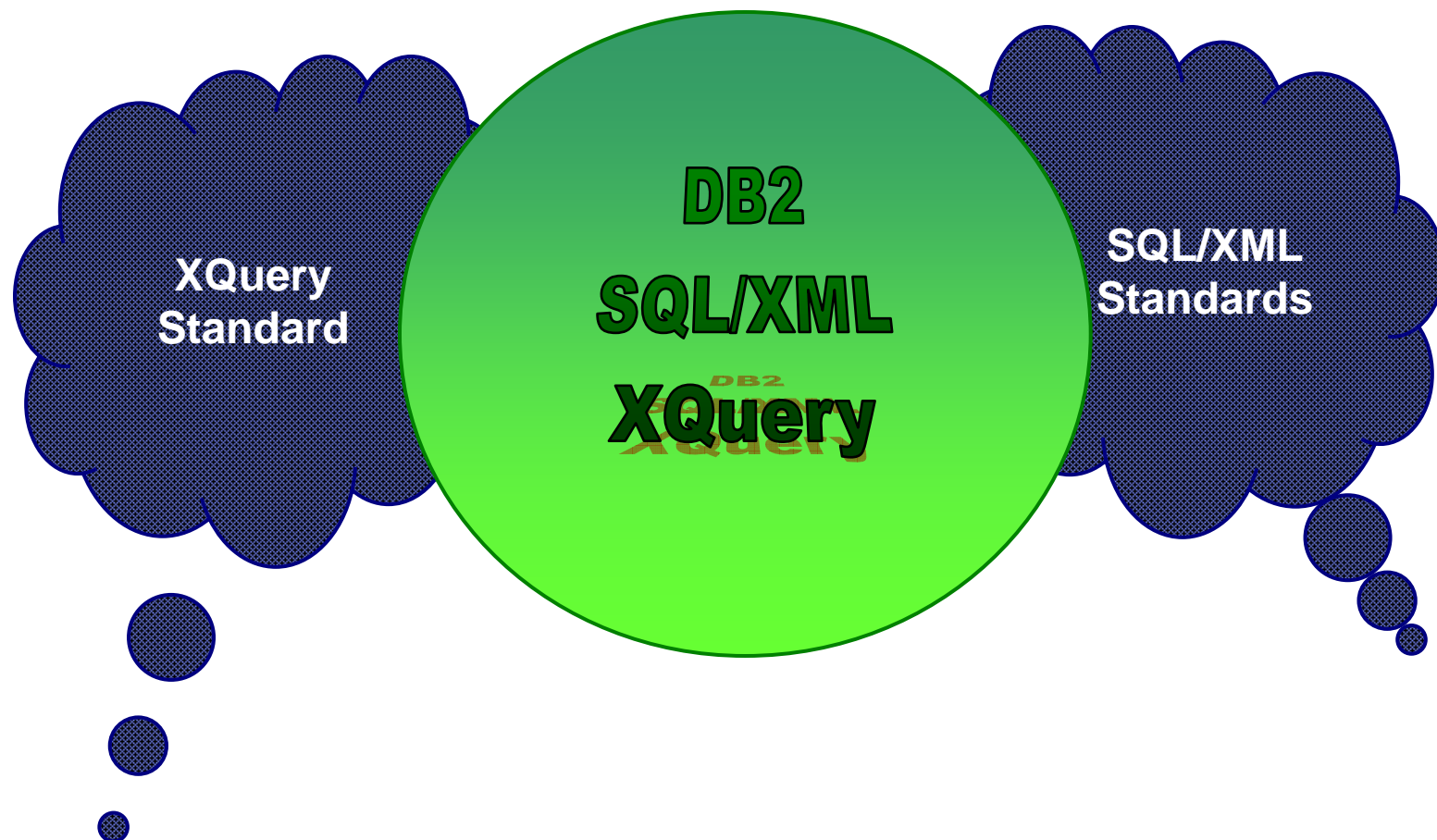
Query DB2 XML Data with SQL

DB2 Information Management Software



© 2005 IBM Corporation

Two Worlds



What you can and can't do with SQL

- Plain SQL statements enable you to retrieve full XML documents
 - ▶ But you can't specify XML-based query predicates and you can't retrieve partial XML documents or specific element values from an XML document
 - ▶ You can't join, aggregate or order by fragments of XML documents using plain SQL
- You need to use SQL with XML extensions, (SQL/XML), XQUERY, or a combination of both



SQL/XML queries

- SQL/XML is designed to bridge between the SQL and XML worlds.
 - ▶ Part of the SQL standard includes specifications for embedding XQuery or XPath expressions within SQL statements
- XPATH is a language for navigating XML documents to find elements or attributes
- XQuery includes support for XPATH



New SQL/XML Functions in SQL 2006

- **XMLPARSE**
 - ▶ parses character/BLOB data, produces XML value
- **XMLSERIALIZE**
 - ▶ converts an XML value into character/BLOB data
- **XMLVALIDATE**
 - ▶ validates XML value against an XML schema and type-annotates the XML value
- **XMLEXISTS**
 - ▶ determines if an XQuery returns a results (i.e. a sequence of one or more items)
- **XMLQUERY**
 - ▶ executes an XQuery and returns the result sequence
- **XMLTABLE**
 - ▶ executes an XQuery, returns the result sequence as a relational table (if possible)
- **XMLCAST**
 - ▶ cast to or from an XML type

Restricting results based on XML element values

- SQL programmers often write queries that restrict the rows returned from a table.

- ▶ Example:

- SELECT lastname FROM db2admin.clients WHERE age > 45**

- But what if you want to restrict your search based on some condition that applies to data in a XML column?



XMLExists function

- Enables you to navigate to an element within your XML document and test for a specific condition.
- When specified as part of a WHERE clause, XMLExists restricts the returned results to only those rows that contain an XML document with the specific XML element value.

xmlexists function

- Restricts results based on an XML element value

```
select name from clients  
where xmlexists('$c/Client/Address[zip="95116"]'  
passing clients.contact as "c")
```

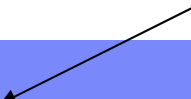
Demo 1



Restricting results based on an XML element value

Retrieves information in the “name” column of the “ clients “ table.


```
select name from clients  
where xmlexists('$c/Client/Address[zip="95116"]'  
passing clients.contact as "c")
```



Restricting results based on an XML element value

The WHERE clause invokes the XMLEExists function, specifying the XPath expression that prompts DB2 to navigate to the “zip” element and check for a value of 95116

select name from clients
where xmlexists('\$c/Client/Address[zip="95116"]'
passing clients.contact as "c")



Restricting results based on an XML element value

```
select name from clients
where xmlexists('$c/Client/Address[zip="95116"]'
passing clients.contact as "c")
```

**The “\$c/Client/Address” clause indicates the path
in the XML document hierarchy
where DB2 can locate the “Zip” element**

Demo 1



xmlquery function

- Retrieve one or more element values from our XML document

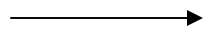
```
select xmlquery('$c/Client/email'  
passing contact as "c")  
from clients  
where status = 'Gold'
```

Demo 2



XMLQuery function

The first line specifies return values
For the “email” sub-element of the root
“Client” element.



```
select xmlquery('$c/Client/email'  
passing contact as "c")  
from clients  
where status = 'Gold'
```

XMLQuery function

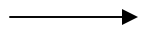
The second and third lines indicate where DB2 can find this information

→ select xmlquery('\$c/Client/email'
→ passing contact as "c")
from clients
where status = 'Gold'

XMLQuery function

The fourth line qualifies your query to indicate
your only interested
in email address of Gold Customers

select xmlquery('\$c/Client/email'
passing contact as "c")
from clients
where status = 'Gold'



Demo 2



XMLTABLE Function: From XML to Relational

- Retrieving multiple XML element values
- Generates tabular output from data stored in XML columns (Relational view of XML data)
- Also includes clauses to map to XML data into result set columns of SQL data types

xmltable function: From XML to Relational

```
select t.comment#,i.itemname,t.customerID,Message  
from items i,  
xmltable('$c/Comments/Comment' passing i.comments as "c"  
         columns Comment# integer path 'CommentID',  
                  CustomerID integer path 'CustomerID',  
                  Message varchar(100) path 'Message') as t
```

Demo 3

Columns included in your result set
Columns surrounded by “ marks and prefixed with
“t” variable are XML
Element values

→ select t.comment#,i.itemname,t.customerID,Message
from items i,
xmltable('\$c/Comments/Comment' passing i.comments as "c"
columns Comment# integer path 'CommentID',
CustomerID integer path 'CustomerID',
Message varchar(100) path 'Message') as t

Third line invokes XMLTable function
to specify the DB2 XML
column containing the target

```
select t.comment#,i.itemname,t.customerID,Message  
from items i,
```

```
→ xmltable('$c/Comments/Comment' passing i.comments as "c"  
        columns Comment# integer path 'CommentID',  
                CustomerID integer path 'CustomerID',  
                Message varchar(100) path 'Message') as t
```

The “columns” clause, spanning lines 4 to 6, identifies the specific XML elements that will be mapped to output columns in the SQL result set., specified on line 1

```
select t.comment#,i.itemname,t.customerID,Message  
from items i,  
xmltable('$c/Comments/Comment' passing i.comments as "c"  
→      columns Comment# integer path 'CommentID',  
→           CustomerID integer path 'CustomerID',  
→           Message varchar(100) path 'Message') as t
```

Demo 3

Retrieving XML data using “for” and “return” clauses of XQuery

```
Select name, xmlquery('for $e in $c/Client/email[1] return $e'  
Passing contact as “c”)  
From clients  
Where status = 'Gold'
```



The first line specifies that the customer names and output from the XMLQuery function will be included in the result set.

→ Select name, xmlquery('for \$e in \$c/Client/email[1] return \$e'
Passing contact as "c")
From clients
Where status = 'Gold'

The second line indicates the the first “email” sub-element of the “Client” element is to be returned

Select name, xmlquery('for \$e in \$c/Client/email[1] return \$e'

→ Passing contact as “c”)

From clients

Where status = 'Gold'



The third line identifies the source of
our XML data, contact column)

```
Select name, xmlquery('for $e in $c/Client/email[1] return $e'  
Passing contact as "c")
```

→ From clients
Where status = 'Gold'



Line 4 tells us that this column
is in the “ clients” table.

Select name, xmlquery('for \$e in \$c/Client/email[1] return \$e'
Passing contact as “c”)
From clients
→ Where status = 'Gold'

Demo 4



Retrieving and transforming XML into HTML

```
select xmlquery('for $e in $c/Client/email[1]/text()  
return <p>{$e}</p>'  
Passing contact as "c")  
From clients  
Where status = 'Gold'
```



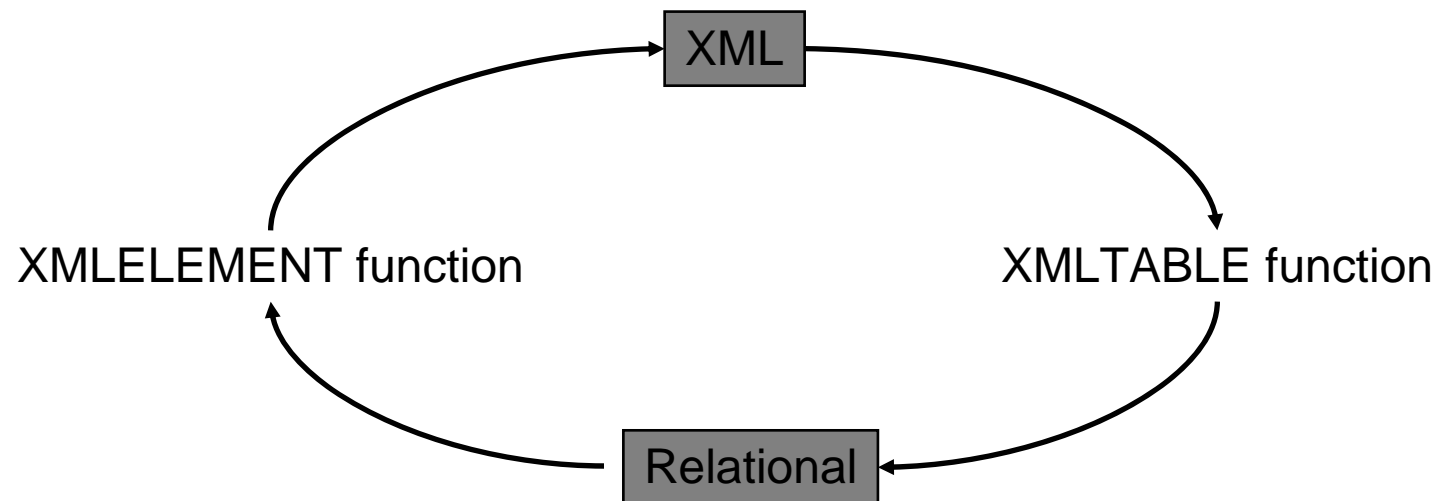
xmlelement function: From Relational to XML

```
select
  xmlelement (name "item",itemname),
  xmlelement (name "id", id),
  xmlelement (name "brand",brandname),
  xmlelement (name "sku",sku)
from items
where srp < 100
```

Sample Output

```
<item>
  <id>4272</id>
  <brand>Classy</brand>
  <sku>981140</sku>
</item>
...
<item>
  <id>1193</id>
  <brand>Natural</brand>
  <sku>557813</sku>
</item>
```

Review: XMLTABLE vs XMLELEMENT





IBM Software Group

Query DB2 XML Data with XQuery

DB2 Information Management Software

A horizontal bar containing a series of small, colorful icons representing various business and technology concepts, including a bar chart, a globe, a network, a document, and a person.

ON DEMAND BUSINESS™

© 2005 IBM Corporation

XQuery

- XQuery supports path expressions to navigate XML hierarchical structure
- XQuery supports both typed and untyped data
- XQuery lacks null values because XML documents omit missing or unknown data
- XQuery returns sequences of XML data



Simple XQuery to return customer contact data

```
Xquery db2-fn:xmlcolumn('CLIENTS.CONTACT')
```

- **db2-fn:xmlcolumn** is a function with a parameter that identifies the table name and column name of an XML column.

Demo 5



FLWOR expression to retrieve client fax data

```
xquery  
for $y in db2-fn:xmlcolumn('CLIENTS.CONTACT')/Client/fax  
return $y
```

```
<fax>4081112222</fax>  
<fax>5559998888</fax>
```

Demo 6

FLWOR expression to retrieve complex XML type

```
xquery  
for $y in db2-fn:xmlcolumn('CLIENTS.CONTACT')/Client/Address  
return $y
```

Demo 7

Path expression with additional filtering predicate

xquery

```
Db2-fn:xmlcolumn('CLIENTS.CONTACT')/Client/Address[zip="95116"]
```

Demo 8



Querying DB2 Xml data and returning results as HTML

```
xquery
<ul> {
for $y in db2-fn:xmlcolumn('CLIENTS.CONTACT')/Client/Address
order by $y/zip
return <li>{$y}</li>
}</ul>
```

Demo 9



Sample HTML

```
<ul>
<li>
<address>
    <street>9407 Los Gatos Blvd.</street>
    <city>Los Gatos</city>
    <state>ca</state>
    <zip>95302</zip>
</address>
</li>
<Address>
<street>4209 El Camino Real</street>
    <city>Mountain View</city>
    <state>CA</state>
    <zip>95302</zip>
</address>
</li>
...
</ul>
```

XQueries with embedded SQL

- db2-fn:sqlquery
 - A function which executes an SQL query and returns only the selected data
 - The SQL Query passed to db2-fn:sqlquery must return XML data
 - This XML data can then be further processed by XQuery



Embedded SQL within XQuery

```
xquery
for $y in
db2-fn:sqlquery('select comments from items where srp > 100')/Comments/Comment
where $y/ResponseRequested='Yes'
return (
  <action>
    {$y/ProductID
      $y/CustomerID
      $y/Message}
  </action>
)
```

Demo 10

Joins with SQL/XML

```
create table dept (unitID char(8), deptdoc xml)
```

```
create table unit (unitID char(8) primary key not null,  
                  name char(20),  
                  manager varchar(20),  
                  ...  
)
```

```
select u.unitID  
from dept d, unit u  
where XMLEXISTS ('$e//employee[name = $m]'  
                passing d.deptdoc as "e", u.manager as "m")
```

```
select u.unitID  
from dept d, unit u  
where u.manager = XMLCAST(XMLQUERY('$e//employee/name '  
                                passing d.deptdoc as "e") as char(20))
```

Joins with SQL/XML & Xquery

```
create table dept (unitID char(8), deptdoc xml)
```

```
create table project(projectDoc xml)
```

With SQL/XML:

```
select XMLQUERY ('$d/dept/employee' passing d.deptdoc as "d")  
from dept d, project p  
where XMLEXISTS ('$e/dept[@deptID=$p/project/deptID] '  
                passing d.deptdoc as "e", p.projectDoc as "p")
```

Equivalent with Xquery:

```
XQUERY  
for $dept in db2-fn:xmlcolumn("DEPT.DEPTDOC")/dept  
  for $proj in db2-fn:xmlcolumn("PROJECT.PROJECTDOC")/project  
where $dept/@deptID = $proj/deptID  
return $dept/employee
```



IBM Software Group

Update and Delete XML Data operations

DB2 Information Management Software



© 2005 IBM Corporation

Update and Delete Operations

- Update and Delete XML Data in one of two ways
 - ▶ SQL UPDATE and DELETE Statements
 - ▶ Stored Procedure
 - DB2XMLFUNCTIONS.XMLUPDATE
- Occur at the Document level



Update Example

```
update clients set contact=(  
  xmlparse(document  
    '<Client>  
    <Address>  
      <street>5401 Julio ave.</street>  
      <city>San Jose</city>  
      <state>CA</state>  
      <zip>95116</zip>  
    </Address>  
    <phone>  
      <work>4084633000</work>  
      <home>4081111111</home>  
      <cell>4082222222</cell>  
    </phone>  
      <fax>4087776666</fax>  
      <email>newemail@someplace.com</email>  
    </Client>'))  
where id = 3227
```



IBM Software Group

XML Indexes

DB2 Information Management Software

ON DEMAND BUSINESS™

© 2005 IBM Corporation

XML Indexing Examples

```
create table customer( info XML);
```

```
create unique index idx1 on customer(info)
```

```
generate key using
```

```
xmlpattern '/customerinfo/@Cid'
```

```
as sql double;
```

```
create index idx2 on customer(info)
```

```
generate key using
```

```
xmlpattern '/customerinfo/name'
```

```
as sql varchar(40);
```

```
create index idx3 on customer
```

```
generate key using
```

```
xmlpattern '//name'
```

```
as sql varchar(40);
```

```
<customerinfo Cid="1004">  
  <name>Matt Foreman</name>  
  <addr country="Canada">  
    <street>1596 Baseline</street>  
    <city>Toronto</city>  
    <state>Ontario</state>  
    <pcode>M3Z-5H9</pcode>  
  </addr>  
  <phone type="work">905-555-4789</phone>  
  <phone type="home">416-555-3376</phone>  
  <assistant>  
    <name>Peter Smith</name>  
    <phone type="home">416-555-3426</phone>  
  </assistant>  
</customerinfo>
```

The diagram illustrates the mapping between SQL XML indexing patterns and specific XML elements. Arrows point from the following SQL patterns to their corresponding XML elements:

- `/customerinfo/@Cid` points to the `Cid` attribute of the `customerinfo` root element.
- `/customerinfo/name` points to the `name` element under the `customerinfo` root.
- `//name` points to the `name` element under the `assistant` element.

XML Indexing Examples

```
create unique index idx1 on customer(info)
generate key using
xmlpattern '/customerinfo/@Cid'
as sql double;
```

```
create index idx2 on customer(info)
generate key using
xmlpattern '/customerinfo/name'
as sql varchar(40);
```

```
create index idx3 on customer(info)
generate key using
xmlpattern '//name'
as sql varchar(40);
```

```
create index idx4 on customer info
generate key using
xmlpattern '//text()'
as sql varchar(40);
```

create table **customer**(**info XML**);



Don't index everything!
Too expensive for
insert, update, delete !



```
<customerinfo Cid="1004">
  <name>Matt Foreman</name>
  <addr country="Canada">
    <street>1596 Baseline</street>
    <city>Toronto</city>
    <state>Ontario</state>
    <pcode>M3Z-5H9</pcode>
  </addr>
  <phone type="work">905-555-4789</phone>
  <phone type="home">416-555-3376</phone>
  <assistant>
    <name>Peter Smith</name>
    <phone type="home">416-555-3426</phone>
  </assistant>
</customerinfo>
```


Java



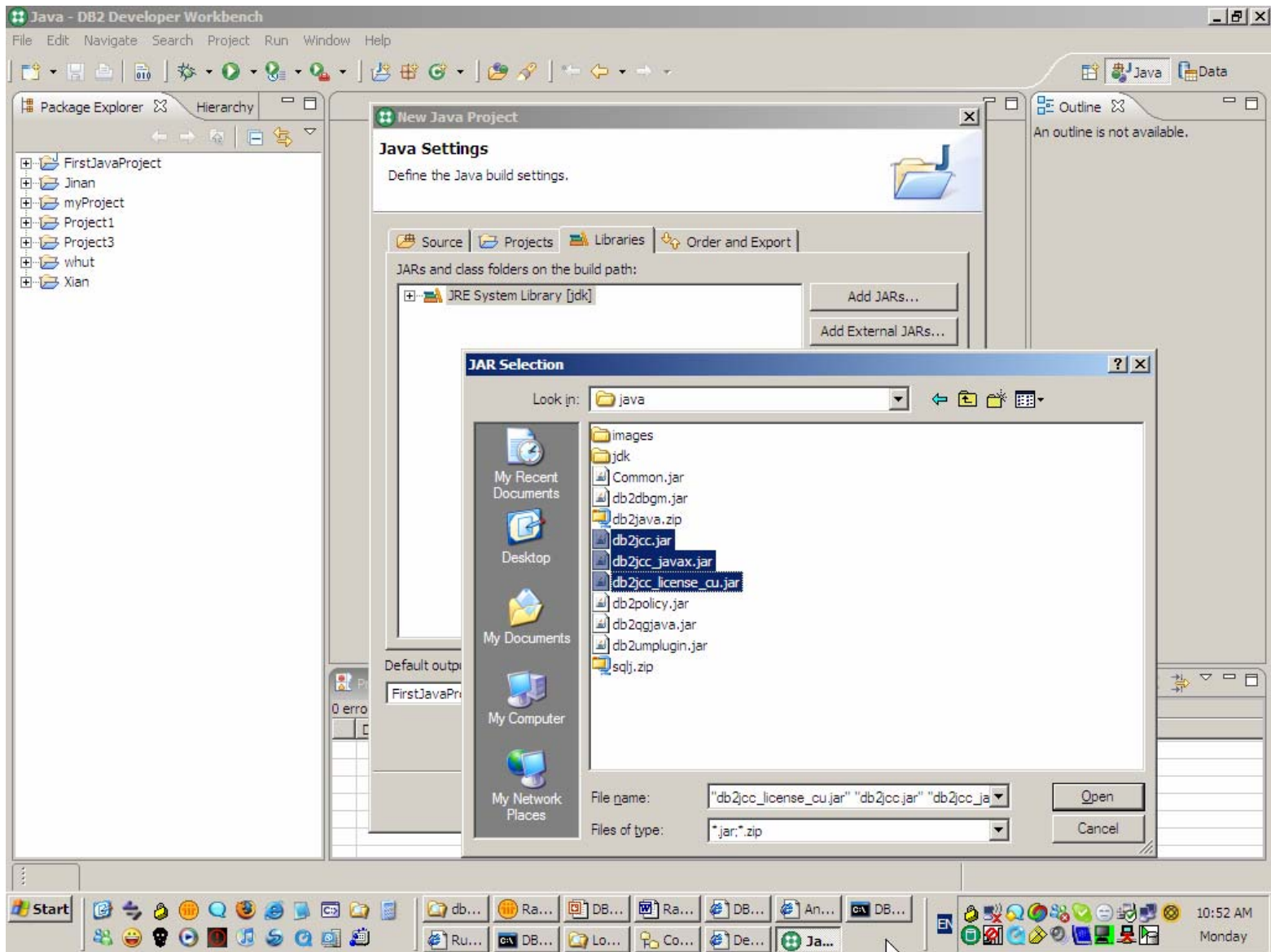
IBM DB2 Driver for JDBC (aka JCC Driver)

- Java driver optimized for all DB2 servers
 - ▶ DB2 for i5/OS (a.k.a iSeries or AS/400)
 - ▶ DB2 for zOS (v8)
 - ▶ DB2 for Linux, UNIX, Windows
- Single driver can be used in type 2 and type 4 modes
 - ▶ db2jcc.jar (com.ibm.db2.jcc)
 - ▶ Type 2 (Requires a DB2 client)
 - ▶ Type 4 (Pure Java client, no need for a DB2 client)
- db2jcc.jar is included in:
 - ▶ DB2 client
 - ▶ DB2 Runtime Client
 - ▶ IBM DB2 Driver for JDBC and SQLJ

Basics

- Can use the DB2 Developer Workbench (DWB)
 - Based on Eclipse
 - Use Java Perspective (Create a new project)
 - For the Java project, add external JARs and class folders to the build path:
 - db2jcc.jar
 - db2jcc_javax.jar
 - db2jcc_license_cu.jar.

Basics



Database connection

```
public class Conn {  
    // for simplicity, I've hard-coded account and URL data.  
    private static String user = "user1";  
    private static String pwd = "mypassword";  
    private static String url = "jdbc:db2:test";  
  
    // this method gets a database connection  
    public static Connection getConn(){  
        Connection conn=null;  
  
        // load the appropriate DB2 driver and  
        // get a connection to the "test" database  
        try {  
            Class.forName("com.ibm.db2.jcc.DB2Driver");  
            conn = DriverManager.getConnection(url, user, pwd);  
            . . .  
        }  
        catch (Exception e) { e.printStackTrace(); }  
        return conn;  
    }  
    // end getConn();  
  
    // this method closes a database connection  
    public static void closeConn(Connection conn){  
        try {  
            if(conn == null) { return; }  
            conn.close();  
        }  
        catch (Exception e) { e.printStackTrace(); }  
        finally {  
            try { conn.close(); }  
            catch (Exception e) { }  
        }  
    }  
    // end closeConn();  
}  
// end class
```

Inserting XML data from a file

```
public static void insertFile(){
    try {
        // for simplicity, I've defined variables with input data
        int id = 1885;
        String name = "Amy Liu";
        String status = "Silver";
        String fn = "c:/XMLFiles/Client1885.xml"; // input file

        // get a connection
        Connection conn = Conn.getConnection();

        // define string that will insert file without validation
        String query = "insert into clients (id, name, status, contactinfo) values (?, ?, ?, ?)";

        // prepare the statement
        PreparedStatement insertStmt = conn.prepareStatement(query);
        insertStmt.setInt(1, id);
        insertStmt.setString(2, name);
        insertStmt.setString(3, status);
        File file = new File(fn);
        insertStmt.setBinaryStream(4, new FileInputStream(file), (int)file.length());

        // execute the statement
        if (insertStmt.executeUpdate() != 1) {
            System.out.println("No record inserted.");
        }
        conn.close();
    }
    catch (Exception e) { . . . }
}
```

Inserting XML data from a character string

```
public static void insertString(){
    try {
        // for simplicity, I've defined variables with input data
        int id = 1885;
        String name = "Amy Liu";
        String status = "Silver";
        String xml =
            "<?xml version='1.0'?'>" +
            "<Client>" +
            "<Address>" +
            "<street>54 Moorpark Ave.</street>" +
            "<city>San Jose</city>" +
            "<state>CA</state>" +
            "<zip>95110</zip>" +
            "</Address>" +
            "<phone>" +
            "<work>4084630110</work>" +
            "<home>4081114444</home>" +
            "<cell>4082223333</cell>" +
            "</phone>" +
            "<fax>4087776688</fax>" +
            "<email>sailer555@yahoo.com</email>" +
            "</Client>";

        // get a connection
        Connection conn = Conn.getConn();

        // define string that will insert file without validation
        String query = "insert into clients (id, name, status, contactinfo) values (?, ?, ?, ?)";

        // prepare the statement
        PreparedStatement insertStmt = conn.prepareStatement(query);
        insertStmt.setInt(1, id);
        insertStmt.setString(2, name);
        insertStmt.setString(3, status);
        insertStmt.setString(4, xml);

        // execute the statement
        if (insertStmt.executeUpdate() != 1) {
            System.out.println("No record inserted.");
        }
        conn.close();
    } catch (Exception e) { . . . }
}
```

Retrieving full XML documents with SQL

```
import java.sql.*;

public static void simpleQuery() {
    PreparedStatement selectStmt = null;
    String query = null; stringDoc = null;
    ResultSet rs = null;
    int clientID = 1885;

    try{
        // get a connection
        Connection conn = Conn.getConn();

        // define, prepare, and execute the query
        // this will retrieve all XML data for a specific client
        query = "select contactinfo from clients where id = " + clientID;
        selectStmt = conn.prepareStatement(query);
        rs = selectStmt.executeQuery();

        // check for results
        if (rs.next() == false) {
            System.out.println("Can't read document with id " + clientID);
        }

        // fetch XML data as a string and print the results
        else {
            stringDoc = rs.getString(1);
            System.out.println(stringDoc);
        }
        conn.close();
    }
    catch (Exception e) { . . . }
}
```


Retrieving part of XML document with SQL/XML

```
String status = "Silver";

try{
    // get a database connection
    // define, prepare, and execute a query that includes
    // (1) a path expression that will return an XML element and
    // (2) a parameter marker for a relational column value
    String query = "SELECT name, xmlquery('$c/Client/email[1]' " +
        " passing contactinfo as \"c\") " +
        " from clients where status = ?";
    PreparedStatement selectStmt = conn.prepareStatement(query);
    selectStmt.setString(1, status);
    ResultSet rs = selectStmt.executeQuery();

    // iterate over and print the results
    while(rs.next() ){
        System.out.println("Name: " + rs.getString(1) +
            "      Email:  " + rs.getString(2));
    }

    // release resources
}
catch (Exception e) { . . . }
```

Retrieving part of XML document with SQL/XML

```
String status = "Silver";
String state = "CA";
String city = "San Jose";
try{
    String query = "SELECT name, xmlquery('$c/Client/email[1]' " +
        "passing contactinfo as \"c\") " +
        "from clients where status = ?" +
        "and xmlexists('$c/Client/Address[state=$state][city=$city]' " +
        "passing contactinfo as \"c\", " +
        "cast(? as char(2)) as \"state\", " +
        "cast(? as varchar(30)) as \"city\" )";
    PreparedStatement selectStmt = conn.prepareStatement(query);
    selectStmt.setString(1, status);
    selectStmt.setString(2, state);
    selectStmt.setString(3, city);
    . . .
}
```

Retrieving part of XML document with XQuery

```
try{
    // get a database connection
    Connection conn = Conn.getConn();

    // define, prepare, and execute an XQuery (without SQL).
    // note that we must hard-code query predicate values.
    String query = "xquery for $y in db2-fn:xmlcolumn" +
        " ('CLIENTS.CONTACTINFO')/Client " +
        "where $y/Address/city='San Jose' and $y/Address/state='CA'" +
        "return <emailList> { $y/email } </emailList>";
    PreparedStatement selectStmt = conn.prepareStatement(query);
    ResultSet rs = selectStmt.executeQuery();

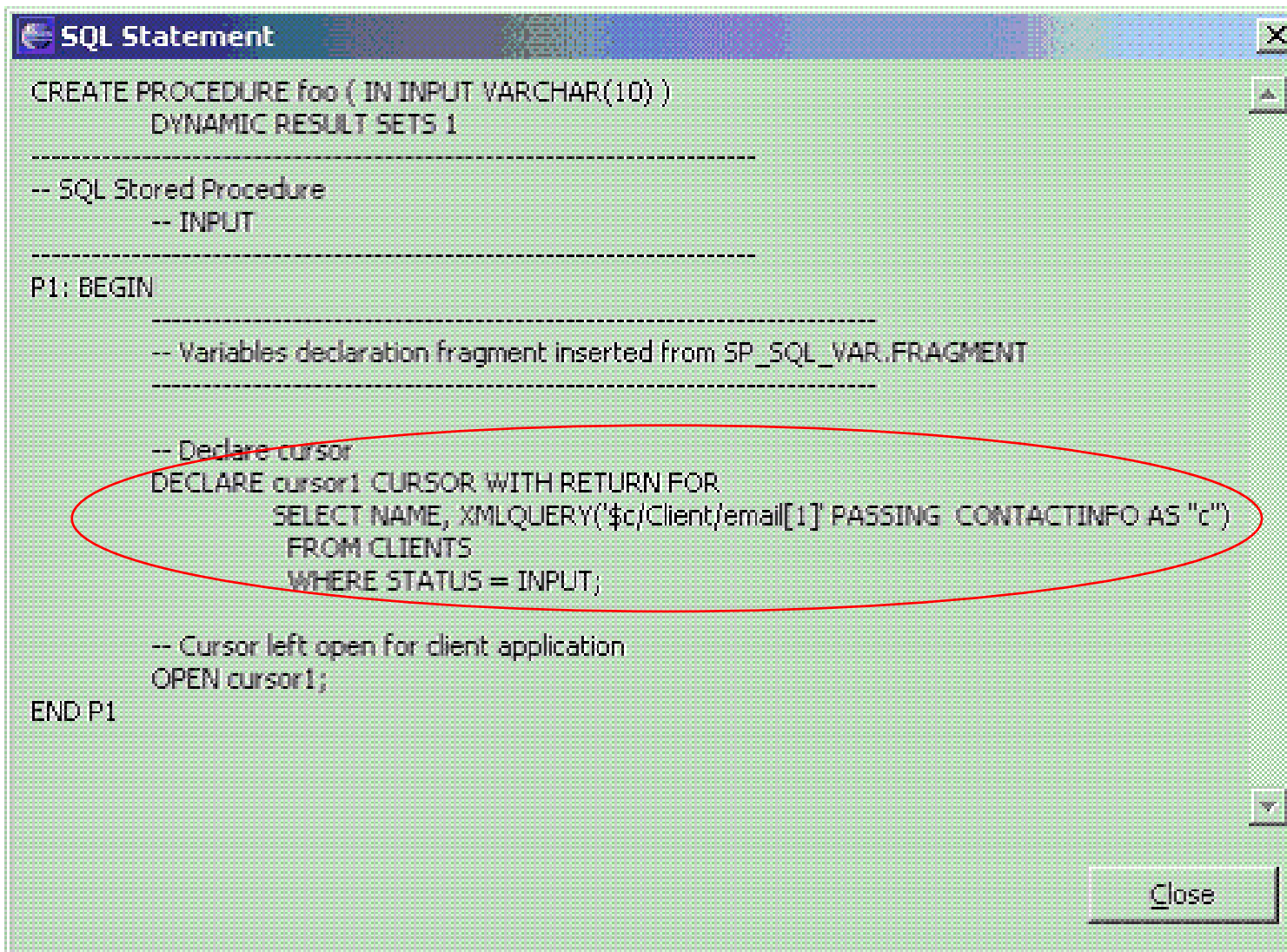
    // iterate over all items in the sequence and print results.
    while(rs.next() ){
        System.out.println(rs.getString(1));
    }

    // release all resources
    // catch and handle any exceptions
    ...
}
```

Update/delete part of XML document

```
String state = "ME";
String query = "delete from clients " +
" where xmlexists('$y/Client/Address[state=$state]' " +
" passing clients.contactinfo as \"y\", " +
" cast(? as char(2)) as \"state\" )";
. . .
PreparedStatement stmt = conn.prepareStatement(query);
stmt.setString(1, state);
. . .
```

SQL PL Stored procedures



```
SQL Statement
CREATE PROCEDURE foo ( IN INPUT VARCHAR(10) )
    DYNAMIC RESULT SETS 1
-----
-- SQL Stored Procedure
-- INPUT
-----
P1: BEGIN
-----
-- Variables declaration fragment inserted from SP_SQL_VAR.FRAGMENT
-----

-- Declare cursor
DECLARE cursor1 CURSOR WITH RETURN FOR
    SELECT NAME, XMLQUERY('$c/Client/email[1]' PASSING CONTACTINFO AS "c")
    FROM CLIENTS
    WHERE STATUS = INPUT;

-- Cursor left open for client application
OPEN cursor1;

END P1
```

Close

Ruby on Rails



What is Ruby on Rails?

- Ruby: Object oriented language
 - Invented by Yukihiro Matsumoto (“Matz”) - 1995
- Rails: Web Framework created using Ruby
- Creates database backed web-based applications
- Agile Software Development
- Hot new technology



Ruby on Rails (RoR) highlights

- Rails based on MVC (Model, View, Controller) architecture
- Convention over configuration
- Easy to perform testing
- In development, ships with a Ruby web server WEBrick
- Popular IDE: RadRail



TPC Index <http://www.tiobe.com/tpci.htm>

Position Jan 2007	Position Jan 2006	Delta in Position	Programming Language	Ratings Jan 2007	Delta Jan 2006	Status
1	1	=	Java	19.160%	-3.10%	A
2	2	=	C	15.807%	-3.20%	A
3	3	=	C++	10.425%	-1.04%	A
4	5	↑	(Visual) Basic	9.123%	+0.03%	A
5	4	↓	PHP	7.943%	-1.46%	A
6	6	=	Perl	6.237%	-0.81%	A
7	7	=	C#	3.521%	-0.03%	A
8	8	=	Python	3.502%	+0.90%	A
9	10	↑	JavaScript	2.845%	+1.31%	A
10	21	11 * ↑	Ruby	2.519%	+2.15%	A
11	11	=	SAS	2.343%	+1.18%	A
12	9	↓↓↓	Delphi	2.336%	+0.75%	A
13	12	↓	PL/SQL	1.570%	+0.54%	A

Job market (Source: indeed.com)

Ruby Job Trends

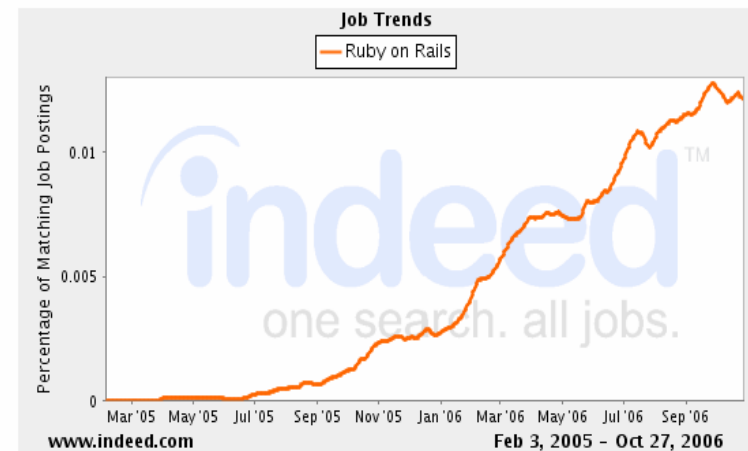
Enter some words, separated by commas



Indeed.com searches millions of jobs from thousands of job sites.
This job trends graph shows the percentage of jobs we find that contain your search terms.

Ruby on Rails Job Trends

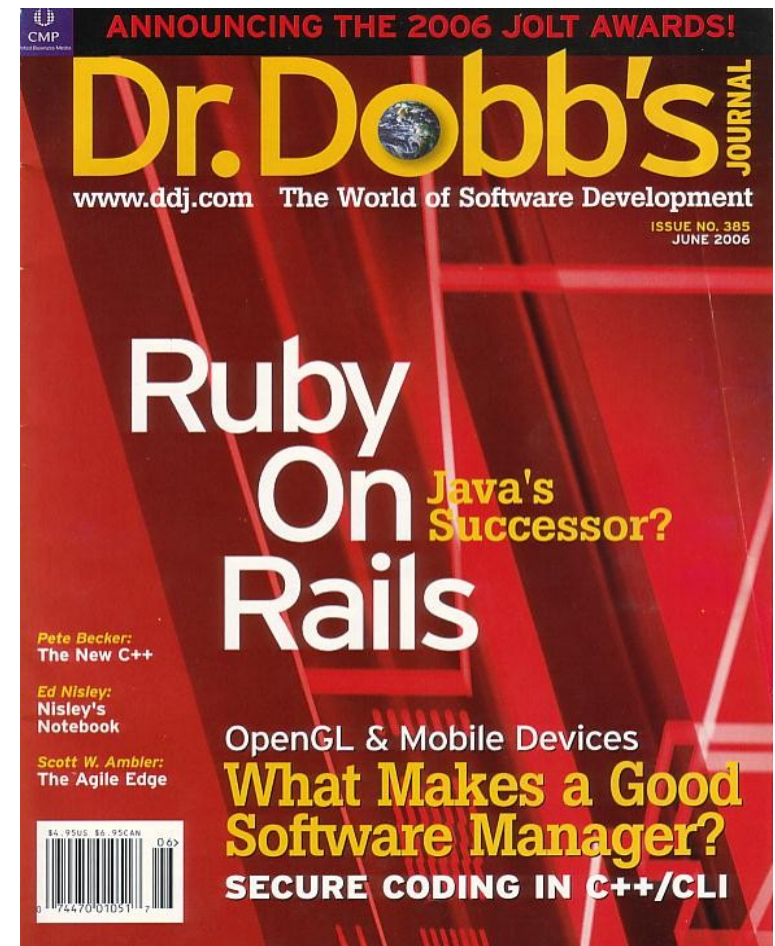
Enter some words, separated by commas



Indeed.com searches millions of jobs from thousands of job sites.
This job trends graph shows the percentage of jobs we find that contain your search terms.

Is Ruby on Rails the next Java?

- Some of the most respected Java and PHP people think so
- Similar demographic as PHP but much faster adoption curve
- PHP community is trying to respond with their own MVC frameworks but most PHP programmers are not embracing MVC
- Enterprise qualities (MVC) but several orders of magnitude more productive than Java
- All-in-one framework: no assembly required, batteries included



Productivity... just an example

```
import java.io.BufferedReader;
import java.io.FileReader;
import java.io.IOException;
import java.util.regex.Matcher;
import java.util.regex.Pattern;

public class Test
{
    public static void main(String[] args)
    {
        try {
            BufferedReader in = new BufferedReader(new FileReader("test.txt"));
            StringBuffer sb = new StringBuffer();
            String str;
            while ((str = in.readLine()) != null)
            { sb.append(str + "\n"); }
            in.close();
            String result = sb.toString();
            Pattern sentencePattern = Pattern.compile("(.*?\n.)\\s+?");
            Pattern javaPattern = Pattern.compile("Ruby");
            Matcher matcher = sentencePattern.matcher(result);
            while (matcher.find()) {
                String match = matcher.group();
                Matcher matcher2 = javaPattern.matcher(match);
                if (matcher2.find())
                    System.err.println(match);
            }
        } catch (IOException e)
        {
            e.printStackTrace();
        }
    }
}
```

`File.open('test.txt').each_line{ || puts l.chomp if l =~ /Ruby/ }`

Credit: Java code by: Peter Szinek, MSc
<http://www.rubyrailways.com>

Ruby utilities

- RI: Ruby Interactive
- IRB: Interactive Ruby Shell
- RubyGems

```
C:\Users\Antonio>ri String#chomp
```

```
----- String#chomp
str.chomp(separator=$/) => new_str
-----
```

Returns a new +String+ with the given record separator removed from the end of _str_ (if present). If +\$/+ has not been changed the default Ruby record separator, then +chomp+ also removes carriage return characters (that is it will remove +\n+, +\r+, or +\r\n+).

```
"hello".chomp      #=> "hello"
"hello\n".chomp    #=> "hello"
"hello\r\n".chomp  #=> "hello"
"hello\n\r".chomp  #=> "hello\n"
"hello\r".chomp     #=> "hello"
"hello\n there".chomp #=> "hello\n there"
"hello".chomp("llo") #=> "he"
```

```
C:\Users\Antonio>irb --simple-prompt
```

```
>> 3+2
=> 5
>> 60*60*24*7*365
=> 220752000
>> puts "Hello Ruby"
Hello Ruby
=> nil
>> exit
```

```
C:\Users\Antonio>gem install Rails --include-dependencies
```

```
Successfully installed rails-1.1.6
Successfully installed activessupport-1.3.1
Successfully installed activerecord-1.14.4
Successfully installed actionpack-1.12.5
Successfully installed actionmailer-1.2.5
Successfully installed actionwebservice-1.1.6
Installing ri documentation for activessupport-1.3.1...
Installing ri documentation for activerecord-1.14.4...
...
```

Startup Toolkit for DB2 on Rails

- Integrated installer that creates a complete DB2 Ruby on Rails development environment on a Windows PC.
- Source code available to do your own builds on other platforms
- Version 2.1 available on IBM alphaWorks
<http://www.alphaworks.ibm.com/tech/db2onrails>
- Helps you install Ruby 1.8.4 and Rails 1.1.6, DB2 Express – C 9 (includes pureXML support)
- IBM developed DB2 Ruby driver and DB2 Rails Adapter

Ruby on Rails Demo (Aurora)



Sign in

Google Web Images Groups News Maps Desktop more »


stolen laptop and private data Search Advanced Search Preferences

Search: ☒ the web ☐ pages from Canada

The "AND" operator is unnecessary – we include all search terms by default. [details]

Web Results 1 - 10 of about 2,100,000 for **stolen laptop and private data**. (0.49 seconds)

News results for **stolen laptop and private data** - View today's top stories

 **Data** is main cost in **laptop** theft - Australian IT - 3 Jul 2006
 Analysis: Government **Data** Security Guidelines Could Lack Teeth - InformationWeek - 2 Jul 2006

Another security breach reported / Stolen laptop had clients ...
Stolen laptop had clients' **private data**, says Ernst & Young ... statement saying that the **laptop** was password protected, and appeared to have been **stolen** in ...
www.sfgate.com/cgi-bin/article.cgi?f=/c/a/2006/02/25/BUG2IHEGCC1.DTL - 29k -
[Cached](#) - [Similar pages](#)

Another security breach reported / Stolen laptop had clients ...
Stolen laptop had clients' **private data**, says Ernst & Young - Carrie Kirby, Chronicle Staff
 Writer Saturday, February 25, 2006 ...
www.sfgate.com/cgi-bin/article.cgi?file=/c/a/2006/02/25/BUG2IHEGCC1.DTL&type=printable -
 11k - [Cached](#) - [Similar pages](#)
[\[More results from www.sfgate.com \]](#)

ING loses another laptop with private data ZDNet Australia: News ...
 ING loses another **laptop** with **private data**. By AAP 19 June 2006 08:33 AM ... employees
 and former employees of the city of Washington, DC, has been **stolen**. ...
www.zdnet.com.au/news/business/soa/ING_loses_another_laptop_with_private_data/0,39023166,39260328,00.htm - 47k -
[Cached](#) - [Similar pages](#)

InfoWatch : Finance firm ING loses private data on 13000 clients
 The **private** details of 13000 ING US Financial Services clients were on a ... Last December a
laptop was **stolen** that contained unencrypted **data** on 8500 ...
www.infowatch.biz/threats?chapter=148831545&id=189411099 - 16k - [Cached](#) - [Similar pages](#)

“Wouldn’t it be nice to give the **business users** the power to create departmental applications **quickly**, with **secured data**, and which are **easily deployable**?”





aurora

TAKE BACK *CONTROL*





- Web based application builder
- Allow the business user to:
 - Create departmental applications
 - IT manage data securely
 - Deploy over the web



Aurora Facts

- Created with Ruby on Rails and DB2
- Developed in **10** weeks
- Developed by **3** programmers who had never seen Ruby on Rails or DB2 previously



Ruby on Rails Lab

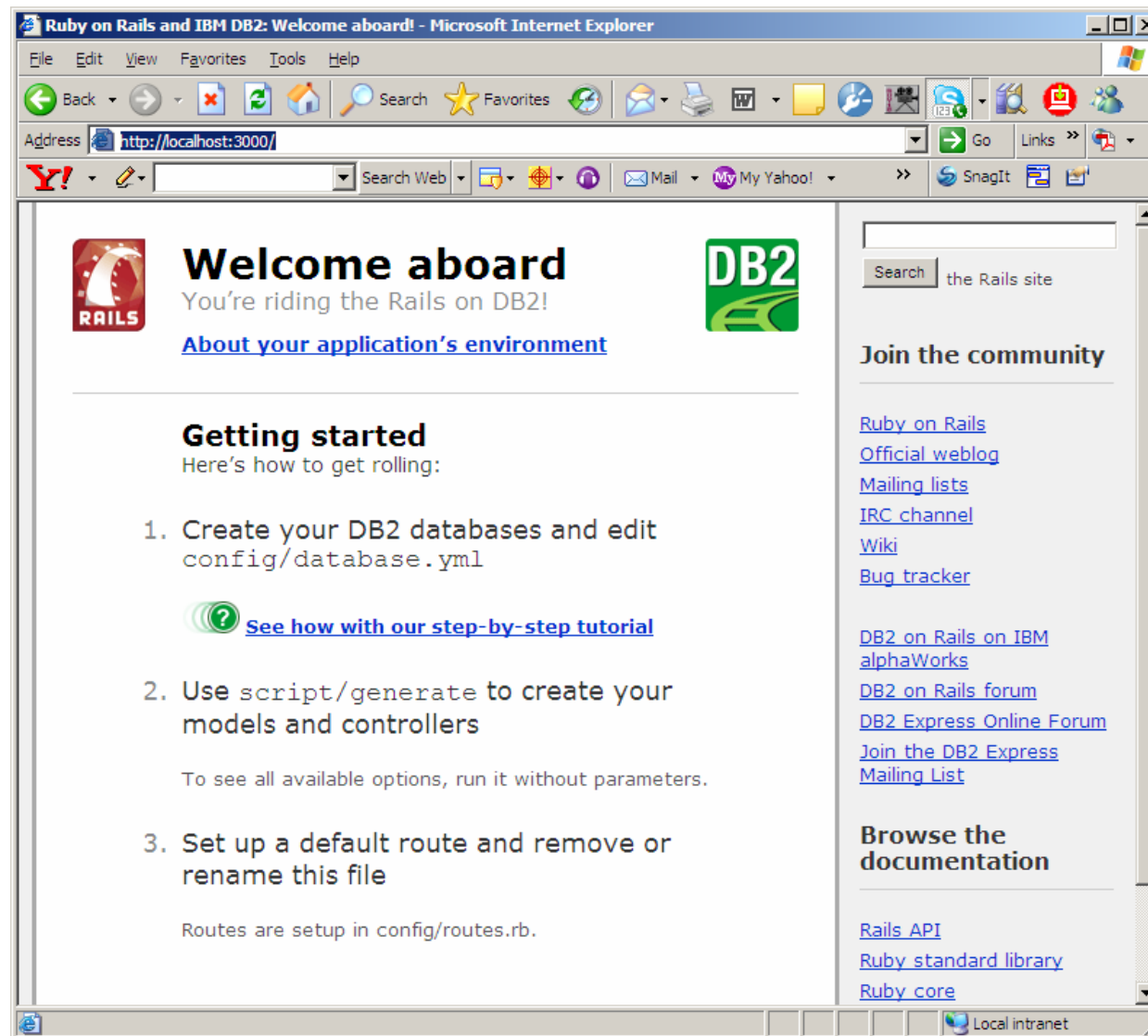


Ruby on Rails Lab

- 1. Create 3 databases (For development, test, production)**
- 2. rails softdir (softdir directory will be created)**
- 3. Edit ...softdir\config\database.yml**
 - ▶ This file has info for connecting to DB2 databases
- 4. Start the built-in Web Server:**
 - ▶ cd ...softdir\script\
 - ▶ ruby server
- 5. Start a browser and redirect to <http://localhost:3000/>**



Ruby on Rails Lab



Ruby on Rails Lab

6. Create the MODEL for the “projects” table

```
cd C:\softdir\script
```

```
ruby generate model Project
```

7. Edit C:\softdir\db\migrate\001_create_projects.rb to add table DDL

8. Run the “migration” (Build the table from Rails)

- ▶ rake migrate

9. Check in DB2 Control Center the table(s) created



Ruby on Rails Lab

Control Center - DB2COPY1

Control Center Selected Edit View Tools Help

Object View

Control Center

- All Systems
 - RAULCHONG
 - Instances
 - DB2
 - Databases
 - RESEARCH
 - SAMPLE
 - SOFTD_D
 - Tables**
 - Views
 - Aliases
 - Nicknames
 - Cache Objects
 - Triggers
 - Schemas
 - Indexes
 - Table Spaces
 - Event Monitors
 - Buffer Pools
 - Applications
 - User and Groups

RAULCHONG - DB2 - SOFTD_D - Tables

Name	Schema	Table space	Comment	Index table space
PROJECTS	ARFCHO...	USERSPACE1		
SCHEMA_INFO	ARFCHO...	USERSPACE1		

111 of 111 items displayed

Table - PROJECTS

Schema : ARFCHONG
Creator : ARFCHONG
Columns : 9

Actions:

- [Open](#)
- [Query](#)
- [Show Related Objects](#)
- [Create New Table](#)

Columns

K...	Name	Data type	Length	Nullable
	ID	INTEGER	4	No
	NAME	VARCHAR	255	No
	DESCRIPTION	CLOB	1048576	No
	HOMEPAGE	VARCHAR	255	Yes
	LICENSE	VARCHAR	255	Yes
	OPENSOURCE	SMALLINT	2	No
	NOTES	CLOB	1048576	Yes
	UPDATED ON	TIMESTAMP	10	Yes
	CREATED ON	TIMESTAMP	10	Yes

Ruby on Rails Lab

10. Create the user interface

- `cd C:\softdir\script`
- `ruby generate scaffold Project Admin`
- scaffolding creates basic code to support create, update, delete (CRUD) operations on the model.
- <http://localhost:3000/admin>
shows the interface where you can add, edit, and remove records



PHP



What is PHP?

***PHP is the most popular Web language in the world today:
70% of the UNIX-based, Apache Web platform market... 40% + of the overall market***

- PHP is an open source, platform independent scripting language that is designed for web application development
- Originally created by Rasmus Lerdorf in 1995
 - ▶ Set of Perl scripts for tracking accesses to his online resume
 - ▶ Original name was 'Personal Home Page Tools'
- PHP 3.0 was a completed rewrite in 1997
- Number of PHP developers has been growing ever since
- It is one of the most widely deployed web languages in the world today .
- Oh yes, PHP means (PHP Hypertext Preprocessor)



Why PHP is popular ?



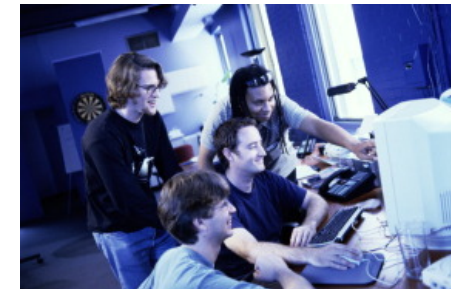
Rapid, iterative development cycles with a low learning curve



Robust, high-performance & scalable; stable & secure



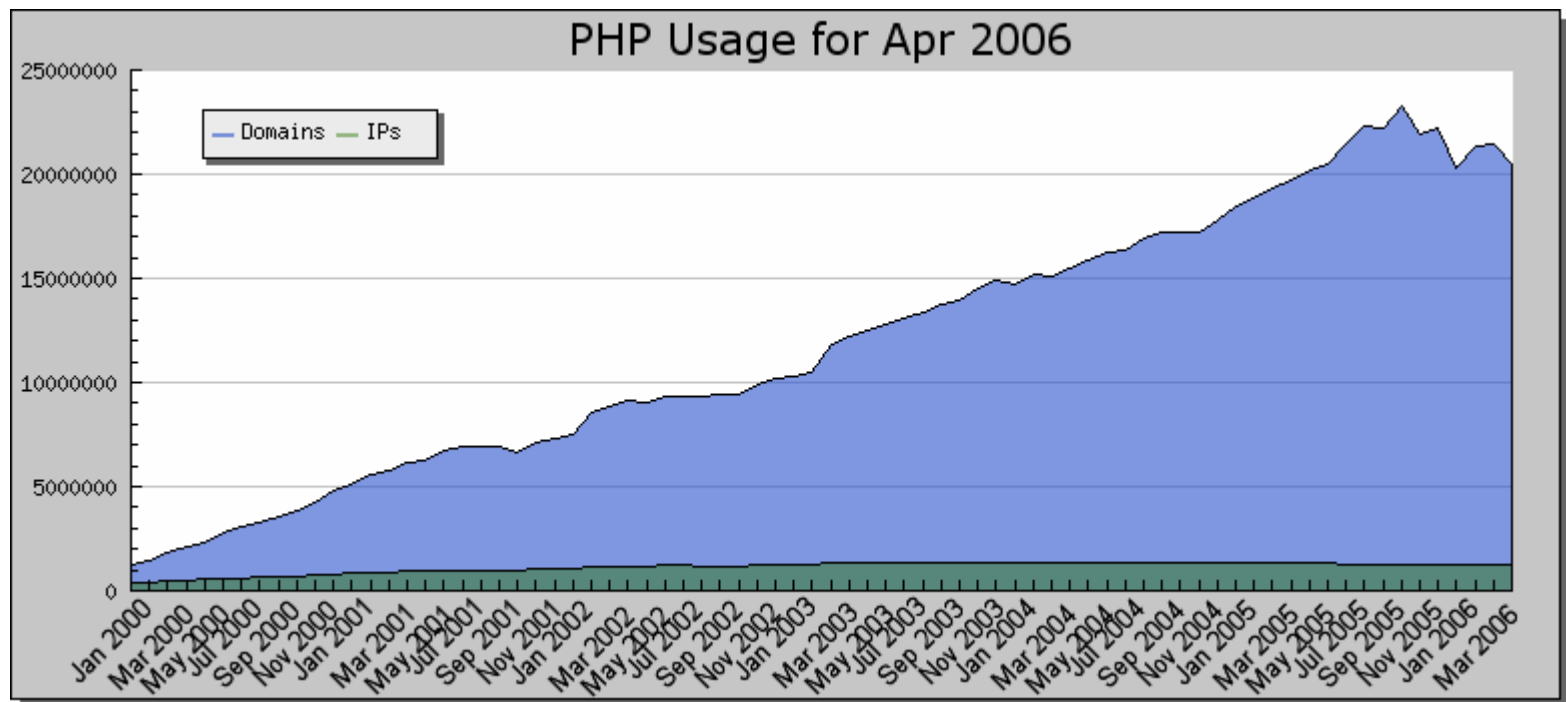
Easily integrated into heterogeneous environments/systems



Proven through widespread deployment; vibrant community

PHP Usage Continues to Expand

- TIOBE Programming Community Index for February 2006
 - ▶ <http://www.tiobe.com/tpci.htm>
- Netcraft – PHP usage
 - <http://www.php.net/usage.php>



PHP – 10 years and just getting started !!

- **First and Foremost**
 - **a simple programming language for the web**
 - **a server-side scripting language (simplified development & deployment)**
 - *Rapid iterative development*
 - **evolving – alternative to J2EE and .NET on the web**
 - Object oriented language features
 - UNICODE support
 - Web Services / XML
 - abstract database interfaces (PDO)
 - Frameworks – PHP Collaboration project
 - http://www.zend.com/php_collaboration_project
- **Well established *community***
 - Core language developers – 10s
 - Language extension developers – 100s
 - Professionals – 1000+ (Zend Certified Engineers)
 - Casual – 1 million +



PHP - Key driver of LAMP Stack

- **LAMP – Technical**
 - ▶ Linux, Apache HTTP Server, MySQL, **PHP**/Perl/Python
 - ▶ Open Source web technology stack
 - Often available on ISPs for reasonable monthly fees
 - ▶ Pure open source software components
- **LAMP – Business (Appealing on many levels)**
 1. Rapid iterative development
 2. Low cost of software acquisition
 3. Scalable and robust scalable LAMP solutions requires significant investment
- **PHP Concerns (CTOs)**
 - ▶ Lack of Professional level PHP developer skills
 - Good software design skills are still important for large projects
 - ▶ Considerable investment in .NET or J2EE already
 - ▶ Many enterprises are not comfortable with MySQL

PHP – Getting Started

welcome.php

```
<?php
$name = $_GET["first"] . ' ' . $_GET["last"];
?>

<h1>Hello <?php echo($name); ?></h1>
```

URL:

<http://localhost/welcome.php?first=Grant&last=Hutchison>

Output:



DB2 Connection options for PHP

- No single abstract data access interface
 - ▶ ODBC/JDBC

- PHP Drivers for DB2
 - ▶ **Recommended one:**
 - Extension for DB2 (ibm_db2)
 - ▶ **New Object Oriented database abstraction layer**
 - PHP Data Objects (PDO)
 - PDO_ODBC – Can be used with DB2 and other databases

- PHP drivers are offered under open-source licenses
 - ▶ <http://www.php.net>



PHP extension for DB2 (ibm_db2)

- Available from the PECL repository under the Apache 2.0 License
 - ▶ Developed and supported by IBM
 - ▶ Full featured support for stored procedures and LOBs
 - ▶ Fast, optimized for DB2
 - ▶ Works with both PHP 4 and PHP 5



Connecting with PHP Data Object (PDO)

- Standard data access interface for PHP
 - ▶ Fast, light weight, and object oriented
 - ▶ PDO_ODBC uses DB2 libraries for native access
 - ▶ Standard database API for multiple database servers
 - ▶ Built into PHP 5.1
 - ▶ <http://pecl.php.net/package/pdo>
 - ▶ http://pecl.php.net/package/PDO_ODBC

Connecting to an uncatalogued database

```
$host = 'localhost';  
$port = 50000;  
$DSN = "DRIVER={IBM DB2 ODBC DRIVER};PORT=$port;  
        HOSTNAME=$host;DATABASE=$database;PROTOCOL=TCPIP;  
        USER=$user;PWD=$password";
```

ibm_db2

```
$uconn = db2_connect($DSN, null, null);
```

PDO_ODBC

```
try {  
    $uconn = new PDO("odbc:$DSN", null, null);  
}  
catch (PDOException $e) { print $e->errmsg(); }
```

Your first ibm_db2 application

```
<?php
$sql = "SELECT name, breed FROM ANIMALS
      WHERE weight < ?";
$conn = db2_connect($database, $user, $password);
$stmt = db2_prepare($conn, $sql);
$res = db2_execute($stmt, array(10));
while ($row = db2_fetch_assoc($stmt)) {
    print "{$row['NAME']} is a {$row['BREED']}.\\n";
}
?>
```

Configuring PHP for ibm_db2

- Modify php.ini (Linux / UNIX):
 extension=ibm_db2.so
 ibm_db2.instance_name=db2inst1
- Modify php.ini (Windows):
 extension=php_ibm_db2.dll
- Or Check out **Zend Core for IBM**



What is Zend Core?

Zend Core is:

A seamless out-of-the-box supported PHP development and production environment

For:

Business-critical web applications

Delivering:

Reliability, **Productivity** and **Flexibility**
needed for running PHP applications

Free download:

<http://ibm.com/software/data/info/zendcore/>

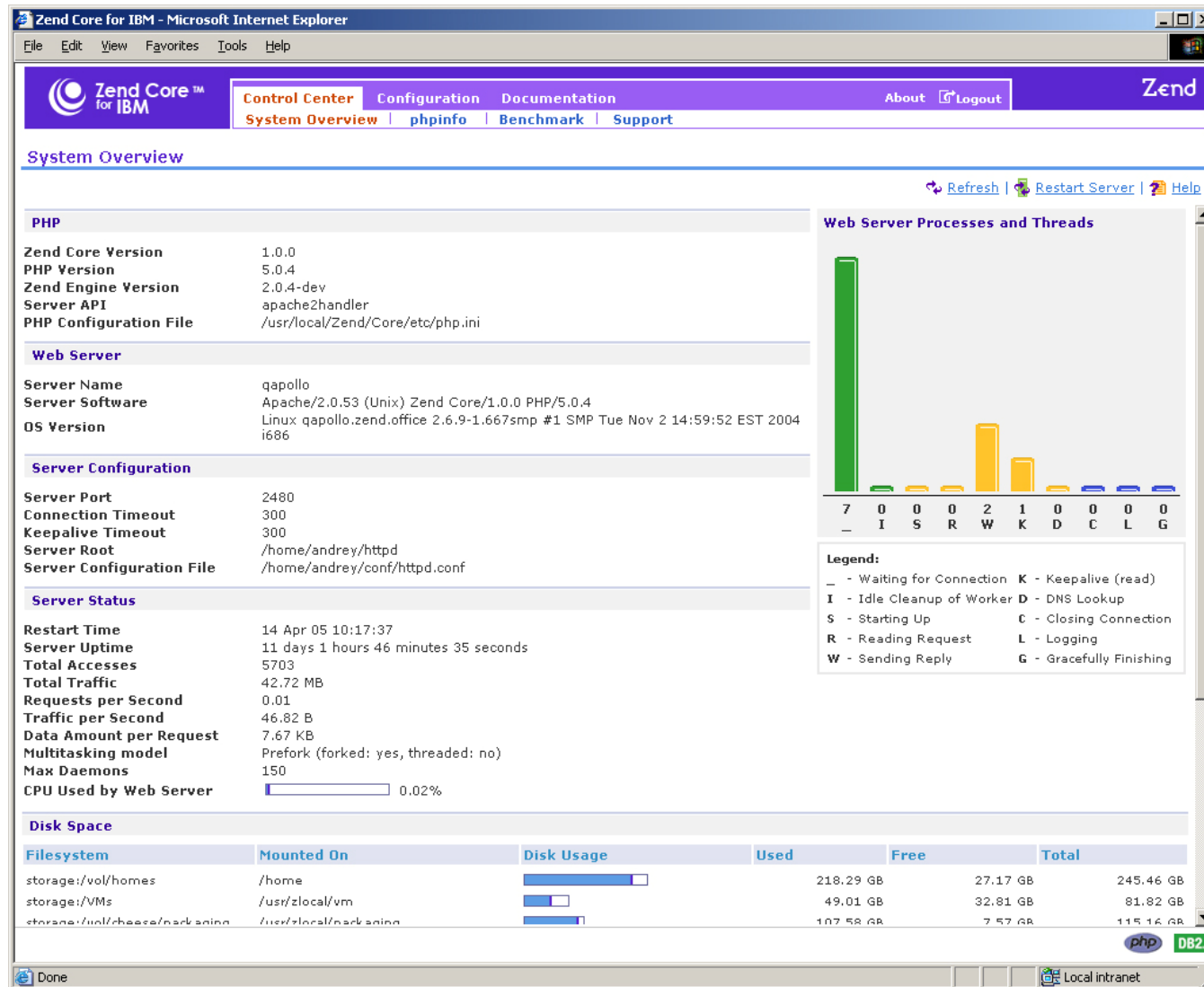


Zend Core for IBM

- Everything you need
 - ▶ Installs
 - DB2 and IDS client
 - Apache HTTP Server (optional)
 - PHP 5.1.4
 - Popular PHP extensions including
 - ibm_db2
 - PDO_INFORMIX
 - Many more...
 - ▶ DB2 Express-C server, IBM Cloudscape server (optional)
 - ▶ Documentation – complete PHP manual
 - ▶ Sample applications for DB2
- Easy to use and configure PHP environment



Zend Core Management & Control



Zend Core PHP Configuration

IBM Cloudscape Server Configuration

Cloudscape Server Status

Server Status	Running
Server Version	CSS10000/10.0.2.0
Server Build	30301
DRDA Product Id	CSS10000
Server Host	localhost
Server Port	1527
Minimal Number of Threads	0
Maximal Number of Threads	0

Java Information

Java Version	1.4.2
Java Vendor	IBM Corporation
Java Classpath	/usr/local/Zend/Core/Cloudscape_10.0/lib/derby.jar; /usr/local/Zend/Core/Cloudscape_10.0/lib/derbytools.jar; /usr/local/Zend/Core/Cloudscape_10.0/lib/derbynet.jar
OS Name	Linux
OS Architecture	x86
OS Version	2.6.9-1.667smp

Extension Configuration

Please Restart Apache | Unsaved configuration | Discard Changes | Save Settings | Restart Server | Help

Zend Core Extensions		
Name	Value	
apache2handler - Apache 2 Handler		
engine	Enable the PHP scripting language engine under Apache	<input checked="" type="radio"/> On <input type="radio"/> Off
last_modified	Send PHP scripts modification date as Last-Modified	<input checked="" type="radio"/> On <input type="radio"/> Off
xbithack	Parse files with executable bit set as PHP regardless of their file ending	<input type="radio"/> On <input checked="" type="radio"/> Off
bcmath- Arbitrary Precision Mathematics		
bz2 - Bzip2 Compression		
calendar - Calendar Functions		
ctype - Character Type Functions		
curl - Client URL Library		
dbase - DBase Access		
dio - Direct IO		

Zend Core Remote Debug & Benchmark

The image displays two screenshots of the Zend Core for IBM web interface, accessed via Microsoft Internet Explorer.

Left Screenshot: Zend Studio Server Settings

The interface shows the Zend Core logo and navigation tabs: Control Center, Configuration (selected), Documentation, About, and Logout. Sub-tabs under Configuration include PHP, Extensions, Zend Products, Other Directives, Zend Studio Server, and Cloudscape. The page title is "Zend Studio Server Settings".

Buttons: Save Settings, Restart Server, Help.

Allowed Hosts

The following Profiling session(s) will be allowed:

Address	Net mask
0.0.0.0	32 (0.0.0.0)

Denied Hosts

The following Profiling session(s) will be denied:

Address	Net mask
0.0.0.0	32 (0.0.0.0)

Allowed Hosts for Tunneling

The following Hosts will be allowed for Debugging:

Address	Net mask
0.0.0.0	32 (0.0.0.0)

Expose Remotely

This setting determines whether the Zend Studio Server will expose its Zend Studio Server that can be Debugged.

Always

Right Screenshot: Benchmark

The interface shows the Zend Core logo and navigation tabs: Control Center (selected), Configuration, Documentation, About, and Logout. Sub-tabs under Control Center include System Overview, PHPinfo, Benchmark (selected), Support, and Updates.

Buttons: Help.

Benchmark

Test URL:

☒ Perform request(s)

☐ Test for second(s)

Concurrent Connections:

Use Keepalive: ☐

Request Headers:

Use Cookies:

Name	Value
<input type="text"/>	<input type="text"/>

Run