



ACTIVATE BUSINESS WITH THE POWER OF I.T.™



# XML Technical Overview

Bill Arledge, Consulting Product Manager  
BMC Software Inc.

4/22/2008

# Agenda



- › What is XML?
- › Why is XML important to your business?
- › Where does PureXML™ in DB2 9 fit in?
- › BMC and XML



# XML Example



```
<Person DOB='1900/01/01'>
```

```
  <Name>
```

```
    <Last>Doe</Last>
```

```
    <First>John</First>
```

```
    <MI>S</MI>
```

```
  </Name>
```

```
  <Address>
```

```
    <Addr-1>900 Any Street</Addr-1>
```

```
    <Addr-2></Addr-2>
```

```
    <City>Anytown</City>
```

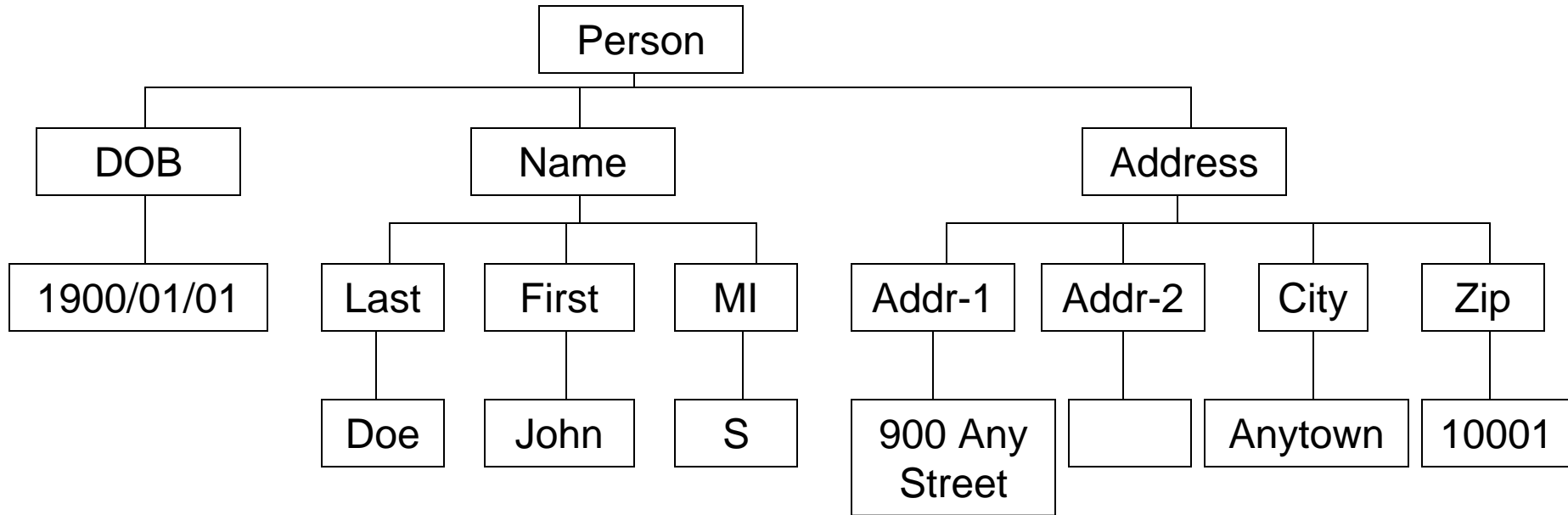
```
    <Zip>10001</Zip>
```

```
  </Address>
```

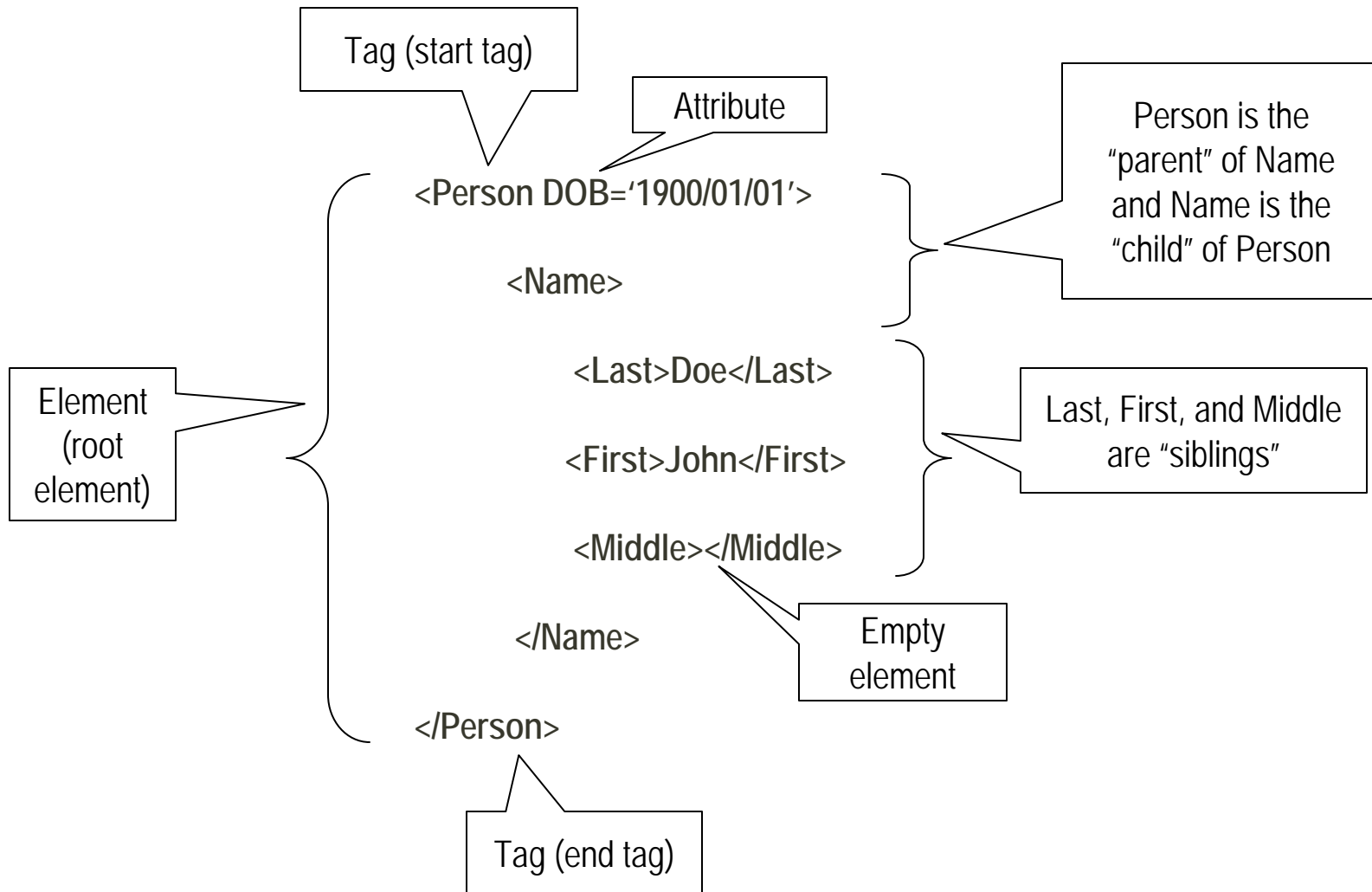
```
</Person>
```



# Tree Representation



# Document Parts



# Advantages of XML



- › Self describing
- › Platform independent
- › Simple syntax
- › Fits any data which can be visualized as a document
- › Electronic Data Interchange (EDI)
- › Necessary component for SOA

# More XML Related Terms



- › XPath: protocol to precisely identify any node in a document
- › XQuery: related protocol to extract data from document or subdocument
- › XSLT: language derived from XML to describe transformations from one document type to another
  - Commonly used to build HTML from XML
- › Schema: language derived from XML to describe contents and semantics of a class of XML documents
- › DTD (Document Type Definition): more limited language to describe structure of a class of XML documents



# More XML Related Terms



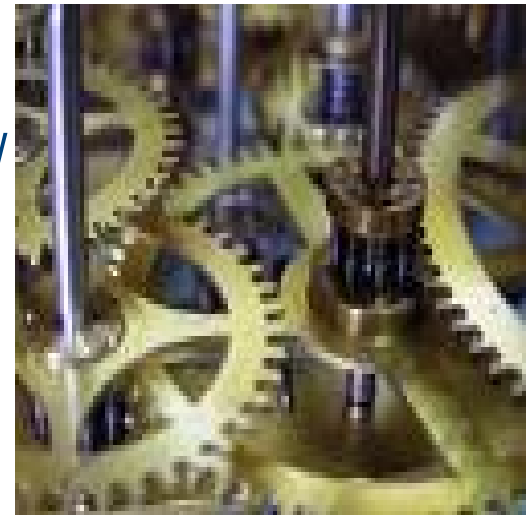
- › XML Parser: software product to parse a text file and perform various levels of syntax checking
- › Well formed: XML document is syntactically legal. There is no semantic implication at all.
- › Validating parser: parser which validates an XML document against a schema.
- › SAX Model: XML parser implementation which uses a callback methodology to allow caller to do something useful with the documents.
- › DOM model: Another implementation which builds a tree representing the document for traversal by the caller.



# Extensions of XML Protocols



- › Many industries are defining standards for document interchange
  - OASIS: Organization for the Advancement of Structured Information Standards
    - Nonprofit group to develop e-business standards
    - 12 “industry focus areas”
  - ACORD: similar group for the Insurance Industry
  - ISO 20022 – securities and banking
  - WSDL (Web Services Description Language)
  - PIDX (Petroleum Industry Data Exchange)



# Issues with XML



› In a word: **persistence**

› 3 primary options

- Specialized XML DBMS
  - Separate installation, learning curve, etc.
- “Shredding” XML documents into relational
  - Slow!
  - Schema changes are hard to handle
  - Need space in each row even for columns which are almost never used
  - Multi table joins
- LOB Usage (or something similar)
  - No direct way to index contents
  - No direct way to access a subset of a document



# XML in DB2 9



- › IBM refers to it as PureXML™
- › Allows storage of XML in DB2 tables
  - Can be accessed with SQL or XQuery
- › DB2 hides complexity from user
- › All the reliability, security, recoverability of the mainframe, with the flexibility and accessibility of XML



# How do I Make it Work?



```
CREATE TABLE INVOICE
(INV_NO INTEGER NOT NULL,
CUST_NO INTEGER NOT NULL,
INV_DATA XML) ...
```

```
CREATE INDEX INV_SHIP_CITY
ON INVOICE(INV_DATA)
GENERATE KEY USING XMLPATTERN
'/Invoice/ShipToAddr/City'
AS VARCHAR(20);
```





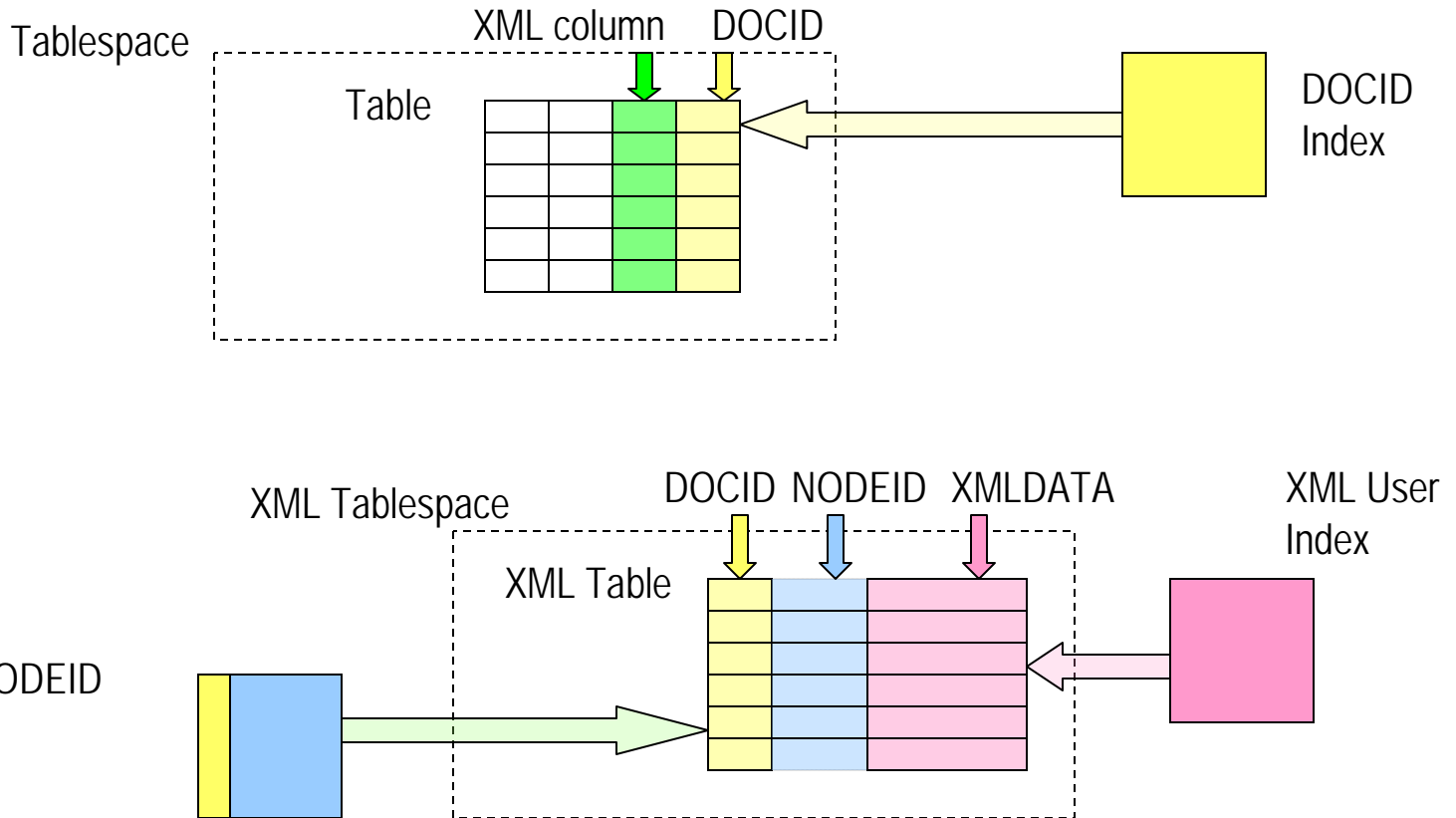
# How do I Make it Work?



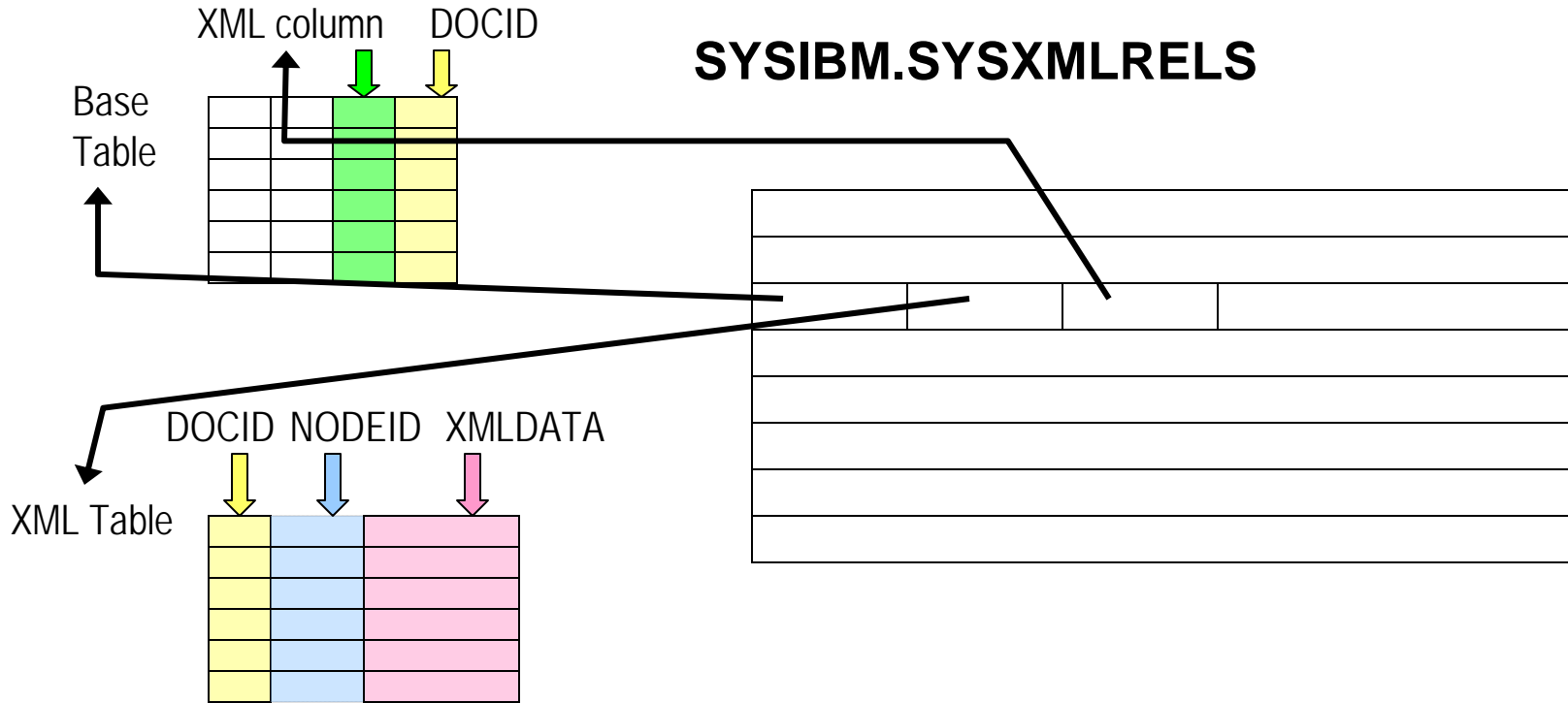
- › Load or insert to get data into XML columns
- › Reorg, Copy, Recover, Rebuild
- › SQL or XQuery to retrieve XML data



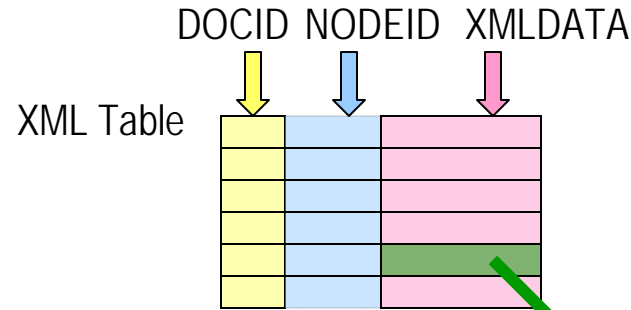
# XML Spaces



# Catalog Tables



# CatalogTables



## SYSIBM.SYSXMLSTRINGS

00010	Person	
00011	DOB	
00013	Name	

...

```
<Person DOB='1900/01/01'>
  <Name>
    <Last>Doe</Last>
    <First>John</First>
    <MI>S</MI>
  </Name>
  <Address>
    <Addr-1>900 Any Street</Addr-1>
    <Addr-2></Addr-2>
    <City>Anytown</City>
    <Zip>10001</Zip>
  </Address>
</Person>
```



# XML User Indexes



- › Extended indexes, vs. simple indexes we know
- › Remember our SQL:

```
CREATE INDEX INV_SHIP_CITY  
ON INVOICE(INV_DATA)  
GENERATE KEY USING XMLPATTERN  
'/Invoice/ShipToAddr/City'  
AS VARCHAR(20);
```



- › One invoice could include multiple ship-to addresses, so
- › More than one index entry (for the same index) can point to the same RID
- › Can have multiple XML indexes for one XML column
- › “AS DECFLOAT” is also an option

# XML and SQL



› Retrieval SQL might look as follows:

```
SELECT XMLQUERY ('//item[productName = $n]'  
    PASSING PO.POrder, P.name as "n")  
    FROM PurchaseOrders PO, Product P;
```

› Or, you could do something like:

```
SELECT POid, POrder FROM PurchaseOrders;
```

– (This is an implicit XMLSERIALIZE)





## › DTD Usage?

- You can use an internal DTD (included with the XML input)
- DTD syntax is parsed, and entity defaults are applied
- The DTD is discarded
- No external DTD support

## › Schema References?

- Schema maintenance is via stored procedures
- XSR\_REGISTER, XSR\_ADDSCHEMADOC, XSR\_COMPLETE, XSR\_REMOVE
- Similar support via JDBC and Java calls
- Schema data is stored in 7 new catalog tables (SYSIBM.XSR\*)
- SYSFUN.DSN\_XMLVALIDATE function





# Caveats



## › Performance

- Data design is still necessary!
- “Extract” static metadata to relational columns
- Don’t use XQUERY without an XML index
- Validation is expensive (but you may have to do it)!
- UPDATE is really delete/insert of XML column
- LOAD uses INSERT like process



# Caveats



## › Maintenance

- Finding names of implicitly created objects is non-trivial
- Use ALL option in LISTDEF
- LOAD must use files for XML columns larger than 32K

## › Migration (DW, Test creation)

- Element and attribute names are not stored with data
- SYSIBM.SYSXMLSTRINGS matches the id actually stored with the name
- DSN1COPY is not an option

# BMC Support for XML



- › IBM has provided a sound infrastructure for XML
- › BMC fully supports this initiative
- › Performance and Recovery support XML as of August, 2007
- › Admin and Utility releases in 2008 will provide “basic support”
- › XML exploitation in the future
  - We want to provide features and tools to help you better use XML
  - We invite your requirements with us



# References



- › To see the XML 1.0 Standard: <http://www.w3.org/TR/xml/>
- › XPATH and XQUERY: <http://www.w3.org/TR/xpath-datamodel/>
- › Schema: <http://www.w3.org/TR/xmlschema-0/>
- › Listing of XML applications and standards: <http://xml.coverpages.org/xmlApplications.html>



# Questions?

