

XML Presentation

Techniques for Sending/Receiving XML Packets Using
SQL Developer 11g Release 2
Thursday, September 22, 2011
10:45AM – 11:45AM
Developer Room 118

Techniques for Sending/Receiving XML Packets Using SQL Developer

11g Release 2

Coleman Leviter

Arrow Electronics

IT Software Systems Engineer

cleviter@ieee.org



CV

- **WMS Group – Thirteen years**
- **VAX Rewrite (.for) to UNIX (.c, .pc)**
- **VAX Forms to Oracle Forms**
- **TIFF file migration to Oracle**
- **XML Development**
- **WMS Development and Support**
- **IOUG Select Contributor '06, '08**
- **ODTUG Journal '09**
- **NYOUG WEB SIG Chair, Steering Committee**
- **IOUG Collaborate '07, '08, '09, '10, '11, '12**
- **Oracle Open World '10, '11**
- **IOUG Development Co-track Manager '10**
- **IOUG Education Co-Chair '11**
- **COLLABORATE 12 - IOUG FORUM CONFERENCE CHAIR**

Presentation Objectives

- Introduction
- Background - Project
- Data Structures
- XML Packet Techniques
- XML Child Node Packets
- CLOB Packets
- Demo
- Questions



Oracle XML History

- **8i 1998 - XML Api**
- **9i 2001 - XML Storage**
- **10g 2004 – XPath, Within**
- **11g 2007 - Binary XML**
- **12g ?????**



Terminology

- XML¹ - Short for eXtensible Markup Language, a specification developed by the World Wide Web Consortium (W3C). It allows designers to create their own customized tags, enabling the definition, transmission, validation, and interpretation of data between applications and between organizations.
- Warehouse Management System - a component of the movement and storage of materials within a warehouse

¹ <http://en.wikipedia.org/wiki/XML>



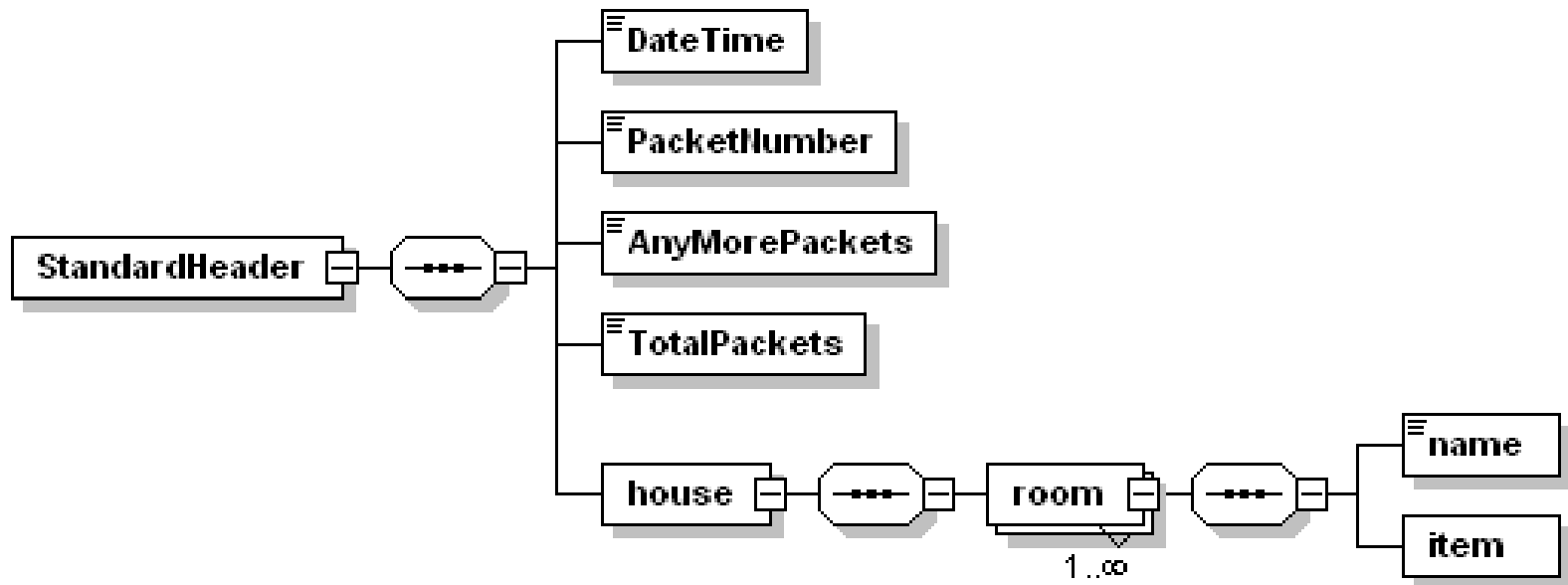
Project Steps

- Develop XSD
- Data Structures
- Temporary Tables
- Use Oracle 11g XML DB for messages
- Use MQ Series for the communications layer – guaranteed message delivery



XSD Design – XML SPY

- XML Schema Definition (XSD) language is the current standard schema language for all XML documents and data.



Generated by XMLSpy

www.altova.com

Nodal Representation

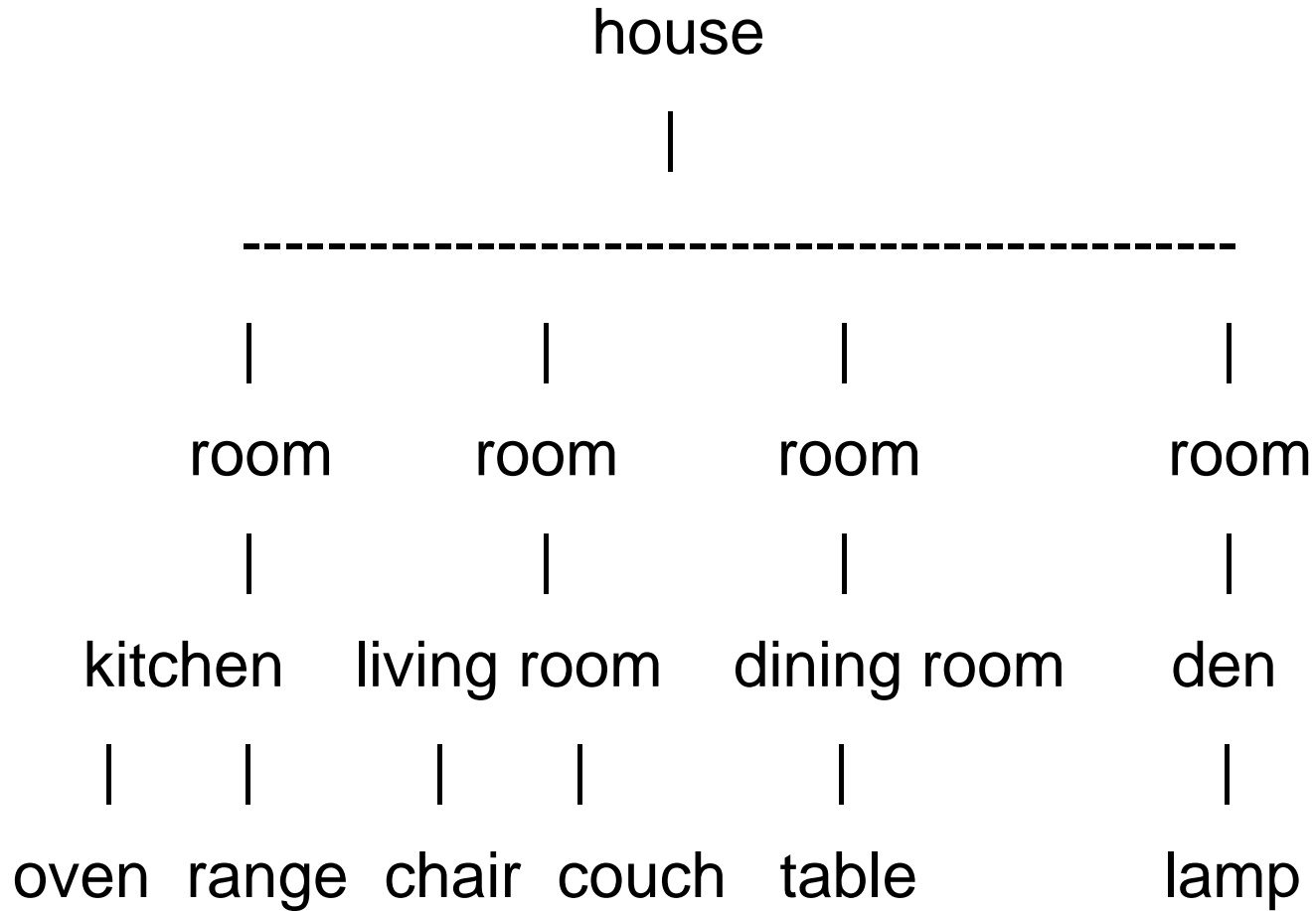


Table Data

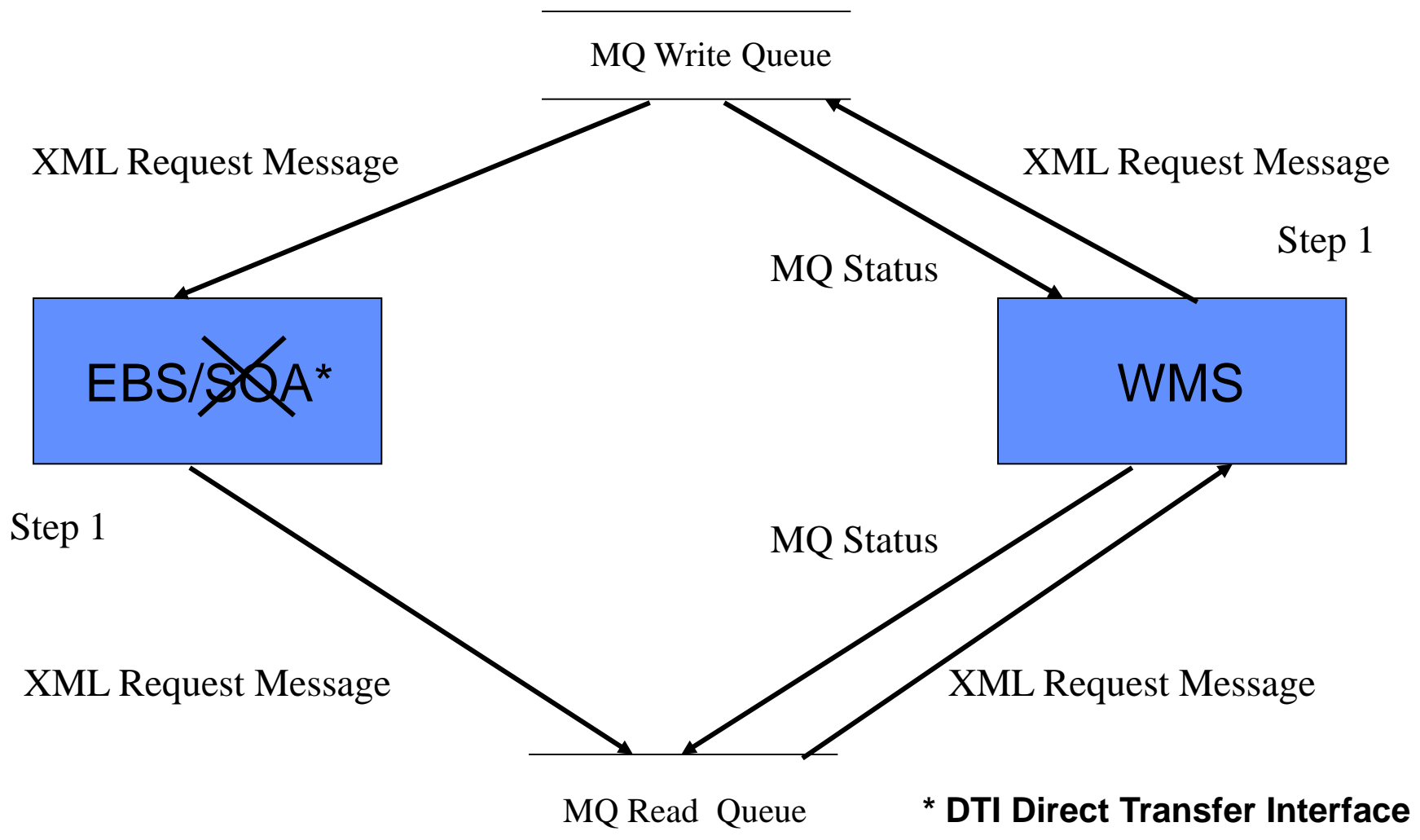
Table Name: **my_relat_table**

| NAME | ITEM |
|-------------|--------------|
| kitchen | sink |
| kitchen | oven |
| kitchen | refrigerator |
| living_room | pictures |
| dining_room | table |
| dining_room | chair |
| dining_room | chandelier |
| den | chair |
| den | lamp |
| den | étagère |

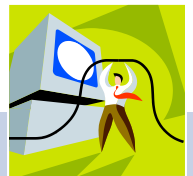
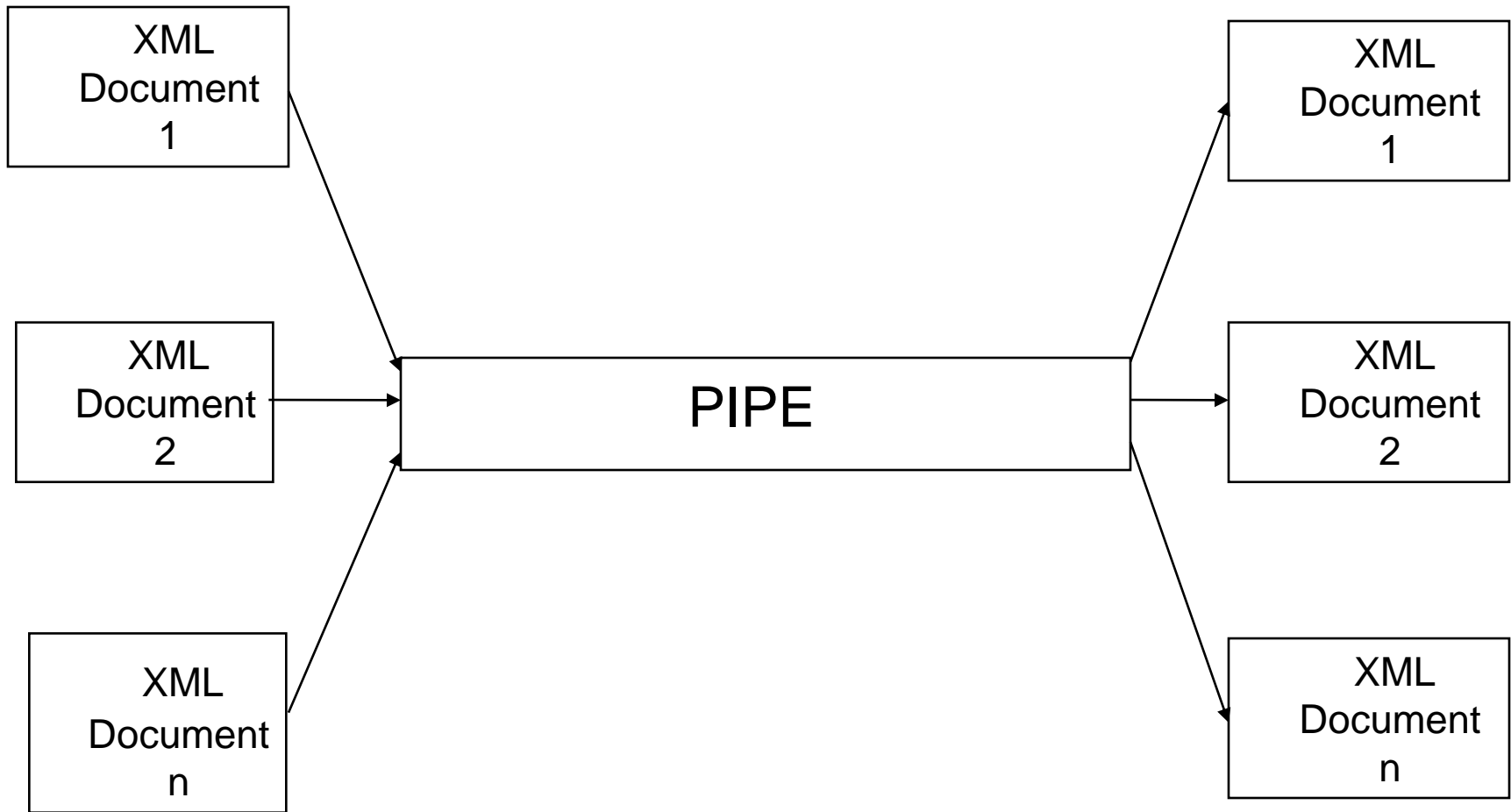
XML Document

```
<?xml version="1.0" encoding="WINDOWS-1252"?>
<house>
  <room>
    <name>kitchen</name>
    <item>sink</item>
  </room>
  <room>
    <name>kitchen</name>
    <item>table</item>
  </room>
  <room>
    <name>kitchen</name>
    <item>counter</item>
  </room>
  <room>
    <name>kitchen</name>
    <item>microwave</item>
  </room>
  etc...
</house>
```

EBS/~~SQA~~*-WMS Data Flow



XML Producer / Consumer Model



Problem

- The pipe is 32k bytes
- XML Documents can range in size upwards past the 32k bytes size of the pipe
- How do know how big to make the pipe?
- Pipe size - 32k bytes? 64K bytes? 128K bytes?
- Where do you stop?



Possible Solutions

- Restrict XML Documents < 32k bytes? ☹️
- Create several different size pipes and use as needed? 😐
- Break up the XML Document and ship through the pipe? 😊



Data Structures

--- **original**

```
CREATE TABLE MY_RELAT_TABLE  
( NAME VARCHAR2(20 BYTE),  
  ITEM VARCHAR2(20 BYTE) );
```

--- **xml, clob**

```
CREATE TABLE MY_XML_TABLE  
( REF_ID NUMBER,  
  XMLCOL SYS.XMLTYPE,  
  CLOBCOL CLOB);
```

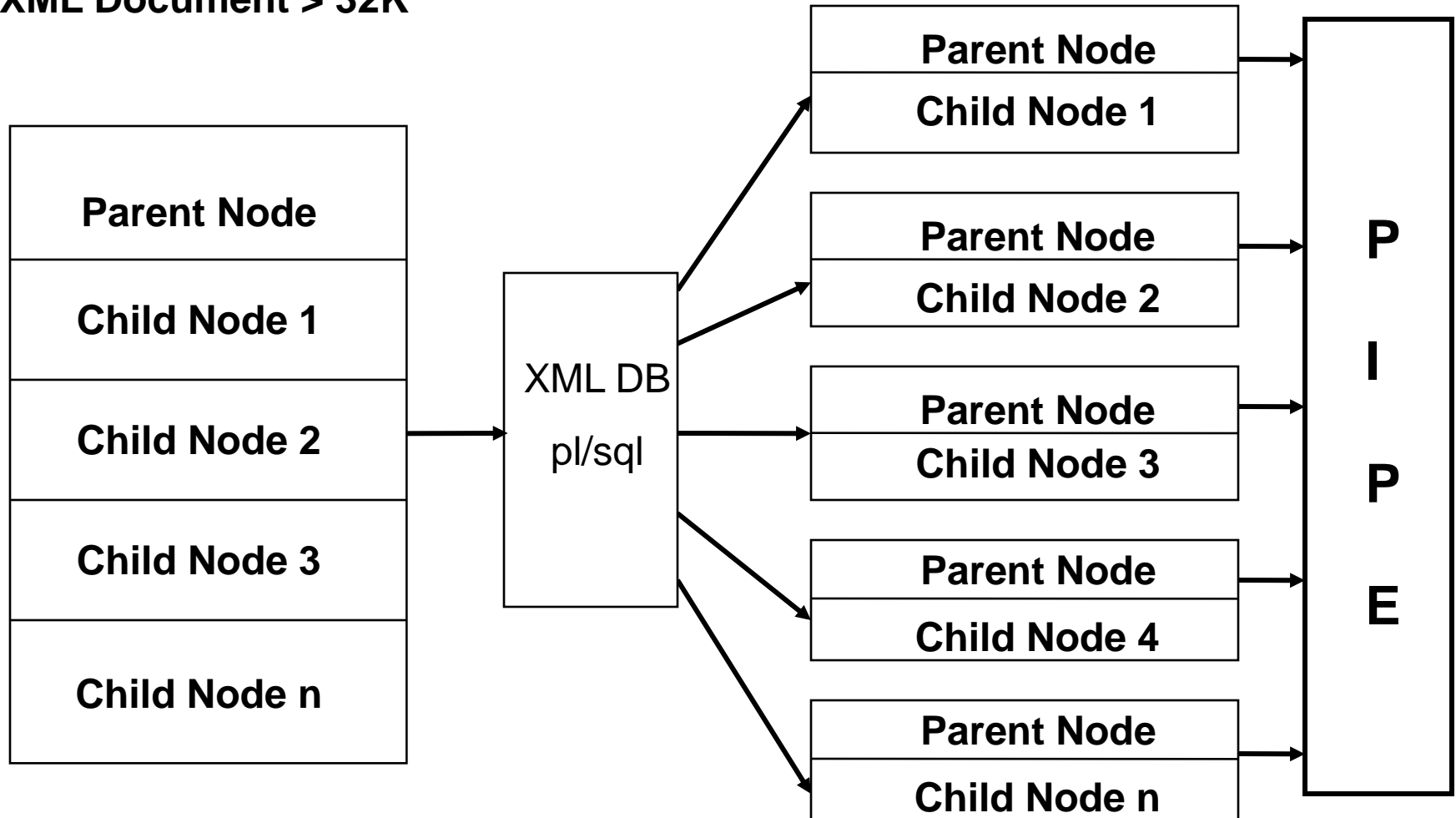
--- **reconstructed**

```
CREATE TABLE MY_RELAT_TABLE_XML_PARSED  
( NAME VARCHAR2(20 BYTE),  
  ITEM VARCHAR2(20 BYTE));
```


XML Packet Creation

Each XML Document < 32 K

XML Document > 32K



XML Packet - Multiple Child Nodes

```
<StandardHeader>
  <DateTime>mm/dd/yyyy hh:mi:ss</DateTime>
  <PacketNumber>1</PacketNumber>
  <AnyMorePackets>Y</AnyMorePackets>
  <TotalPackets>4</TotalPackets>
  <house>
    <room>                                     ←----- child 1
      <name>kitchen</name>
      <item>sink</item>
    </room>
    <room>                                     ←----- child 2
      <name>kitchen</name>
      <item>table</item>
    </room>
    <room>                                     ←----- child 3
      <name>kitchen</name>
      <item>counter</item>
    </room>
  </house>
</StandardHeader>
```

Producer - XML Create Child Node Packet - 1

-- Create Standard Header with root node "house"

```
<StandardHeader>
  <DateTime>mm/dd/yyyy hh:mi:ss</DateTime>
  <PacketNumber>1</PacketNumber>
  <AnyMorePackets>Y</AnyMorePackets>
  <TotalPackets>4</TotalPackets>
  <house>
    <room>
      <name>kitchen</name>
      <item>counter</item>
    </room>
    <room>
      <name>kitchen</name>
      <item>microwave</item>
    </room>
  </house>
</StandardHeader>
```

Producer - XML Create Child Node Packet - 2

-- Create Standard Header with root node "house"

```
<StandardHeader>
  <DateTime>mm/dd/yyyy hh:mi:ss</DateTime>
  <PacketNumber>2</PacketNumber>
  <AnyMorePackets>Y</AnyMorePackets>
  <TotalPackets>4</TotalPackets>
  <house>
    <room>
      <name>living_room</name>
      <item>pictures</item>
    </room>
    <room>
      <name>living_room</name>
      <item>chair</item>
    </room>
  </house>
</StandardHeader>
```

Producer - XML Create Child Node Packet - 3

-- Create Standard Header with root node "house"

```
<StandardHeader>
  <DateTime>mm/dd/yyyy hh:mi:ss</DateTime>
  <PacketNumber>3</PacketNumber>
  <AnyMorePackets>Y</AnyMorePackets>
  <TotalPackets>4</TotalPackets>
  <house>
    <room>
      <name>dining_room</name>
      <item>étagère</item>
    </room>
    <room>
      <name>dining_room</name>
      <item>chair</item>
    </room>
  </house>
</StandardHeader>
```

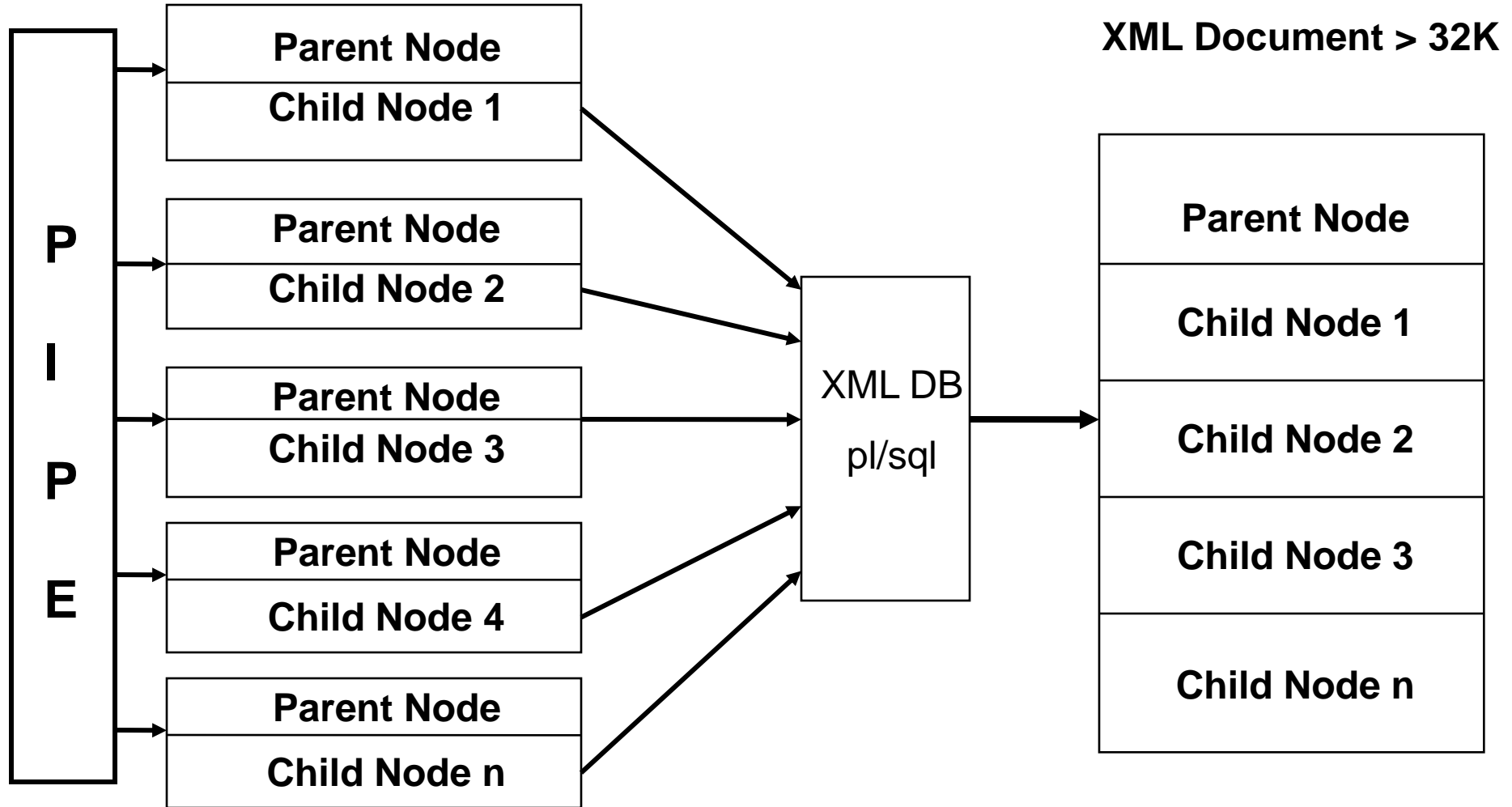
Producer - XML Create Child Node Packet - 4

-- Create Standard Header with root node "house"

```
<StandardHeader>
  <DateTime>mm/dd/yyyy hh:mi:ss</DateTime>
  <PacketNumber>4</PacketNumber>
  <AnyMorePackets>N</AnyMorePackets>
  <TotalPackets>4</TotalPackets>
  <house>
    <room>
      <name>den</name>
      <item>fireplace</item>
    </room>
    <room>
      <name>den</name>
      <item>table</item>
    </room>
  </house>
</StandardHeader>
```

XML Document Re-Construction

Each XML Document < 32 K



XML Document

```
<?xml version="1.0" encoding="WINDOWS-1252"?>
<house>
  <room>
    <name>kitchen</name>
    <item>sink</item>
  </room>
  <room>
    <name>kitchen</name>
    <item>table</item>
  </room>
  <room>
    <name>kitchen</name>
    <item>counter</item>
  </room>
  <room>
    <name>kitchen</name>
    <item>microwave</item>
  </room>
  etc...
</house>
```


Restored Table Data

Table Name: **my_relat_table_xml_parsed**

| NAME | ITEM |
|-------------|--------------|
| kitchen | sink |
| kitchen | oven |
| kitchen | refrigerator |
| living_room | pictures |
| dining_room | table |
| dining_room | chair |
| dining_room | chandelier |
| den | chair |
| den | lamp |
| den | étagère |

Q.E.D.

(demo)

XML (CLOB) Packet Creation

CLOB < 32 K

XML Document > 32K

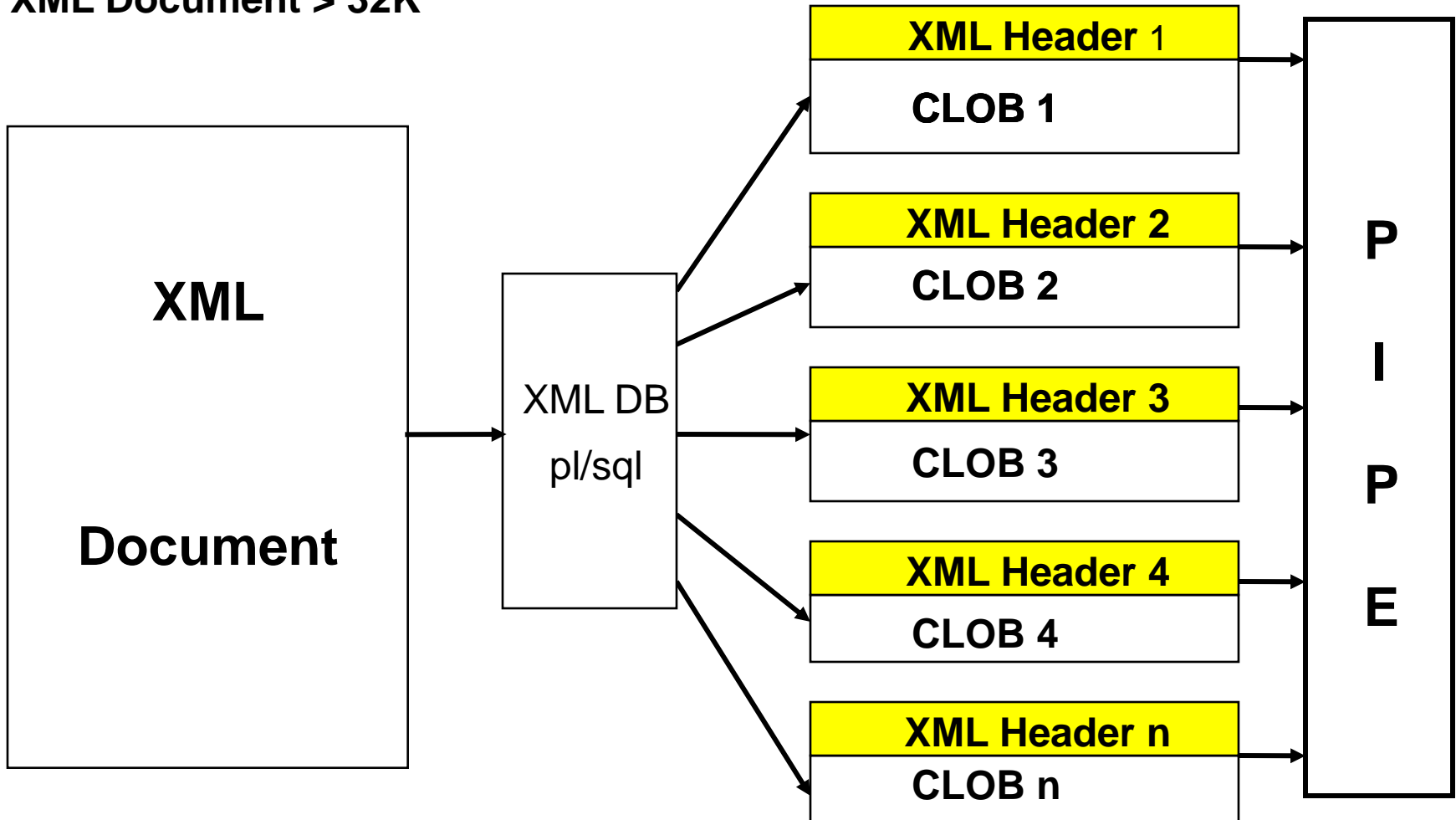


Table Data

Table Name: **my_relat_table**

| NAME | ITEM |
|-------------|--------------|
| kitchen | sink |
| kitchen | oven |
| kitchen | refrigerator |
| living_room | pictures |
| dining_room | table |
| dining_room | chair |
| dining_room | chandelier |
| den | chair |
| den | lamp |
| den | étagère |

XML Document

```
<?xml version="1.0" encoding="WINDOWS-1252"?>
<house>
  <room>
    <name>kitchen</name>
    <item>sink</item>
  </room>
  <room>
    <name>kitchen</name>
    <item>table</item>
  </room>
  <room>
    <name>kitchen</name>
    <item>counter</item>
  </room>
  <room>
    <name>kitchen</name>
    <item>microwave</item>
  </room>
  etc...
</house>
```

CLOB Packet - 1

<StandardHeader>

<DateTime>mm/dd/yyyy hh:mi:ss</DateTime>

<PacketNumber>1</PacketNumber>

<AnyMorePackets>Y</AnyMorePackets>

<TotalPackets>4</TotalPackets>

</StandardHeader>

```
<?xml version="1.0" encoding="UTF-8"?><house><room><name>kitchen</name><item>sink</item></room><room><name>kitchen</name><item>table</item></room><room><name>kitchen</name><item>counter</item></room><room><name>kitchen</name><item>microwa  
ve</item></room><room><name>kitchen</name><item>ove  
n  
</item></r*
```

***cutoff point**

CLOB Packet - 2

<StandardHeader>

<DateTime>mm/dd/yyyy hh:mi:ss</DateTime>

<PacketNumber>2</PacketNumber>

<AnyMorePackets>Y</AnyMorePackets>

<TotalPackets>4</TotalPackets>

</StandardHeader>

```
oom><room><name>kitchen</name><item>range</item>
</room><room><name>kitchen</name><item>refrige
rator</item></room><room><name>living_room</na
me><item>pictures</item></room><room><name>liv
ing_room</name><item>chair</item></room><room>
<name>living_room</name><item>television</item
></room><room><name>li*
```

***cutoff point**

CLOB Packet - 3

<StandardHeader>

<DateTime>mm/dd/yyyy hh:mi:ss</DateTime>

<PacketNumber>3</PacketNumber>

<AnyMorePackets>Y</AnyMorePackets>

<TotalPackets>4</TotalPackets>

</StandardHeader>

```
ving_room</name><item>couch</item></room><room><
name>dining_room</name><item>table</item></roo
m><room><name>dining_room</name><item>etagere<
/item></room><room><name>dining_room</name><it
em>chair</item></room><room><name>dining_room<
/name><item>chandelier</item></room><room><nam
e>den</name><item>fire*
```

***cutoff point**

CLOB Packet - 4

<StandardHeader>

<DateTime>mm/dd/yyyy hh:mi:ss</DateTime>

<PacketNumber>4</PacketNumber>

<AnyMorePackets>N</AnyMorePackets>

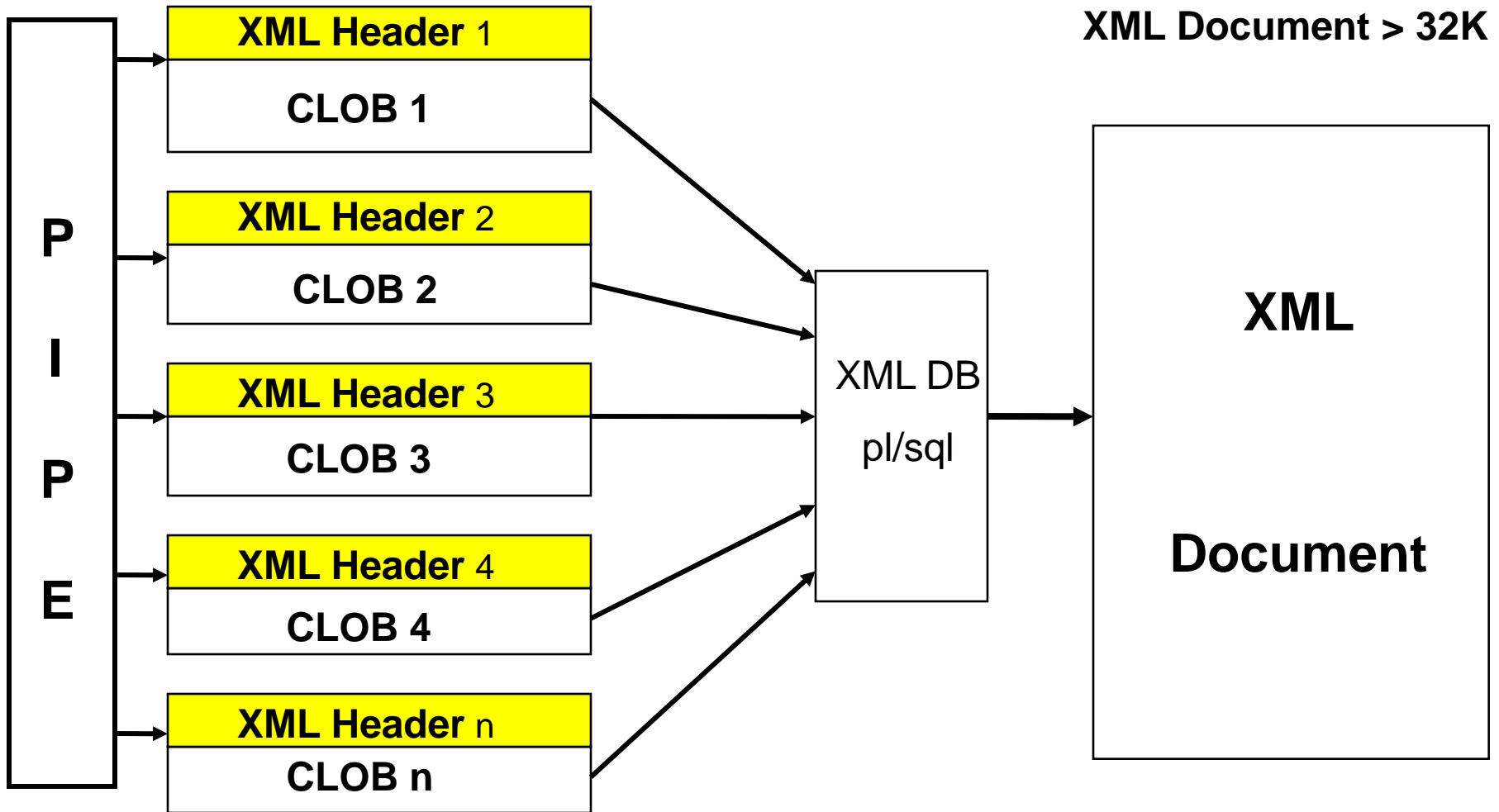
<TotalPackets>4</TotalPackets>

</StandardHeader>

place</item></room><room><name>den</name><item>table</item></room><room><name>den</name><item>chair</item></room><room><name>den</name><item>lamp</item></room><room><name>den</name><item>étagère</item></room></house>

CLOB (XML) Document Construction 2 of 2

(CLOB + XML Header) < 32 K



XML Document - Reconstructed

```
<?xml version="1.0" encoding="WINDOWS-1252"?>
<house>
  <room>
    <name>kitchen</name>
    <item>sink</item>
  </room>
  <room>
    <name>kitchen</name>
    <item>table</item>
  </room>
  <room>
    <name>kitchen</name>
    <item>counter</item>
  </room>
  <room>
    <name>kitchen</name>
    <item>microwave</item>
  </room>
  etc...
</house>
```

Restored Table Data

Table Name: my_relat_table_xml_parsed

| NAME | ITEM |
|-------------|--------------|
| kitchen | sink |
| kitchen | oven |
| kitchen | refrigerator |
| living_room | pictures |
| dining_room | table |
| dining_room | chair |
| dining_room | chandelier |
| den | chair |
| den | lamp |
| den | étagère |

Q.E.D.

(demo)

Technique Comparison - Summary

- XML Child Node Packets - multiple child relationship to one parent.
- May also apply to many children belonging to many parents.
- Complications - “bookkeeping” - accordion like effect
- Data Structures - agreement with both sides

- XML header prefixed to a CLOB packet
- The entire XML document is divided into arbitrary sections or CLOBs.
- The receiver must know the size of the XML header prior to any data transfer
- The XML header is segregated from the payload
- With all the CLOB pieces recovered, reconstruct the original XML document.

References

- ¹ Wikipedia, http://en.wikipedia.org/wiki/Moore's_law
- ² XML DB, http://download-west.oracle.com/docs/cd/B28359_01/appdev.111/b28369/toc.htm
- ³ Wikipedia, <http://en.wikipedia.org/wiki/Prettyprint>
- ⁴ Oracle dbms_lob, http://download.oracle.com/docs/cd/B28359_01/appdev.111/b28419/d_lob.htm#i1016062
- ⁵ Wikipedia, <http://en.wikipedia.org/wiki/Metadata>
- ⁶ Wikipedia, <http://en.wikipedia.org/wiki/Q.E.D.>



Questions

