

XEON: an Enterprise Middleware for the Internet

Luca Bompani

Paolo Ciancarini

Fabio Vitali

Dept. Of Computer Science
University of Bologna

Contents

- Introduction
- Other Existing Middleware
- SOAP and DOM
- XMLC
- XEON
- Conclusions

Introduction

- Internet contains a lot of useful information, but difficult to retrieve.
- Data are accessible with of a lot of different protocols.
- The introduction of XML has increased the cohesion of the internet.
- XEON is an XML based middleware designed to provide an uniform interface to all the internet resources, both data and services.

Other Middleware

- CORBA: OMG open architecture
- COM: Microsoft proprietary architecture
- RMI: Java RPC architecture

All this middleware are Object-Oriented

SOAP (1)

- Simple Object Access Protocol
- SOAP is a lightweight protocol for exchange of information in a decentralized, distributed environment.
- It is an XML based protocol.
- It defines a set of encoding rules for expressing instances of application-defined data types, and a convention for representing remote procedure calls and responses.

SOAP (2)

POST /StockQuote HTTP/1.1
Host: www.stockquoteserver.com
Content-Type: text/xml;
charset="utf-8"
Content-Length: nnnn
SOAPAction: "Some-URI"

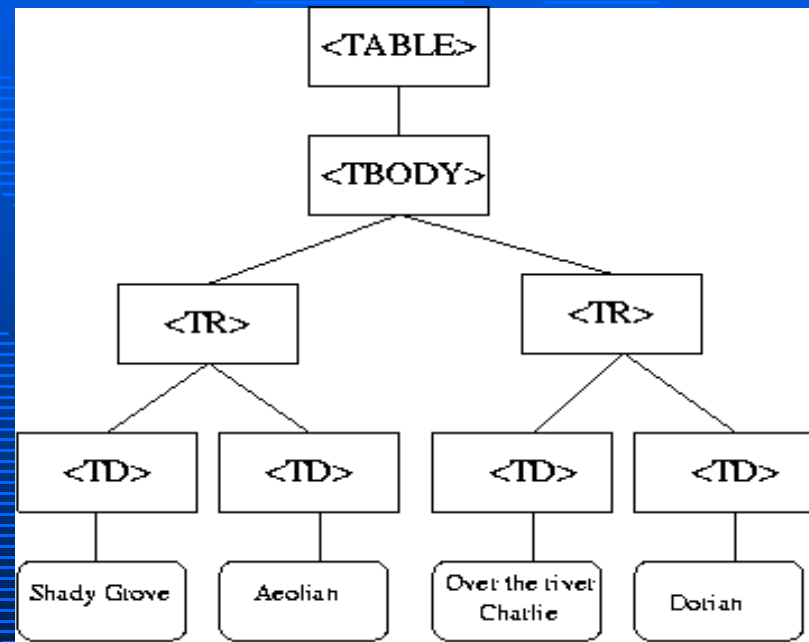
```
<SOAP-ENV:Envelope  
  <SOAP-ENV:Body>  
    <m:GetLastTradePrice  
      xmlns:m="Some-URI">  
      <symbol>DIS</symbol>  
    </m:GetLastTradePrice>  
  </SOAP-ENV:Body>  
</SOAP-ENV:Envelope>
```

DOM (1)

- Document Object Model
- Is a platform- and language-neutral interface that allows programs and scripts to dynamically access and update the content, structure and style of documents.
- Provides a standard set of objects for representing XML documents, a standard model of how these objects can be combined, and a standard interface for accessing and manipulating them.

DOM (2)

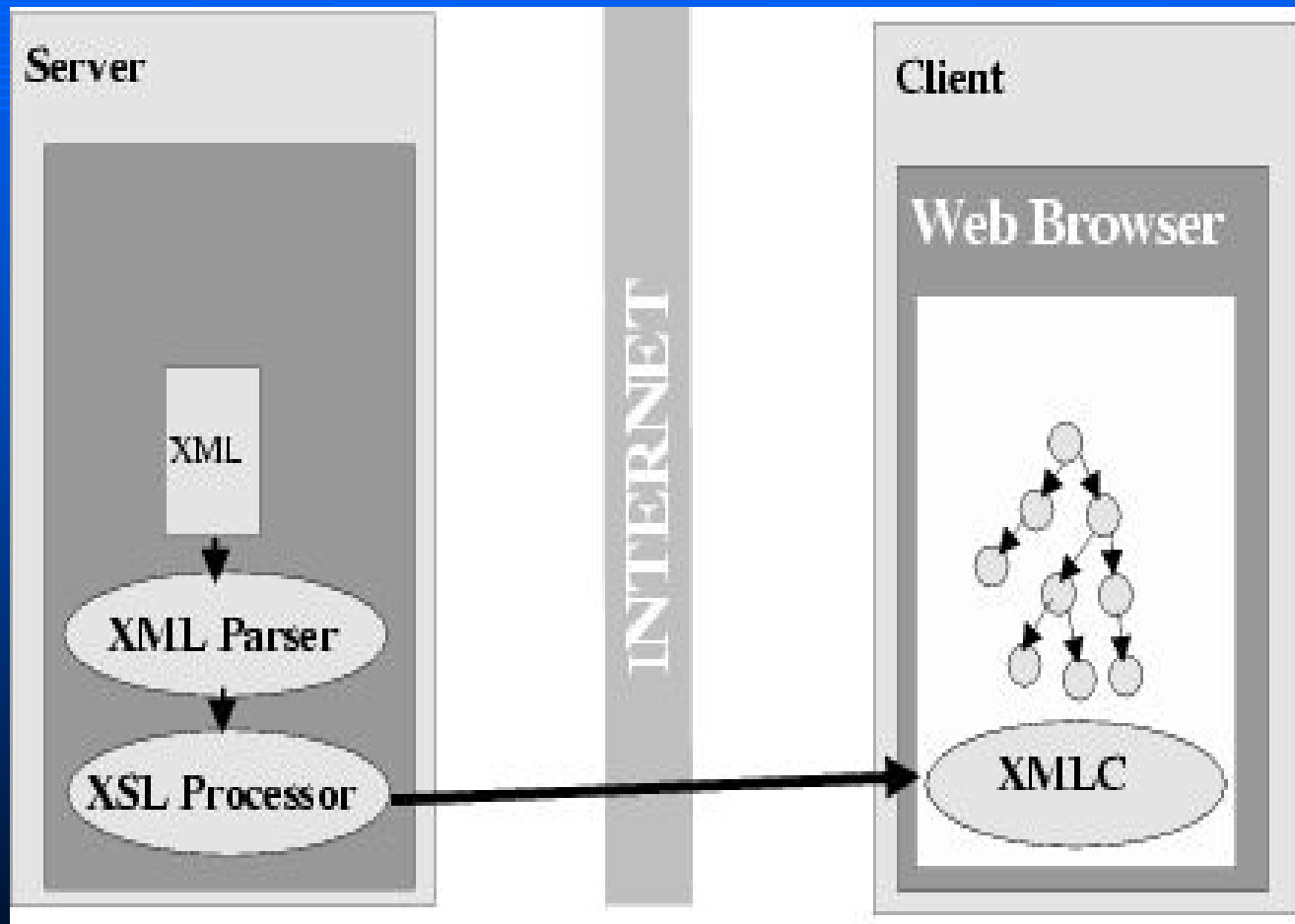
```
<TABLE>
<TBODY>
<TR>
  <TD>Shady Grove</TD>
  <TD>Aeolian</TD>
</TR>
<TR>
  <TD>Over the River,
Charlie</TD>
  <TD>Dorian</TD>
</TR>
</TBODY>
</TABLE>
```



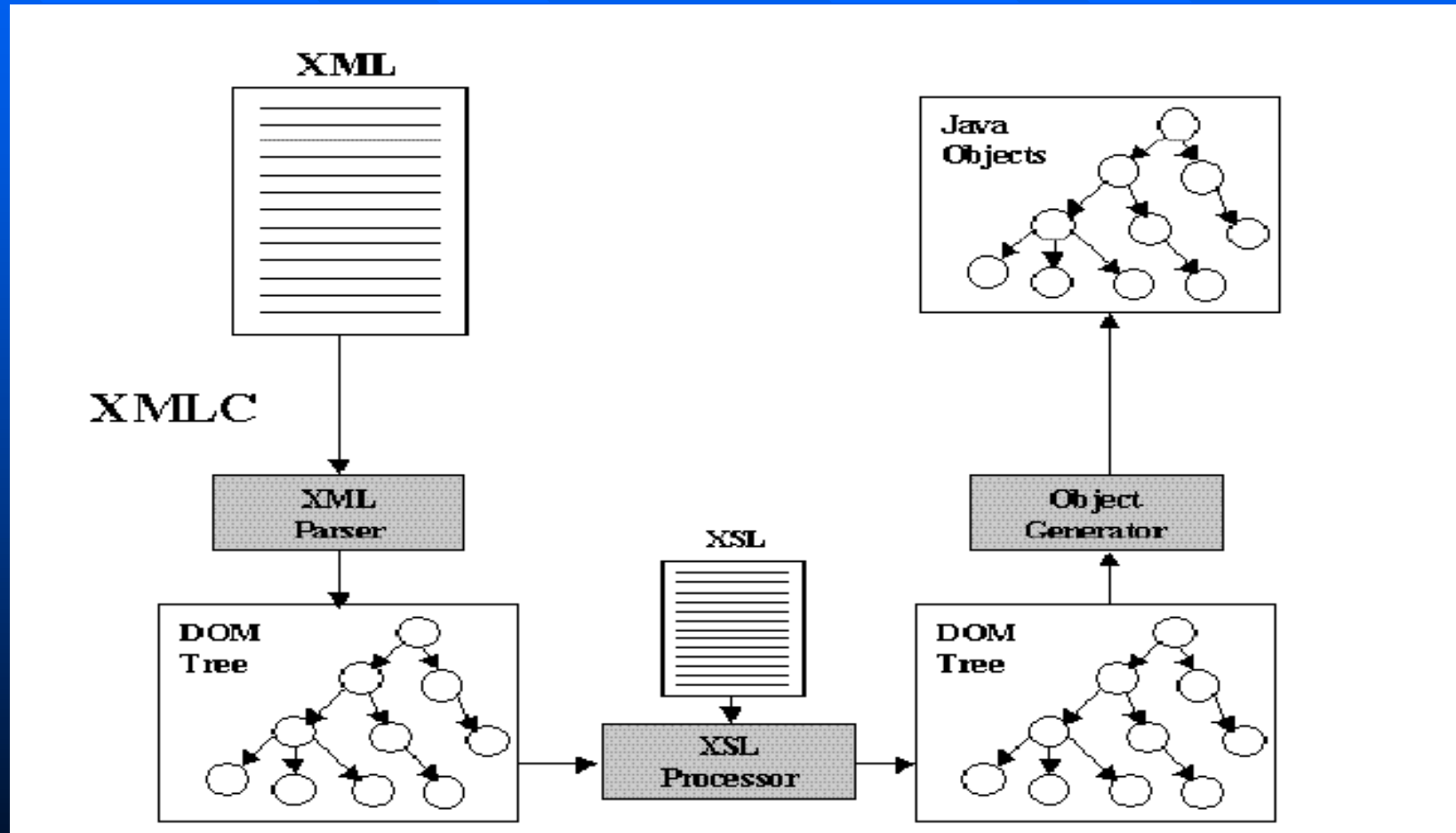
XMLC (1)

- XMLC is an engine that build a tree of JavaBeans from an XML document and then provide access to the same tree throw a DOM interface.
- It was projected as part of the Displet project, to provide custom visualization of XML documents, and in particular of Software Engineering special notations.

XMLC (2)



XMLC (3)



XMLC (4)

```
r(x,y) == x*y  
  
SYNTAX: re: normal isTwice type name: exp{+1,y+2,z+3} normal  
  
_isTwice_ N..N  
forall (j N) isTwice (count = 2)  
  
TREE := tp | fork(N-TREE-TREE)  
  
x = y..z  
  
{NAME, PHONE}  
PhoneDB{K,L,Z}  
_known: P NAME  
phone: NAME .. PHONE  
known = dom phone  
  
{X,Y}  
test: X x Y -> X  
forall x: X, y: Y • first(x, y) = x
```

XMLC (5)

Applet Viewer: XMLManager.class

Applet

XMLManager mix test

This xml file contains markup elements from three different sets

- StateMate: elements to describe states
- ZIF: the Z Interchange format
- HTML: HTML 1.0 implementation.

Click [here](#) to see the xml source of this document

LEVEL MANAGER CONTROL

```
graph TD
    A["isTwice : N ↔ N  
∇ z, j : N • z isTwice j ↔ z=2*j"]
    B["[X,Y]  
first: X × Y → X  
∇ x : X, y : Y • first(x, y) = x"]
    C["PhoneDB [K,L,Z]  
_ known: P NAME  
_ phone: NAME ↔ PHONE  
known = dom phone"]
    A -- ACTIVE --> B
    A -- MORE --> C
    B -- UPDATE --> C
```

Applet started.

XMLC(6)



XEON (1)

- XEON is a middleware based on XML that provide an uniform access to all the resources and the services on the Internet.
- Instead of define a completely new set of API, XEON extends the interfaces of the XML Document Object Model.

XEON (2)

XEON extends the DOM defining some new interface to handle method invocations, node locks and access control:

- Invokable
- Lockable
- Protectable

XEON (3)

The invokable interface handles method invocations:

```
interface invokable {  
    readonly NamedNodeMap methods;  
    Element getMethod(in DOMString name)  
    Element invoke(in DOMString name,  
                  in NamedNodeMap params)  
    raises(DOMException);  
}
```

XEON (4)

The lockable interface handles node locking:

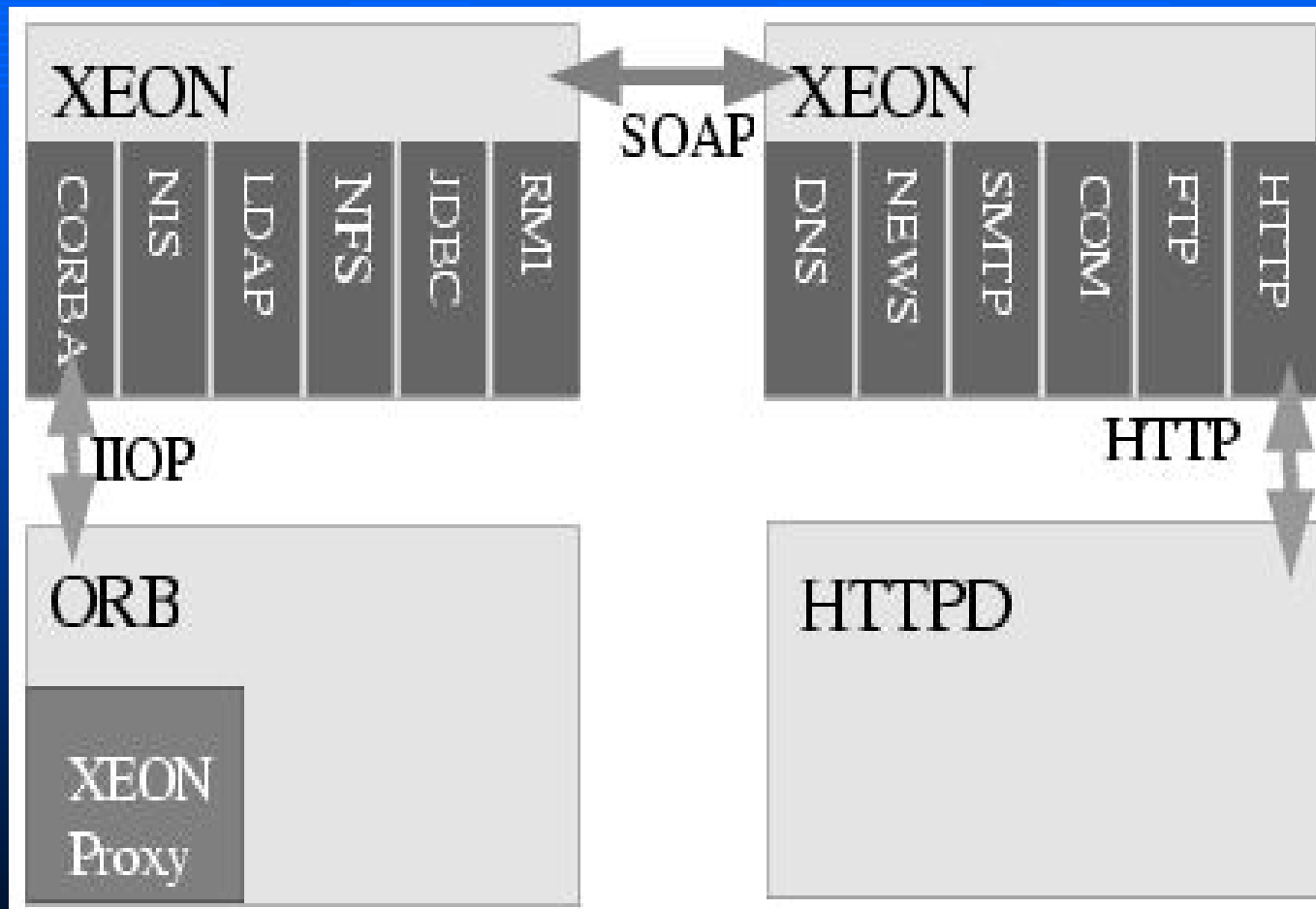
```
Interface lockable{  
    readonly boolean locked;  
    void lock(in int timeout) raises  
        (DOMException);  
}
```

XEON (5)

The Protectable interface handle the access control of users to node:

```
Interface protectable{  
    readonly NamedNodeMap rights;  
}
```

XEON (6)



Conclusions

- At the present the XEON architecture isn't yet implemented.
- It introduces a new way to see the web.
- It allows users to use standard XML tools to browse all the resources of the Net.
- It provide an uniform environment for mobile agents. They can move around the network interacting with the hosting environment always with the same API.