

CS155b: E-Commerce

Lecture 10: Feb. 13, 2003

XML and its relationship to B2B commerce

Acknowledgements: R. Glushko, A. Gregory,
and V. Ramachandran

Let's Discuss Integration

I am a computer manufacturer. I want to create a website through which people can custom-build and purchase computers.

- What supplies are needed?
- How do I coordinate the production of parts?
- How do I know that I have enough parts to fulfill website orders? Can I track order progress?
- How can I analyze the requests to order the materials I need?
- Can I automate this process?
- Can I use the Internet to improve efficiency?

Another Example

I run a financial services company that provides an online brokerage where customers can place orders and evaluate their portfolios.

- What information will my customers want to access through my website?
- What are the sources of this information?
- How and when does this information change?
- Can I automate information updates?

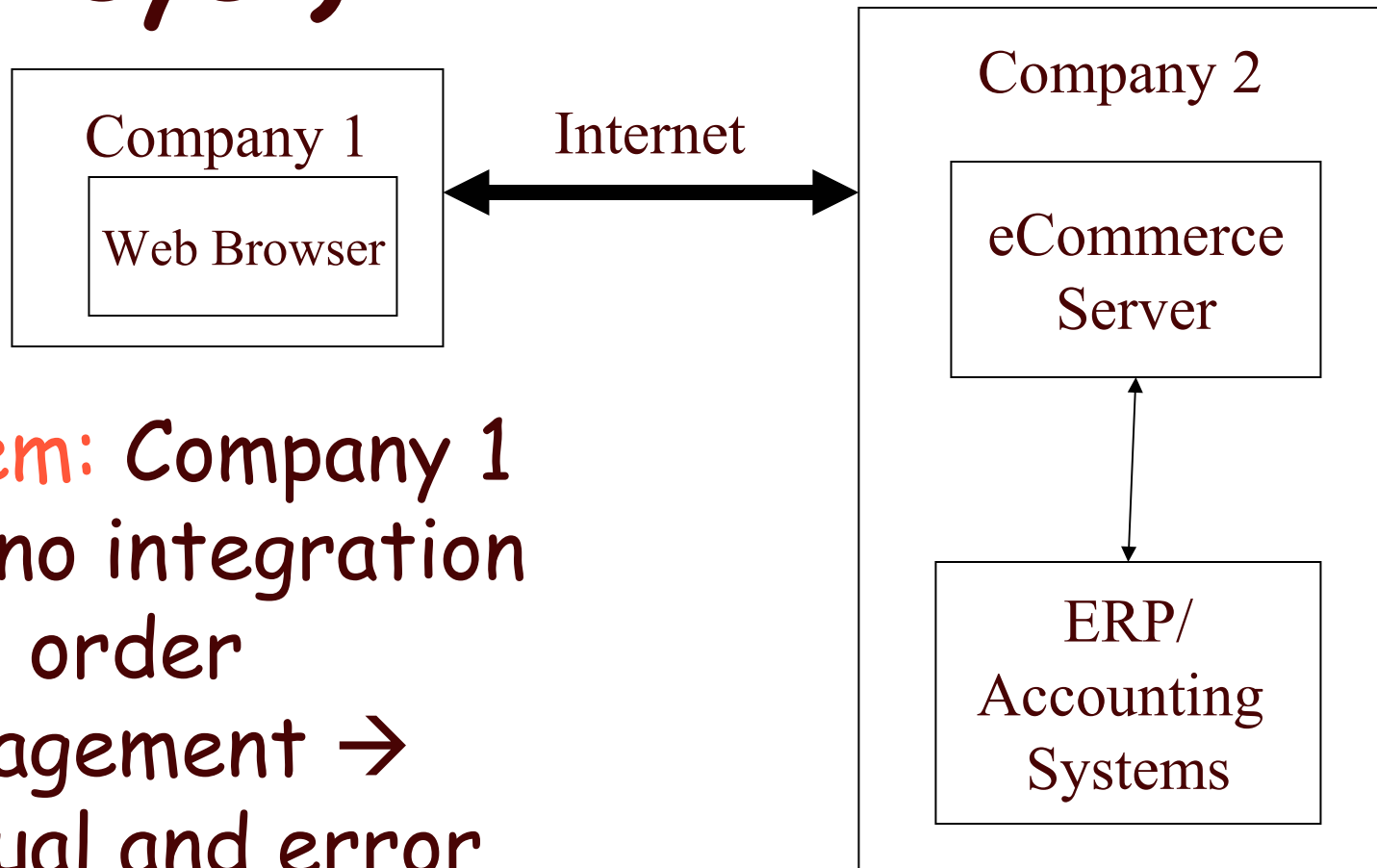
Remove the Human?

- We can use computers to analyze incoming orders and requests for information.
- We can use computers to order supplies online and search for vendors.
- We can use computers to search through databases of information and automatically format results for display.
- All of the above can be done through the standard WWW interface, **but separately**. Can computers automatically connect these operations?

HTML's Limitations for Integration

- The Web was created as a **publishing medium**, not as an **e-commerce platform**
- HTML, the Web's language for encoding information, is **format-oriented** and meant to be understood "by eye"
 - Simple structures: headings, lists, links
 - Browsers are "hard wired" to render HTML as web pages
- **No content-based encoding** means that HTML can't be effectively searched or processed by business applications

Connecting with HTML ("by eye")



Problem: Company 1
has no integration
with order
management →
manual and error
prone data entry

The XML Revolution

- The Web was created to **publish information** for people.
 - "Eyes-only" was dominant design perspective
 - Hard to search
 - Hard to automate processing
- The Web is using XML to become a platform for **information exchange between computers (and people)**.
 - Overcomes HTML's inherent limitations
 - Enables the new business models of the network economy

Extensible Markup Language

- Instead of a **fixed set of format-oriented tags** like HTML, XML allows you to **create whatever set of tags are needed** for your type of information.
- This makes any XML instance **"self-describing"** and easily understood by computers and people.
- XML-encoded information is **smart enough to support new classes** of Web and e-commerce applications.

Why XML?

Sample Catalog Entry in HTML

```
<TITLE> Laptop Computer </TITLE>  
<BODY>  
<UL>  
<LI> IBM Thinkpad 600E  
<LI> 400 MHz  
<LI> 64 Mb  
<LI> 8 Gb  
<LI> 4.1 pounds  
<LI> $3200  
</UL></BODY>
```

XML's Big Idea: Document Types

- Customer Profiles
- Vendor Profiles
- Catalogs
- Datasheets
- Price Lists
- Purchase Orders
- Invoices
- Inventory Reports
- Bill of Materials
- Payments
- Deposits
- Credit Reports
- Schedules
- Directories
- ...*whatever you need*

In XML the formal definition of permitted elements, attributes, and the rules by which they combine is called a **Document Type Definition** or **DTD** or **schema**.

Catalog Entry in XML

```
<COMPUTER TYPE="Laptop">  
  <MANUFACTURER>IBM</MANUFACTURER>  
  <LINE> ThinkPad</LINE>  
  <MODEL>600E</MODEL>  
  <SPECIFICATIONS>  
    <SPEED UNIT = "MHz">400</SPEED>  
    <MEMORY UNIT="MB">64</MEMORY>  
    <DISK UNIT="GB">8</DISK>  
    <WEIGHT UNIT="POUND">4.1</WEIGHT>  
    <PRICE CURRENCY="USD">3200</PRICE>  
  </SPECIFICATIONS>  
</COMPUTER>
```

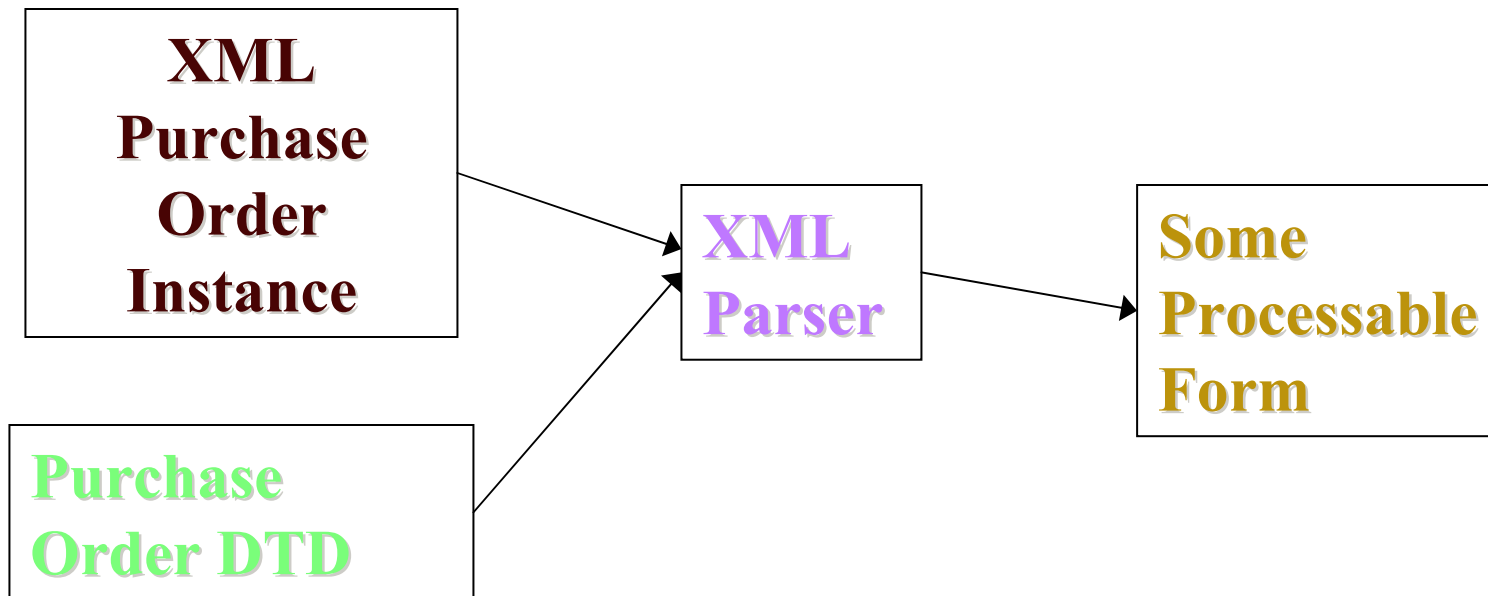
Smart Processing with XML

- `<COMPUTER>` and `<SPECIFICATIONS>` provide logical containers for extracting and manipulating product information as a unit
 - Sort by `<MANUFACTURER>`, `<SPEED>`, `<WEIGHT>`, `<PRICE>`, etc.
- Explicit identification of each part enables its automated processing
 - Convert `<PRICE>` from "USD" to Euro, Yen, etc.

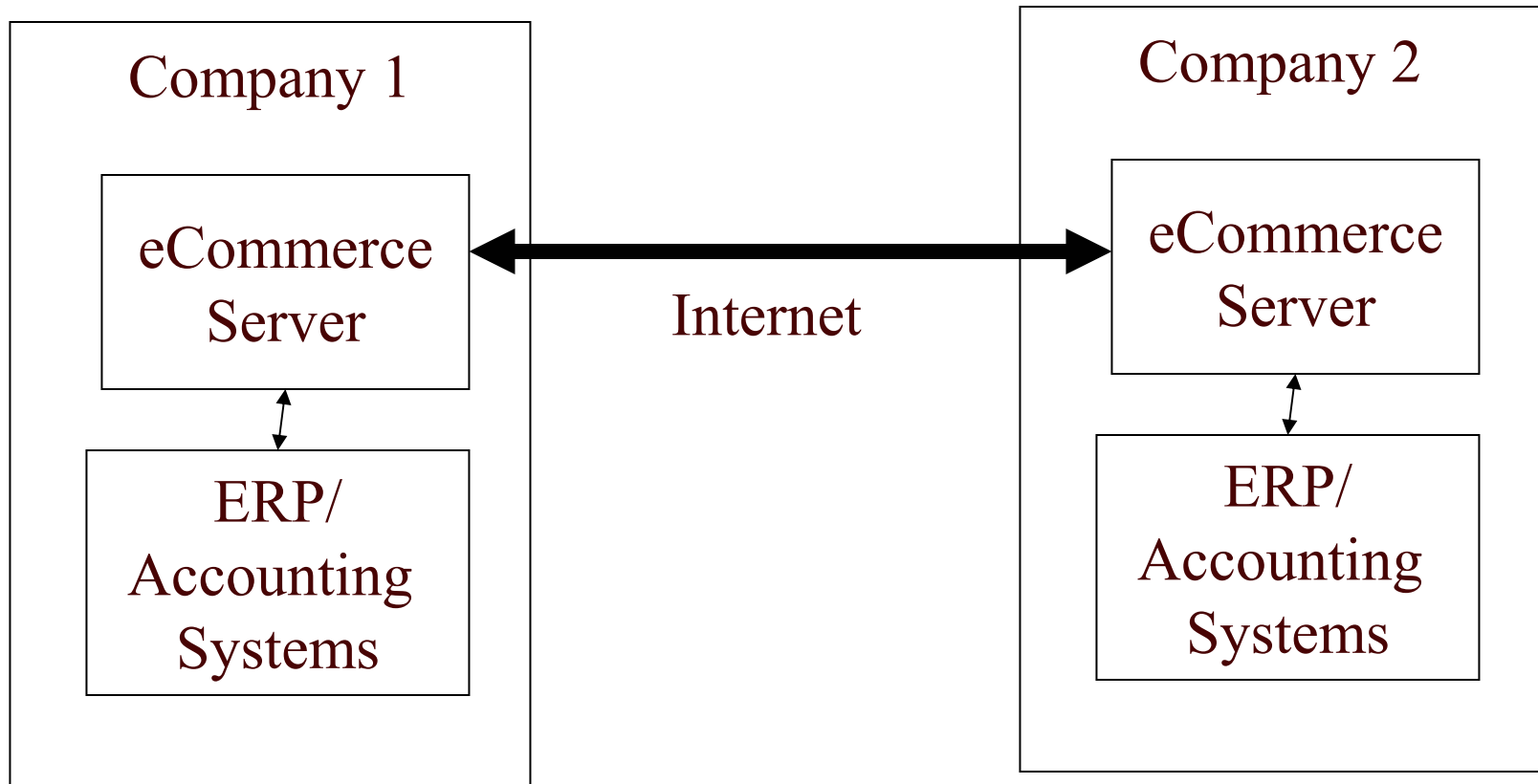
DTDs, Parsers, and Validation

- From any **DTD (document type definition)**, an **XML parser** can be generated that:
 - reads a document instance (the XML data stream);
 - identifies the markup in it; and
 - creates a **processable form** of some kind that is used by an application.
- The parser can also test the XML document for conformance with the rules of the DTD.
 - A document instance that follows the rules of the DTD is "valid."

DTDs And Validation



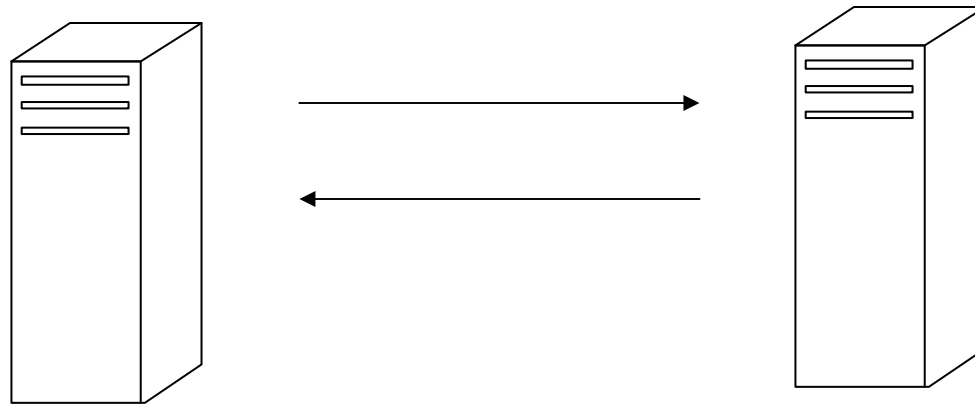
Connecting using XML



Benefit: XML can be processed automatically with huge cost savings

Problem: Company 1 and Company 2 have to agree on document format

Business Processes are XML Document Exchanges



If you send me a **request** for a **catalog**, I will send you a **catalog**

If you send me a **purchase order** and I can fill it, I will send you a **purchase order response**

Significance of XML Document Exchange Architecture

- **Document exchange** is a natural way to think about doing business
- Easy to provide **"open" marketplace** with 3rd party buying and selling apps
- Easy to add and maintain services
- Document exchange between marketplaces is fundamentally the same as within a marketplace
- Services can be reused across marketplaces

XML is Part of the Solution

- XML has the potential to enable a standards-conforming, open and extensible architecture for electronic commerce.
- XML standards could enable ubiquitous connectivity and interoperability and create the network effects of "describe once, {sell, buy} anywhere" and reusable marketplace services.

Homework for Feb. 18, 2003

- Reminder: Second written homework assignment must be submitted online by 5pm Tuesday.
 - Use Yale Classes server
 - Instructions are on course web page and on the first page of the homework.
- Prepare for Garfinkel's invited lecture on Tuesday: read Chapters 1 and 2 of *Database Nation* (print form only).