Why XML (eXtensible Markup Language)

• To allow the definition and use of many application-specific markup languages
• Slogan: XML does for data what Java does for programs: platform-independent and vendor-independent format
• www.w3.org/TR
Why markup languages?

• Contain plenty of concrete syntax. Easy to parse and human readable.
• Address = “<address>” … “</address>”.
• Graph = “<graph>” … “</graph>”.
Connections

• Languages
  – grammar for grammars
  – grammar for language
  – sentence of language

• Markup Languages
  – XML: grammar for grammars of markup langs
  – grammar for specialized markup language
  – document
Applications of XML

- MathML
- XMI? Define interchange format for UML objects
- Textual description of objects
- Textual description of multimedia: SMIL Synchronized Multimedia Integration Language
Simplified example using proposed XML/Document Content Description (Microsoft/IBM)

<DCD>
  <ElementDef Type="Booking">
    <Description> Airline res. </Description>
    <Group Order="Seq">
      <Element> LastName </Element>
      <Element> FirstInitial </Element> ...
      <Element> SeatRow </Element>
    </Group>
  </ElementDef>
</DCD>
Simplified example using proposed XML/Document Content Description (Microsoft/IBM)

```xml
<ElementDef Type="SeatRow" Model="Data"
    DataType="int" Min="1" Max="72" />
</DCD>
```

Example Booking object:

```xml
<Booking>
    <LastName> Finkelstein </LastName>
    <FirstInitial> L </FirstInitial>
    <SeatRow> 33 </SeatRow> ...
</Booking>
```
Very Simplified XML

CanonicalXML = List(Pi) Element List(Pi).
Element = Stag Inner Etag.
Inner : DataChar | Pi | Element.
Stag = “<“ Name Atts “>”.
Etag = “</” Name “>”.
Pi = “<?” ... “?>”.
Atts = List(Att).
Att = Name = “””” ... “”””.
List(S) ~ {S}. 