

**Birkbeck  
(University of London)**

***MSc in Advanced Information Systems  
MRes in Computer Science  
MSc in E-Commerce***

**School of Computer Science and Information Systems**

**Development of Internet Applications—Solutions**

**Monday 24 May 2004 (14:30–16:30)**

1. (a) The names of the 5 built-in entities are amp, lt, gt, quot and apos. In order to refer to an entity, its name is prefixed by & and followed by ;. (6 marks)

- (b) XML schema allows element types to be local to a context, e.g.,

```
<xsd:element name="book">
  ...
  <xsd:element name="title" type="typeA"/>
  ...
</xsd:element>

<xsd:element name="employee">
  ...
  <xsd:element name="title" type="typeB"/>
  ...
</xsd:element>
```

The type of each occurrence of title is different, depending on the context in which each occurs. The same cannot be achieved using DTDs, since all element names are global in a DTD. (7 marks)

- (c) The following axis includes all elements whose start tags follow the end tag of the context node. The following-sibling axis includes all elements whose start tags follow the end tag of the context node and whose end tags appear before the end tag of the parent of the context node. Thus the following and following-sibling axes overlap. The descendant axis includes all nodes whose start and end tags follow the start tag and precede the end tag of the context node. Thus the descendant axis does not overlap with either of the following or following-sibling axes. (6 marks)
- (d) Servers use a combination of filename suffixes/extensions and URL prefixes to decide which processing mechanism to use. The default is that the URL is processed as a request for static content. A URL path beginning with /servlet/ might indicate that the target is a Java servlet, while one beginning with /cgi-bin/ might indicate that the target is a CGI script. A URL path where the target filename ends in .cgi might indicate that the target is a CGI script, while one ending in .php or .cfm might indicate template processing with PHP or Cold Fusion. (6 marks)

2. (a) Namespaces were defined after DTDs, so are not supported by DTDs. One can declare an element name such as `person:title` in a DTD, but it is just a 12 character name; the prefix has no significance. Once such names have been defined, one cannot default them or change the prefix. In the following DTD fragment

```
<!ELEMENT title ...>
<!ATTLIST title
  xmlns          CDATA #FIXED "http://www.person.com">
<!ELEMENT person:title ...>
<!ATTLIST person:title
  xmlns:person  CDATA #FIXED "http://www.person.com">
```

`title` and `person:title` are different element names in a valid document but they are the same according to namespaces specification. (10 marks)

- (b) With respect to a context node `c`, the expression returns those child CD elements of `c` that have a `publisher` attribute value equal to `Deutsche Grammophon` and that have more than 2 performance elements as child nodes. (7 marks)
- (c) The code retrieves information from a database table called `staff` that is on the server `eros`. One column (attribute) of `staff` is `username`. The `QueryString` is that part of the URL that follows the `?` in the URL. The value of the `name` parameter from the `QueryString` (i.e. `ptw`) is substituted into the SQL query so that the query becomes `select * from staff where username = 'ptw'`. (8 marks)
3. (a) i. The operator `+` means that one or more occurrences of the content particle that precedes it are required. The operator `?` means that the content particle that precedes it is optional. (2 marks)
- ii. It would return the value `normal`. This is because the attribute `type` takes on its default value when no value is specified in the original document. (3 marks)
- iii. There should be at least one to element after the `from` element. The contents of the `paragraph` element is not allowed to contain any nested elements since it is of type `PCDATA`. (4 marks)
- (b) The full XPath syntax is
- (i) `/self::node()/`,
  - (ii) `/parent::node()/` and
  - (iii) `/descendant-or-self::node()/`. (6 marks)
- (c) Cache is storage used for temporarily storing responses to requests. Its purpose is to (i) reduce response time, (ii) reduce network load, or (iii) save money by avoiding access to Internet. Caching differs from replication (mirroring) in that replication is the explicit duplication of pages, usually at a different site and usually under explicit human control. Replication does not occur as a result of an HTTP request and each replica can potentially be updated independently. (10 marks)

4. (a) `<xsd:complexType name="fileType">`  
`<xsd:sequence>`  
`<xsd:element name="name" type="xsd:string"/>`  
`<xsd:element name="last-modified" type="xsd:dateTime"/>`  
`<xsd:element name="owner-permissions">`  
`<xsd:sequence>`  
`<xsd:element name="permission" type="permissionType"`  
`minOccurs="0" maxOccurs="3"/>`  
`</xsd:sequence>`  
`</xsd:element>`  
`<xsd:element name="user-permissions">`  
`<xsd:sequence>`  
`<xsd:element name="permission" type="permissionType"`  
`minOccurs="0" maxOccurs="3"/>`  
`</xsd:sequence>`  
`</xsd:element>`  
`</xsd:sequence>`  
`</xsd:complexType>`

`<xsd:simpleType name="permissionType">`  
`<xsd:restriction base="xsd:string">`  
`<xsd:enumeration value="read"/>`  
`<xsd:enumeration value="write"/>`  
`<xsd:enumeration value="execute"/>`  
`</xsd:restriction>`  
`</xsd:simpleType>`

(14 marks)

(b) `<file>`  
`<name>myfile.txt</name>`  
`<last-modified>1988-11-11T11:00:00Z</last-modified>`  
`<owner-permissions>`  
`<permission>read</permission>`  
`<permission>write</permission>`  
`<permission>execute</permission>`  
`</owner-permissions>`  
`<user-permissions>`  
`<permission>read</permission>`  
`</user-permissions>`  
`</file>`

(6 marks)

(c) Servers retain no information about past requests.

(2 marks)

(d) GET: request the document named by the request-URI. HEAD: return only header information of the request-URI (e.g., test for validity, recent modification). POST: submit information to entity given by the request-URI.

(3 marks)

5. (a) The simplest solution is probably the following:

```
<xsl:stylesheet version="1.0"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform">

<xsl:template match="/programme">
  <html><body>
  <h1><xsl:value-of select="degree"/> ( <xsl:value-of select="year"/> )
  </h1>
  <table>
    <xsl:apply-templates select="//name"/>
  </table></body></html>
</xsl:template>

<xsl:template match="name">
  <tr><td><xsl:value-of select="."/></td>
    <td><xsl:value-of select="name(..)"/></td></tr>
</xsl:template>

</xsl:stylesheet>
```

(13 marks)

- (b) Some possibilities are: OnClick, OnDbClick onMouseDown, onMouseUp, onMouseOver, onMouseMove, onMouseOut onKeyPress, onKeyDown, onKeyUp. (4 marks)
- (c) A proxy is an intermediary program acting as both client and server. Requests to a proxy can be serviced internally (e.g. using a cache). A client's request must be explicitly addressed to the proxy. A gateway is a program acting as an intermediary for a server (e.g. a firewall). The client does not know it is not communicating with the origin server. (8 marks)

6. This question may be answered only by MSc AIS and MRes students

- (a) An OEM data graph is a rooted, labelled, directed graph. Its edge labels map to strings and only its leaf nodes have labels; these map to data values. There is no ordering of the edges leaving a node. Some advantages of such a model are that it is easy to add new data, to integrate heterogeneous data and to query the data without knowing the data types. Some disadvantages are that the model loses type information and makes query optimisation harder. (10 marks)
- (b) Let the constraint be that a document (doc element) must contain one or more paragraphs (par elements), and that the first paragraph has a different content model to subsequent paragraphs. This can be modelled using a regular tree grammar as follows:

```
D -> doc (P1, P2*)
P1 -> par (...)
P2 -> par (...)
```

or using RELAX NG as follows:

```
<define name="D">
  <element name="doc">
    <ref name="P1"/>
    <zeroOrMore>
      <ref name="P2"/>
    </zeroOrMore>
  </element>
```

```

    </element>
</define>

<define name="P1">
  <element name="par">
    ...
  </element>
</define>

<define name="P2">
  <element name="par">
    ...
  </element>
</define>

```

(10 marks)

- (c) Given a document or database  $d$ , we denote the result of evaluating query expression  $Q$  on  $d$  by  $Q(d)$ . Then expression  $Q_1$  *contains* expression  $Q_2$  if  $Q_1(d) \supseteq Q_2(d)$  for all  $d$ . Expression  $Q_1$  is *equivalent* to expression  $Q_2$  if  $Q_1$  contains  $Q_2$  and  $Q_2$  contains  $Q_1$ . (5 marks)

7. This question may be answered only by MSc E-Commerce students

- (a) In traditional e-commerce, the goal is to reduce costs through automation and increased efficiency. In e-services the goal is to increase revenue through enhancing service and building profitable customer relationships. The following table shows some contrasting aspects of the two approaches:

e-commerce	e-service
goods	services
tangible products	information products
advertising	two-way dialogue
cost reduction	revenue expansion
efficiency focus	satisfaction focus
supply chain	information flows
product profitability	customer profitability
mass marketing	one-to-one marketing
commodities	customisation
low margins	high margins

(10 marks)

- (b) 

```

<script language = "JavaScript">
  var xmlDoc = new ActiveXObject("Microsoft.XMLDOM");
  xmlDoc.async="false";
  xmlDoc.load("file.xml" );
  var nameElements = xmlDoc.getElementsByTagName("price");
  for ( i = 0; i < nameElements.length; i++ )
    document.write("<p>", nameElements.item(i).firstChild.nodeValue, "</p>");
</script>

```

(15 marks)