

Database and Information Systems

Homework 4

October 7, 2003; Due October 16 at 1:30 PM

Problem 1:

1.

```
<result>
{
  for $ip in document("db-inproc.xml")/dblp/inproceedings,
    $au in $ip/author
  where $au = "Jeffrey D. Ullman"
  return $ip
} </result>
```
2.

```
<result>
{
  for $pr in document("db-proc.xml")/dblp/proceedings[year/text() = "1999"],
    $pt in $pr/title,
    $pk in $pr/@key,

    $ip in document("db-inproc.xml")/dblp/inproceedings,
    $scr in $ip/crossref/text(),
    $it in $ip/title/text()

  where $pk = $scr
  return <proceedings>{$pt}<paper-title>{$it}</paper-title></proceedings>
}
</result>
```
3.

```
<result>
{
  for $pr in fn:document("db-proc.xml")/dblp/proceedings[year/text() = "1999"],
    $pt in $pr/title,
    $pk in $pr/@key

  return <proceedings>
    {$pt}
    {
```

```

for $ip in document("db-inproc.xml")/dblp/inproceedings,
    $scr in $ip/crossref/text(),
    $it in $ip/title/text()

where $pk = $scr
return <paper-title>{$it}</paper-title>
}
</proceedings>
}
</result>

4. <result>
{
for $ip in document("db-inproc.xml")/dblp/inproceedings,
    $au in $ip/author/text(),
    $it in $ip/title/text()

where $au = "Jeffrey D. Ullman"
return <paper>{$it}
<coauthors>
{
for $co in $ip/author
where $co/text() != $au
return $co
}
    </coauthors>
</paper>
}
</result>

5. <result>
{
for $pr in fn:document("db-proc.xml")/dblp/proceedings[year/text() = "1988"],
    $pt in $pr/title,
    $pk in $pr/@key
let $inproc := document("db-inproc.xml")/dblp/inproceedings[crossref/text()
    = $pk]

return <conf-98>
{$pt}
<count> {
    fn:count($inproc)
} </count>

```

```

        </conf-98>
    }
</result>

6. <result>
{
for $pr in fn:document("db-proc.xml")/dblp/proceedings[year/text() =
    "1996" and fn:contains(title/text(), "VLDB") and
    not(fn:contains(title/text(), "Workshop"))],
    $pk in $pr/@key,
    $au in fn:distinct-values(document("db-inproc.xml")/dblp/inproceedings
        [crossref/text() = $pk]/author)
order by $au/text()
return $au
}
</result>

7. <result>
{
for $pr in fn:document("db-proc.xml")/dblp/proceedings[year/text() =
    "1996" and fn:contains(title/text(), "VLDB") and
    not(fn:contains(title/text(), "Workshop"))],
    $ti in $pr/title,
    $pk in $pr/@key,
    $au in fn:document("db-inproc.xml")/dblp/inproceedings[crossref/text()
        = $pk]/author[fn:position() = fn:last()]/text()
return <paper> { $ti }
    <last-author> { $au } </last-author>
</paper>
}
</result>

```

Problem 2:

There are many possible schemas for this one, and virtually anything reasonable was acceptable.

The “distributes” and “authors” relationship sets might be encoded in XML simply via *containment* of attributes: e.g., a distributor might be represented a parent element with a set of CDs inside it. In an RDBMS, we must explicitly encode every relationship set as a separate relation (unless it is 1:n or 1:1).