Mid-term Exam -- ELG 5125

October 28, 2003; 17:30 – 19:20 (no documentation allowed)

Please answer 5 questions out of the 6 questions below. Each of the 5 questions counts for 20 points (for a total of 100). Note: this means, you have on average 20 minutes per question, plus 10 minutes for final revision.

Please limit your answers to approximately half a page per question. If your answer for some question is longer than one page, I will ignore what you wrote on the additional page. Please start a new page for each question.

Question 1: Network QoS

The access link from a home computer to the Internet over a telephone connection with a modem of 56kbps limits the amount of data that can be sent and received by the computer.

(a) Let us assume that low-quality video is streamed from a server on the Internet to the home computer. Does it make sense to model the Internet access link of the home computer with the model of a "leaky bucket" or a "token bucket" ? (see Annex for diagrams representing these models) – What would be the values of the characteristic parameters of these bucket models (bucket size and leaking rate for the leaky bucket, or token arrival rate and bucket size for the token bucket) ? – Please discuss whether the leaky bucket and/or the token bucket model applies, and what the appropriate parameters would be.

(b) Compare the application of streamed video with a file transfer application. Would the bucket model(s) also apply to the case that a file is transfered from a remote server to the home computer ? – Please discuss.

Question 2: RTP and RTCP

The structure of the RTP and RTCP protocol headers is shown in the diagrams in the Annex. The RTCP protocol provides feedback to the sender about the network quality of service seen by the receivers.

(a) Please explain for each of the fields in the RTP header what they are useful for.

(b) Please explain what kind of feedback information is provided by the RTCP protocol.

(c) In the case of multicasting to millions of receivers, how does the RTCP protocol avoid that the sender is flooded with too many feedback packets ? – Please explain.

Question 3: XML etc.

A car dealer has the following statistics about car sales. We assume that this information is stored in XML format and includes for each year a sequence of pairs of values: the name of the car model and the number of cars sold. The following is an example of a character string representing such an XML-encoded statistics:

```
<?xml version=1.0"?>
<sales>
        <yearlyInfo>
```

```
        <year>1999</year>
        <models>
            <sold model="Audi"> 25</sold>
            <sold model="Toyota"> 38</sold>
        </models>
    </yearlyInfo>
    <yearlyInfo>
        <year>2000</year>
        <models>
            <sold model="Audi"> 31</sold>
        </models>
    </yearlyInfo>
</sales>
```

(a) Please write a DTD which defines this structure of sales statistics. (Note: as a reference, you can find an example DTD for a recipe and an example recipe XML object in the annex; this is the example discussed in class).
(b) What are the advantages (and disadvantages) of XML as compared with other ways of encoding this information in a datafile ? (please explain; what other ways of encoding do you consider ?) – If the main purpose of encoding is the transmission of the information from an application running in one computer to another application running in another computer, are there other ways of encoding this information that could be useful ?

Question 4: Adaptive multimedia applications

Give an example of an "adaptive distributed multimedia application" (that is, a distributed multimedia application that adapts to different quality of service requirements of different users, to limitations of the clients' workstations and to limitations of the quality of service from the network or a server.

(a) Explain what kind of limitations could be imposed by the client workstation, the network or the server, and what the requirements of the user may be.
(b) Explain how your selected adaptive application would manage the QoS adaptation.

Question 5: QoS over the Internet

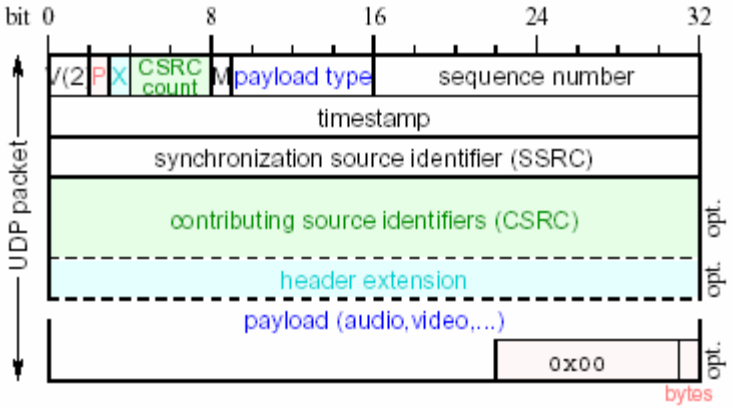
(a) What is the main difference between the proposals of "Differentiated Services" (DiffServ) and "Integrated Services" (IntServ) for the Internet ? – Please explain.
(b) For multimedia applications, one would like some kind of QoS guarantees for available bandwidth, delay, jitter and loss rate. Please discuss how the DiffServ and IntServ address these issues.

Question 6: Digital representation of media

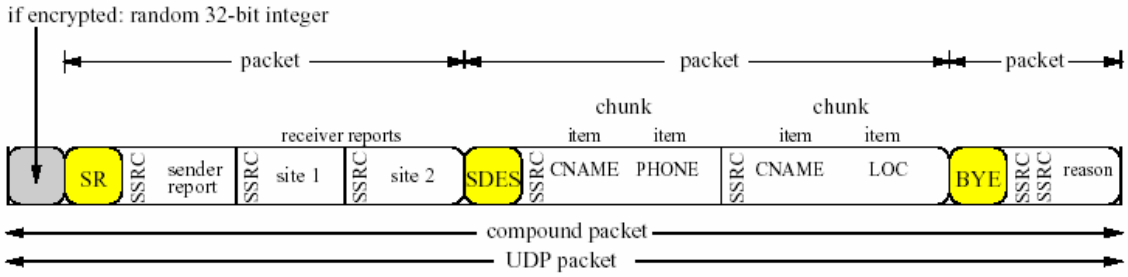
We consider the conversion of the original analogue multimedia information (which captures the real world) into a digital representation. What factors determine the quality of the obtained digital representation in the case of (a) audio, (b) image, and (c) video information ? – Explain the nature of these factors. We do NOT consider here the

question of how to compress this information into a compact coding scheme (e.g. MPEG encoding).

RTP packet header



RTCP packet structure



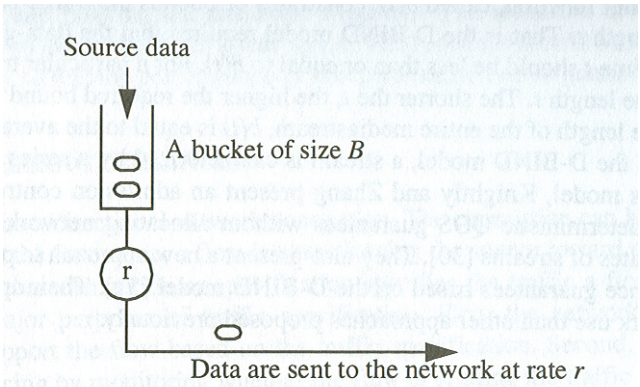


Figure 5.15 Operation of a simple leaky bucket.

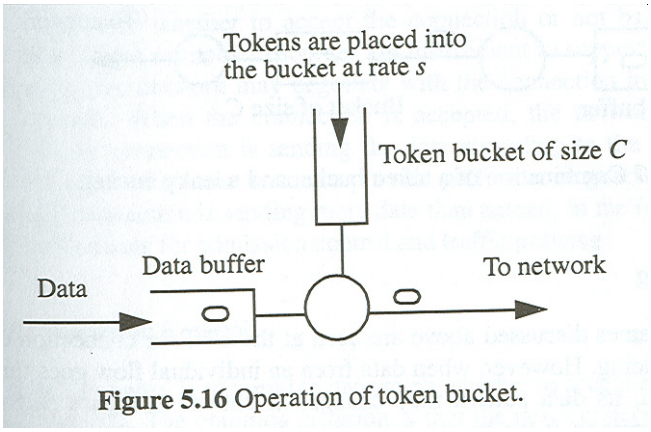


Figure 5.16 Operation of token bucket.

XML Example: Recipes

```
<?xml version="1.0"?>
<Recipe>
   <Name>Lime Jello Marshmallow Cottage Cheese Surprise</Name>
   <Description>
     My grandma's favorite (may she rest in peace).
   </Description>
   <Ingredients>
      <Ingredient>
         <Qty unit="box">1</Qty>
         <Item>lime gelatin</Item>
      </Ingredient>
      <Ingredient>
         <Qty unit="g">500</Qty>
         <Item>multicolored tiny marshmallows</Item>
      </Ingredient>
      <Ingredient>
         <Qty unit="ml">500</Qty>
         <Item>Cottage cheese</Item>
      </Ingredient>
      <Ingredient>
         <Qty unit="dash"/>
         <Item optional="1">Tabasco sauce</Item>
      </Ingredient>
   </Ingredients>
   <Instructions>
      <Step>
Prepare lime gelatin according to package instructions
      </Step>
      <!-- And so on... -->
   </Instructions>
</Recipe>

 - - - - - - - - - - - - - - -

<!-- This is the example DTD for the example XML -->
<!ELEMENT Recipe (Name, Description?, Ingredients?,
Instructions?)>
<!ELEMENT Name (#PCDATA)>
<!ELEMENT Description (#PCDATA)>
<!ELEMENT Ingredients (Ingredient)*>
<!ELEMENT Ingredient (Qty, Item)>
<!ELEMENT Qty (#PCDATA)>
<!ATTLIST Qty unit CDATA #REQUIRED>
<!ELEMENT Item (#PCDATA)>
<!ATTLIST Item optional CDATA "0"
                  isVegetarian CDATA "true">
<!ELEMENT Instructions (Step)+>
```