INTERNET-DRAFT C. W. Ng
Document: draft-ng-opes-irmlqos-01.txt P. Y. Tan
Expires: August 2002 H. Cheng
Panasonic Singapore Labs
July 2001

Quality of Service Extension to IRML

Status of this Memo

Abstract

The Intermediary Rule Markup Language (IRML) [2] is an XML-based language that can be used to describe service-specific execution rules for network edge intermediaries under the Open Pluggable Edge Services (OPES) framework, as described in [3] and [4]. This memo illustrates examples of employing the IRML for Quality of Service (QoS) policing and control, and proposes a QoS sub-system extension to IRML for better QoS support in the OPES framework.

Table of Contents

1. Introduction

The Intermediary Rule Markup Language (IRML) [2] is an XML-based language that can be used to describe service-specific execution rules for network edge intermediaries under the Open Pluggable Edge Services (OPES) framework, as described in [3] and [4]. This memo specifies a Quality of Service (QoS) subs-system in the IRML, and illustrates examples of employing the IRML for QoS policing and control.

This memo begins in Section 2 by illustrating a few scenarios where QoS policing and control can be incorporated into the OPES intermediary. From there, a set of preliminary requirements for QoS sub-system extension to the IRML is drafted in Section 3. Section 4 defines the set of QoS parameters used in the ôpropertyö and ôvariableö elements in the proposed QoS sub-system, and Section 5 presents some examples illustrating possible use of the QoS sub-

system.

1.1. Terms Used

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC-2119 [5].

2. Example Services for QoS Policing in OPES Services

2.1. Adaptation of HTML Contents

By far, Hyper Text Markup Language (HTML) pages are the most common content transported by the Hyper Text Transfer Protocol (HTTP). These HTML contents are usually static, making them ideal candidate to be cached at the network edge. However, with increasing number of "thin" clients, it is virtually impossible to have a HTML page that is suitable to be viewed for all the possible browsers. Adaptation of HTML pages to suit the clientÆs browser is widely employed (through means of server-side includes such as PHP or client-side JavaScripts). One excellent example would be the adaptation of HTML to WML (Wireless Markup Language) pages.

Adaptation of HTML pages can also be employed to suit the user or access-provider QoS requirements. For example, it might be necessary to remove redundant information when translating HTML pages to WML for a mobile phone client with a very limited bandwidth (and limited screen size). It will also be helpful to replace the <IMG> tags in the original HTML page to their ALT text equivalence. For another client who is using a Personal Digital Assistant (PDA) with a Wideband-CDMA connection, the translated WML may include more of the original HTML contents and some pictures.

Bandwidth is not the only consideration. For the example of the PDA client quoted above, when the channel condition is poor, it might be desirable to reduce the amount of text in the translated WML to ensure a more speedy delivery.

Such HTML adaptation is not only limited to the wireless scenario. For a client with a wired connection to the Internet, it is sometimes necessary to sub-sample the embedded images in the HTML

pages to reduce their size so as to meet the maximum delay requirements imposed by the user or access-provider. This can occur when the allocated bandwidth is small, or when the connection is congested (e.g. the user is downloading a couple of files simultaneously).

2.2. Dynamic Adaptation of Streaming Contents

Streaming of audio-visual (AV) contents over the Internet has become increasing popular over the years. AV streaming poses different technical challenges as compared to HTML delivery, with stricter QoS requirements, such as maximum delay and constant/variable bit-rates. When the transport technologies employed within the network core are best-effort delivery in nature, it is very difficult to guarantee the required quality of service. Thus, to maintain the perceived QoS by the user, it is often necessary to dynamically adapt the AV stream to the fluctuating network connection conditions [6]. Ideally such adaptation services should be placed near the end-user, so as to reduce the errors in estimating the network conditions at the network edge. This also allows for intermediary to cache the AV contents, and directing the contents to the adaptation service depending on the QoS requirements and channel conditions.

The use of OPES intermediaries to adapt the AV contents requires that the QoS parameters, such as allocated/requested bandwidth and channel conditions, be made available to the rule decision engine.

2.3. QoS Policy Control

Often, the OPES services will reside on an intermediary box that is provided by the access provider. Thus one major application of the IRML is to implement policy rules based on the Service Level Agreement (SLA) between the access provider and the end-user. QoS is one important aspect of SLA.

To illustrate, consider the following scenario where an end-user has a service policy of 64kbps bandwidth. Suppose the end-user is in the middle of watching a movie at 56kbps when a 16kbps voice-over-IP (VoIP) stream arrives. Instead of simply rejecting the VoIP connection, with IRML services, the access provider (or even the end-user) can now specify policy rules to perform any of following more desirable actions:

(a) adapts the VoIP stream to the remaining bandwidth,

(b) adapts the movie stream to give room for the VoIP stream (e.g. mute the audio from the movie), or

(c) adapts both the movie and VoIP streams.

2.4. Load-Balancing

The OPES framework in [3] allows for the adaptation services to be performed remotely on a separate, dedicated server, such as an ICAP server. This allows for scalability. However, when the number of connections is small, it might not be desirable to perform remote callout, as the overhead incurred will add transmission delays. IRML can be employed to redirect content adaptation to a remote server when the load on the intermediary is high, and to use a local proxylet when the load is low. Such decision can only be done if IRML is extended to environment properties, such as server load.

In addition to serverÆs processing load, such decision may based on the end-userÆs QoS requirement as well. For instance, when the server load is moderate, it might process adaptation services for end-users with stricter QoS requirements, and invoke remote adaptation services for end-users with lower QoS requirements.

3. Requirements for QoS Sub-System

In consideration to the example services illustrated in the previous sections, a preliminary set of requirements for the QoS sub-system extension to the IRML can be outlined as follow:

- It SHOULD enable rule modules to match the end-user QoS policy requirements against pre-defined labels. Such QoS policy requirements MAY include, but not limited to, the allocated bandwidth to the end-user, the requested bandwidth for the current connection, and the required delay bound, if any.

- It SHOULD enable rule modules to match the transmission statistics of the end-user connection with the intermediary against pre-defined labels.

- It SHOULD enable rule modules to match the current system load of the intermediary against pre-defined labels. The system load MAY be inferred from percentage load, and/or the number of connections the intermediary is handling.

The above set of requirements is not exhaustive. Further research work needs to be carried out to evaluate the applicability of these requirements, and append additional requirements if deem

appropriate.

4. QoS Sub-System of IRML

In order to extend QoS-aware services in the OPES intermediary, it is proposed that a QoS sub-system be specified to extend the recognized property names of the "property" and ôvariableö elements in IRML to include various QoS-related properties. These values include static configuration parameters like QoS policy parameters, and dynamic parameters like network conditions values and processing load of the intermediaries. Note that to utilize these parameters, the ôpropertyö or ôvariableö elements MUST specify a ôsub-systemö attribute of ôQoSö.

Furthermore, since these parameters have numerical values, the QoS sub-system also override the ômatchesö and ônon-matchesö attributes of the ôpropertyö element to handle basic arithmetic comparison instead of regular expression. These will be described in the following sub-sections.

4.1. QoS Policy Properties

The following are the proposed QoS policy parameters to that are defined for the "property" element in the QoS sub-system. These values are access control parameters, thus the rule engine can obtain their values via an interface to an access control module. Specification of such interface/module is out of scope.

Property Name	Value
"allocated-bandwidth"	the allocated bandwidth for the end-user
"requested-bandwidth"	the requested bandwidth for this connection
"available-bandwidth"	the amount of bandwidth available for the end-user
"delay-bound"	the maximum delay requested

4.2. Network Status Properties

The following are the proposed network status parameters that are defined for the "property" element of the QoS sub-system. These

dynamic values reflect the current network link status. The rule engine can obtain these values either via an interface to a traffic monitoring module, or a Simple Network Management Protocol (SMNP) agent [7]. In the case of RTP connections, the values of these parameters can also be extracted from the RTCP sender/receiver reports [8]. Specification of such interface/module is out of scope.

Property Name	Value
"r-octets-count"	the accumulated number of octets received by the end-user
"s-octets-count"	the accumulated number of octets received by the intermediary
"r-packets-count"	the accumulated number of packets received by the end-user
"s-packets-count"	the accumulated number of packets received by the intermediary
"r-packets-lost"	the total number of packets not received by the end-user
"s-packets-lost"	the total number of packets not received by the intermediary
"r-fraction-lost"	the fraction of packets lost reported by the end-user since the previous report
"s-fraction-lost"	the fraction of packets lost reported by the intermediary since the previous report
"r-jitter"	the inter-arrival jitter reported by the end-user
"s-jitter"	the inter-arrival jitter reported by the intermediary

Because the rule engine should be stateless, it might be necessary for the module providing the values of the QoS parameters to provide additional information about the difference in the parameters values after a specific interval.

Property Name Value

"r-octets-diff"	the difference in accumulated number of octets received by the end-user of two most recent consecutive reports
"s-octets-diff"	the difference in the accumulated number of octets received by the intermediary of two most recent consecutive reports
"r-packets-diff"	the difference in the accumulated number of packets received by the end-user of two most recent consecutive reports
"s-packets-diff"	the difference in the accumulated number of packets received by the intermediary of two most recent consecutive reports
"r-packets-lost-diff"	the difference in the total number of packets not received by the end-user of two most recent consecutive reports
"s-packets-lost-diff"	the difference in the total number of packets not received by the intermediary of two most recent consecutive reports

4.3. Intermediary Load Properties

The following are the proposed server load parameters that are defined for the "property" element in the QoS sub-system. These values are system environment parameters, thus the rule engine can obtain their values via an interface to a system module. Specification of such an interface/ module is out of scope.

Property Name	Value
"system-processing-load"	the current processing load in percentage of the intermediary
"system-connections"	the current number of established end-user connections

4.4. Overriding ômatchesö and ônon-matchesö Attributes in QoS Sub-System

Since the QoS parameters defined for the ôpropertyö element are all numerical in nature, the ômatchesö and ônon-matchesö attributes of

the ôpropertyö element are overridden in the QoS sub-system to handle arithmetic comparison instead of regular expression. In QoS sub-system, the ômatchesö and ônon-matchesö handle two types of arithmetic comparisons: (1) arithmetic relation between the QoS property and a specified numerical constant, and (2) the membership of the QoS property in a specified numerical range.

Comparison between QOS property and specified numerical constant can be constructed using the ô<ö, ô>ö, ô>=ö, and ô<=ö operators. For example, the following rule

```
<property name=ösystem-processing-loadö
          matches=ö<40ö
          sub-system=öQoSö>
```

will be evaluated to be true if the current system load of the intermediary is less than 40%, and evaluated to be false otherwise.

Membership of the QoS parameter in a specified numerical range can be constructed using the ô[min,max]ö, ô[min,max)ö, ô(min, max]ö, and ô(min,max)ö mathematical symbols. For example, the following rule

```
<property name=öallocated-bandwidthö
          matches=ö[128000,256000)ö
          sub-system=öQoSö>
```

will be evaluated to be true if the bandwidth allocated is greater or equal to 128kbps and less than 256kbps.

The table below shows the evaluation results when a QoS property is evaluated with various ômatchesö attributes. In the table, ôAö and ôBö represents numerical constants (which can include a decimal point).

ômatchesö Attribute	Evaluation Result
ô<Aö	True if the QoS parameter is less than A; False otherwise.
ô<=Aö	True if the QoS parameter is less than or equal to A; false otherwise.
ô>Aö	True if the QoS parameter is greater than A;

```
                               false otherwise.
   ô>=Aö                       True if the QoS parameter is greater than or
                               equal to A; false otherwise.
   ô[A,B]ö                     True if the QoS parameter is greater than or
                               equal to A, and less than or equal to B; false
                               otherwise.
   ô[A,B)ö                     True if the QoS parameter is greater than or
                               equal to A, and less than B; false otherwise.
   ô(A,B]ö                     True if the QoS parameter is greater than A
                               and less than or equal to B; false otherwise.
   ô(A,B)ö                     True if the QoS parameter is greater than A
                               and less than B; false otherwise.
```

The ônon-matchesö attribute, when specified instead of ômatchesö, will be evaluated to be true when the arithmetic comparison is evaluated to be false, and vice versa.

4.5. The ôcase-sensitiveö Attribute

Though the ôpropertyö element in the standard IRML has an optional ôcase-sensitiveö attribute, they are not used in the QoS sub-system. This is a direct consequence of overriding the ômatchesö and ônon-matchesö attributes to handle arithmetic comparison instead of regular expression. Use of the ôcase-sensitiveö attribute is ignored in the QoS sub-system.

4.6. The ôcontextö Attribute

The current specification of the QoS sub-system does not define the interpretation of the ôcontextö attribute. It may be possible to use this attribute to identify the interface/module by which the value of the QoS parameter is extracted, such as ôSNMPö or ôRTCPö. Until its appropriate interpretation can be revealed after further analysis, the use of the ôcontextö attribute is ignored by the QoS sub-system.

5. Examples

The first example below illustrates the case where the adaptation of HTML page to WML page is performed with consideration to the

allocated bandwidth of the client.

```
<rule processing-point="4">
  <!-- check the allocated bandwidth and adapts accordingly -->
  <property name="allocated-bandwidth" matches="<9600"
            subsystem=öQoSö>
    <!-- Bandwidth is low, use tiny WML  -->
    <execute>
      <service>
        <uri>opes://localhost/html2wml?target=tiny</uri>
      </service>
    </execute>
  </property>
  <property name="allocated-bandwidth" matches="[9600,128000)"
            sub-system=öQoSö>
    <!-- Bandwidth is moderate, use small WML  -->
    <execute>
      <service>
        <uri>opes://localhost/html2wml?target=small</uri>
      </service>
    </execute>
  </property>
  <property name="allocated-bandwidth" matches=">=128000"
            sub-system=öQoSö>
    <!-- Bandwidth is high, use large WML  -->
    <execute>
      <service>
        <uri>opes://localhost/html2wml?target=large</uri>
      </service>
    </execute>
  </property>
</rule>
```

The second example illustrates the scenario where an adaptation service is carried out locally or remotely based on the system processor load of the OPES intermediary. It also illustrates the adaptations of video stream based on the connection status.

```
<rule processing-point="4">
  <!-- check the system load to determine using local or
       remote services -->
  <property name="system-processing-load" matches=">=80"
            sub-system=öQoSö>
    <!-- System load is high, use remote services -->
    <property name="r-fraction-lost" matches="[0.3,1]">
      <!-- high packets loss (>=30%) in end-user,
           adapts the video stream to small.  -->
```

```
        <execute>
          <service>
            <uri>opes://video.adpat.server/video-adpat</uri>
          </service>
        </execute>
      </property>
    </property>
    <property name="system-processing-load" non-matches=ö>=80ö
              sub-system=öQoSö>
      <!-- System load is not high, use local services -->
      <property name="r-fraction-lost" matches="[0.3,1]">
        <!-- high packets loss (>=30%) in end-user,
             adapts the video stream to small.  -->
        <execute>
          <service>
            <uri>opes://localhost/video-adpat</uri>
          </service>
        </execute>
      </property>
    </property>
  </rule>
```

The third example shows the adaptation of different content format for different traffic conditions gathered from the network monitoring module for a specific network node or a group of network nodes in delivering Audio-Visual content. Access to the types of content format is based on the different network conditions supplied by the network monitoring module. The rule for accessing the type of content format is being specified based on the type of property used.

```
  <rule processing-point="1">
    <!-- check the allocated bandwidth and adapts accordingly -->
    <property name="r-jitter" matches=">0.5">
      <!-- receiver jitter is high, adapt to use still pictures  -->
      <execute>
        <service>
          <uri>opes://stillimage.server/jpeg-only</uri>
          <parameter name=öresourceö type=ödynamicö>
            <variable name=örequest uriö context=ösystemö/>
          </parameter>
        </service>
      </execute>
    </property>
    <property name="r-fraction-lost" matches=">0.3">
      <!-- receiver accumulated loss is high, suggest using Forward
           error correction coded content -->
```

```
      <execute>
        <service>
          <uri>opes:// videoFEC.correct.server/video-FEC<</uri>
          <parameter name=öresourceö type=ödynamicö>
            <variable name=örequest uriö context=ösystemö/>
```



```
          </parameter>
        </service>
      </execute>
    </property>
  </rule>
```

6. IAB Considerations

This proposal is an extension to the IRML Specifications [2]. It is to the authorsÆ best knowledge that there exists no inconsistency between this memo and the IRML Specification in regards to IABÆs architectural and policy considerations.

7. Security Considerations

All security considerations in [2] are applicable to the QoS sub-system. There is no security issue to the authorsÆ best knowledge that is specific to the QoS sub-system.

8. References

[1] Bradner, S., "The Internet Standard Process û Revision 3", BCP 9, RFC 2026, October 1996.

[2] Beck, A., Hoffman, M., "IRML: A Rule Specification Language for Intermediary Services", Work In Progress, draft-beck-opes-irml-02.txt, November 2001.

[3] Tomlinson, G., Orman, H., Condry, M., Kempf, J., "Extensible Proxy Services Framework", Work In Progress, draft-tomlinson-epsfw-00.txt, 2000.

[4] Beck, A., Hoffman, M., Condry, M., "Example Services for Network Edge Proxies", Work In Progress, draft-beck-opes-esfnep-01.txt,

November 2000.

[5] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[6] Wu., D., Hou, Y. T., Zhang, Y., "Scalable Video Coding and Transport over Broad-band Wireless Networks", Proc. IEEE, vol. 89, no. 1, pg 6-20, Jan 2001.

[7] Case, J.D., Fedor, M., Schoffstall, M.L., Davin, C., ôSimple Network Management Protocol (SNMP)ö, RFC 1157, May 1990.

[8] Schulzrine, H., Casner, S., Frederick, R., Jabcobson, V., "RTP: A Transport Protocol for Real-Time Applications", RFC 1889, January 1996.

9. Author's Address

Chan-Wah Ng
Panasonic Singapore Laboratories Pte Ltd
Blk 1022 Tai Seng Ave #04-3530
Tai Seng Industrial Estate
Singapore 534415
Phone: (+65) 550 5420
Email: cwng@psl.com.sg

PekûYew TAN
Panasonic Singapore Laboratories Pte Ltd
Blk 1022 Tai Seng Ave #04-3530
Tai Seng Industrial Estate
Singapore 534415
Phone: (+65) 550 5470
Email: pytan@psl.com.sg

Hong CHENG
Panasonic Singapore Laboratories Pte Ltd
Blk 1022 Tai Seng Ave #04-3530
Tai Seng Industrial Estate
Singapore 534415

Phone: (+65) 550 5477
Email: hcheng@psl.com.sg