A Comprehensive Framework for Broadcasting Multimedia Content in the Future Mobile Internet

Cristian Hesselman[1,2]*, Ing Widya*[2]*, Henk Eertink*[1]*, and Erik Huizer*[2]

[1]Telematica Instituut, The Netherlands
`{hesselman, eertink}@telin.nl`
[2]University of Twente, The Netherlands
`{widya, huizer}@cs.utwente.nl`

Abstract

In this paper, we present a comprehensive role-based framework for multimedia broadcasts (e.g., for applications such as TV, radio, e-cinema, and e-learning) taking place in a heterogeneous mobile Internet. The added value of the framework is the strong separation between application and network-level roles in combination with support for unrestricted mobility. The framework concentrates on the perceptual quality levels at which broadcasts can be received and does not require any IP-level QoS or mobility functions. We are currently developing an application-level platform that is based on the framework.

1. Introduction

The mobile Internet of the future will consist of many different Internet Service Providers (stationary and moving, public and private) that offer IP-level connectivity to a wide variety of mobile devices (from laptops down to PDAs) [1-3]. The network infrastructure of these Internet Service Providers (ISPs) will consist of one or more subnets [4], which may be based on different network technologies (e.g., wireless LAN hot spots in cellular environments) [1]. The coverage areas of such heterogeneous subnets may overlap, thus opening up possibilities for the delivery of seamless services to mobile users.

We are currently developing an application-level platform that supports multimedia broadcasts [5-10] (e.g., for applications such as TV, radio, e-cinema, and e-learning [11]) taking place in the environment outlined above.

In this paper, we introduce the framework that we are using as a context for the design and development of our platform. The framework is based on functional roles (cf. [13, 14]). It follows current trends in content distribution in that it separates network-level roles (i.e., ISPs) from application-level roles [11, 15, 16] (e.g., performed by Akamai-like organizations). The principal application-level roles are mobile clients (i.e., mobile devices) and access streamers. Access streamers are mobility and quality aware and send multimedia broadcasts to mobile clients via ISPs. In this paper, we do not require any QoS and mobility functions (e.g., IntServ, Mobile IP, etc.) from ISPs.

We focus on the challenge of unrestricted mobility [3, 17-21]. This means that clients must be able to move across subnets, ISPs and access streamers, possibly in an overlay situation [18, 20] and while they receive a broadcast.

The added value of our framework is the strong separation between application and network-level roles in combination with support for unrestricted mobility. As a result of our application-network separation, a user has a subscription with at least one home access streamer and one home ISP (cf. [14]). For the same reason, we also distinguish between application-level handoffs [24-29], network-level handoffs [18, 20, 30-35], and application and network-level roaming agreements [21, 36, 37].

We furthermore concentrate on the perceptual quality level [11, 22, 23] at which a client receives a broadcast (e.g., at a 'TV' quality level) and the required changes as a result of mobility [23] (e.g., from a 'TV' to a 'videophone' quality level).

The work that comes closest to ours is that of MarconiNet [12]. However, they do not use a strict application-network separation, nor do they consider quality issues. In addition, they do not seem to explicitly address an environment involving different wired and wireless network technologies.

In this paper, we zoom in on the framework's information model and ignore the underlying protocols and mobility mechanisms (e.g., handoff mechanisms).

The rest of this paper is organized as follows. We first discuss our model of a multimedia broadcast in Section 2. We then explain the roles of our framework in more detail in Section 3, concentrating on clients and access streamers. We consider the case of a user accessing broadcast services in Section 4, and the user moving around in Section 5. We close with conclusions and future work in Section 6.

2. Channels

A *channel* [38] carries the content of a multimedia broadcast (e.g., 'world news'). Channels are available at various perceptual *quality levels* [11, 22, 23] and can for instance be based on perceived quality assessments [23, 39]. Quality levels can be described in the form of parameters such as frame rate and pixel count [40], or in the form of well-known end-user friendly labels such as 'videophone' or 'TV' quality [11, 22]. In this paper, we will use the latter to better align with our end-user orientation.

Each quality level ultimately maps to a set of encoded and packetized digital *streams*. These streams may be based on multiple well-defined packet formats (e.g., RTP formats [41] ‘video-h261’ and ‘video-mpeg4’ [42]). Clients must receive all the streams of a specific packet format (e.g., those based on ‘video-mpeg4’) to get the associated quality level [7, 10, 43].

Box 1 shows a pseudo-XML description (cf. [44]) of a channel carrying the world news of a TV network. Receivers of the channel can for instance tune into quality level ‘videophone’ by subscribing to the appropriate streams (e.g., the MPEG4/RTP ones).

```
<channel name=”world news” source=“tv_network.nl”>
  <qlevel name=“TV” resolution=“1000x1000” frame-rate=“25”
          next-lower=“videophone” next-higher=“-”> ... </qlevel>
  <qlevel name=“videophone” resolution=“500x500” frame-rate=“15”
          next-lower=“thumbnail” next-higher=“TV”>
    <streams codec=“mpeg4” format=“rtp” profile=“mpeg4-avp-xx”>
      <stream name=“base” bandwidth=“500kbps”> ... </stream>
      <stream name=“enhancement1” bandwidth=“200kbps” ... > ... </stream>
    </streams>
    <streams codec=“h261” format=“rtp” profile=“h261-avp-31”>
      ...
    </streams>
    <streams codec=“jpeg” format=“rtp” profile=“jpeg-avp-26”>
      ...
    </streams>
  </qlevel>
  <qlevel name=“thumbnail” resolution=“250x250” frame-rate=“5”
          next-lower=“-” next-higher=“videophone”> ... </qlevel>
</channel>
```

Box 1: Channel description.

3. Roles

The primary roles of our framework are (mobile) clients, access streamers, and ISPs.

A *client* receives a channel from an access streamer via (a subnet of) an ISP, depacketizes and decodes the channel’s a/v content, and presents it to the mobile end-user. Examples of clients are laptops, PDAs, and pen tablets.

An *access streamer* delivers channels to mobile clients at mobile-specific quality levels (e.g., ‘mobile TV’[1]). An access streamer keeps track of the subnets through which it serves its clients at which quality levels. It uses this information as an input for a policy-based procedure that controls the availability of quality levels (see Section 4). For example, an access streamer could use a policy that only makes high-end quality levels available when there are enough clients that have the capabilities to receive the quality level.

An *ISP* conveys channels from access streamers to clients. The network infrastructure of an ISP consists of wired or wireless IP subnets. In this paper, we do not require any QoS and mobility functions (e.g., IntServ, Mobile IP, etc.) from ISPs.

An access streamer may be *co-located* with an ISP [12] (e.g., in the case of a walled-garden telco operator). Alternatively, an access streamer may be *separated* from the ISPs it serves. In this case, the access streamer (e.g., an Akamai-like organization) connects to the ISPs through backbone connections. Figure 1 illustrates both possibilities for the distribution of channel ‘world news’ via an access streamer AS and an ISP I.

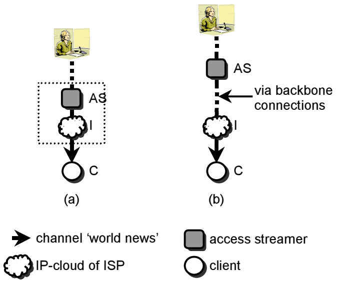

Figure 1: Collocated (a) and separated (b).

The set of access streamers that a client can *reach* from one subnet may different from the set of access streamers it can reach from another subnet. For example, the client in Figure 2a can reach access streamers AS1 from subnet N1 and AS2 from subnet N2. This situation might for instance occur when AS1 and N1 belong to one organization (cf. Figure 1a), and AS2 and N2 to another (e.g., two telco operators). Alternatively, N1 and N2 could be part of two different ISPs that each use separate access streamers AS1 and AS2 (e.g., two Akamai-like organizations). Figure 2b shows an example in which a client can reach the same access streamer (AS) from two different subnets, N3 and N4. This situation can for instance occur when N3 and N4 belong to different ISPs that both use AS as a separate access streamer. Figure 2c and 2d show similar configurations for overlays [18, 20]. We will use Figure 2c as a running example throughout this paper.

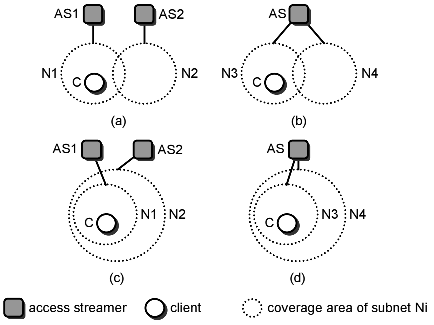

Figure 2: Reachability.

The access streamers that a client can reach are either the original *source* of a channel or an *intermediary*. In addition, the distribution of a channel may involve streamers that are not mobility-aware (i.e., that are not access streamers). We use the roles of an origin streamer and a surrogate streamer [15] to model these situations.

An *origin streamer* is the source of a channel. It manages the multimedia content of content creators (e.g., CNN, Walt Disney, and universities), digitizes the content into channels, and forwards them to access or surrogate streamers. An origin streamer that also possesses mobility functions is an access streamer.

A *surrogate streamer* is an intermediary that can manipulate channels [5, 6, 22, 23, 43, 45, 46]. It can for instance reduce the quality of channel ‘world news’ (Figure 1) from a ‘TV’ to a ‘videophone’ level, or transcode to another encoding format. However, the content provider ultimately determines which operations are allowed (e.g., no transcoding for commercial content). A surrogate streamer receives

---

[1] From now on, we prefix quality labels with ‘m-’ instead of ‘moblile’ (e.g., ‘m-TV’).

channels from origin streamers or surrogate streamers, and forwards them to access streamers or to other surrogate streamers. A surrogate streamer that also possesses mobility functions is an access streamer.

Figure 3a shows an access streamer (AS) that is the origin source of channel ‘world news’ (end-to-end approach). Figures 3b and 3c show access streamers that are intermediaries. The access streamer of Figure 3b directly connects to an origin streamer (OS), whereas the access streamer of Figure 3c connects to the origin streamer through one (or more) surrogate streamers (SS). The streamers and the clients thus form a tree. For simplicity, Figure 3 does not show ISPs.

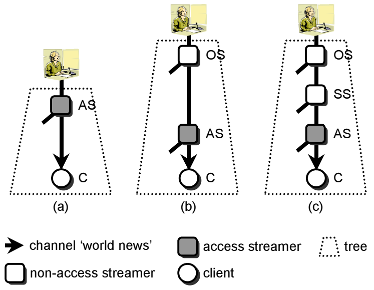

Figure 3: End-to-end (a) and proxied (b, c).

4. Access

In this section, we address the case in which a user logs onto a client to access broadcast services. Since a client may be able to reach multiple access streamers, ISPs, and subnets from its current location (cf. Figure 2), service access may require the client to interact with a mixture of home and foreign access streamers and ISPs. In this section, we assume that clients do not move.

In our framework, access to broadcast services involves four logical phases: subscription (Section 4.1), browsing (Section 4.2), establishment (Section 4.3), and usage (Section 4.4). In this paper, we concentrate on the results of these phases. We also summarize their relationship to the roles of our framework (Section 4.5).

4.1 Subscription

The result of the subscription phase consists of a delivery agreement between the end-user and an access streamer (the user’s home access streamer) and a connectivity agreement between the end-user and an ISP (the user’s home ISP).

A *delivery agreement* defines the quality levels at which the user can potentially receive channels from the access streamer. For each supported quality level, a delivery agreement also defines constraints that limit the availability of the quality level, for example to a certain time of day, or to certain types of clients (e.g., clients with a sufficiently large display).

Box 2 shows an example of a delivery agreement between user tom@domain and its home access streamer AS1. The agreement shows that tom@domain can potentially receive three quality levels from AS1: ‘m-TV’, ‘m-videophone’, and ‘m-thumbnail’. The constraints limit the availability of these quality levels to ‘m-TV’ and ‘m-thumbnail’ between 9am and 1pm, or to ‘m-videophone’ and ‘m-thumbnail’ between 1pm and 9pm.

```
<delivery-agreement user=‶tom@domain″>
  <access-streamer name=‶AS1″>
    <!-- Quality levels the access streamer supports for tom@domain -->
    <qlevel name=‶m-TV″ ...>
      <constraint time=‶9am-1pm″> </constraint>
    </qlevel>
    <qlevel name=‶m-videophone″ ...>
      <constraint time=‶1pm-9pm″> </constraint>
    </qlevel>
    <qlevel name=‶m-thumbnail″ ...>
      <!-- No constraints -->
    </qlevel>
  </access-streamer>
  ...
</delivery-agreement>
```

Box 2: Delivery agreement.

The set of supported quality levels and the constraints of a delivery agreement are based on the policies of the access streamer. These policies may pertain to resource usage (e.g., high-end quality levels such as ‘m-TV’ are not supported during rush hour) and subscription types (e.g., a ‘bronze’ subscription only supports low-end quality level ‘m-thumbnail’).

A *connectivity agreement* defines which subnets a user can access (e.g., 802.11 and UMTS subnets) and the maximum amount of IP-level resources (typically bandwidth) the user can consume on these subnets. Box 3 shows an example in which tom@domain can use the 802.11 and UMTS subnets of its home ISP, ISP1. The 802.11 subnet is however only accessible between 9am and 1pm.

```
<connectivity-agreement user=‶tom@domain″>
  <isp name=‶ISP1″>
    <subnet address=‶1.2.3.0″ mask=‶0.0.0.255″ type=‶802.11″ ...>
      <bandwidth max=‶500″>
        <constraint start=‶9am″ end=‶1pm″> </constraint>
      </bandwidth>
    </subnet>
    <subnet address=‶1.2.4.0″ mask=‶0.0.0.255″ type=‶UMTS″ ...>
      <bandwidth max=‶200″>
        <!-- No constraints -->
      </bandwidth>
    </subnet>
  </isp>
  ...
</connectivity-agreement>
```

Box 3: Connectivity agreement.

Delivery agreements and connectivity agreements are pre-negotiated [14] (beyond the scope of this paper). They typically exist as long as the end-user has a subscription with the associated access streamer and ISP, respectively.

4.2 Browsing

The browsing phase starts when the user has logged onto a client and starts the application from which it can access channels (typically an electronic program guide like the Mbone tool sdr). The result of the browsing phase consists of a list of high-level channel descriptions (e.g., in the form of their names) that the client presents to the user. The client uses the access streamers (home and foreign) it can reach at its current location to construct the list.

4.3 Establishment

The establishment phase begins when a user selects a channel (e.g., ‘world news’) and results in a channel state and a connectivity state.

A *channel state* indicates which reachable access streamers support the channel the user has selected. For each of these access streamers, the channel state also specifies which quality levels are available to the client to receive the channel. A channel state exists as long as the user has selected the channel.

Box 4 illustrates what a channel state might look like for Figure 2c. In this case, user tom@domain uses mobile client C. C can choose between access streamers AS1 and AS2 that both offer ‘world news’ at the client’s current location. Access streamer AS1 offers C two of tom@domain’s quality levels (‘m-videophone’, and ‘m-thumbnail’, cf. Box 2) for channel ‘world news’. Observe that the channel state of Box 4 also indicates how the access streamer or the client expects to receive feedback on the quality levels. RTCP can used for this purpose, but due to the coarse granularity of RTCP feedback we prefer to use an application-level beaconing algorithm [47] with a frequency described in the channel state.

```
<!-- The user has selected channel ‘world news’ -->
<channel-state user=“tom@domain” client=“C”>
  <!-- AS1 and AS2 are reachable and support ‘world news’ -->
  <access-streamer name=“AS1” is-home=“true”>
    <feedback type=”beacons” frequency=”1/100”> ... </feedback>
    <!-- ‘m-TV’ quality level is unavailable -->
    <channel name=“world news” origin-streamer=“tv_network”>
      <!-- These quality levels that are actually available from AS2 -->
      <!-- TV level is unavailble (e.g., due to lack of resources) -->
      <qlevel name=“m-videophone” ...>
        <streams codec=“mpep4” format=“rtp” profile=“mpeg4-avp-xx”>
          <stream name=“base” group=“224.5.6.7” ...> ... </stream>
          <stream name=“enhancement1” group=“224.5.6.8” ...> ... </stream>
        </streams>
        <streams codec=“h261” format=“rtp” profile= “h261-avp-31”>
          ...
        </streams>
        <!-- jpeg version of this quality level is unavailble -->
      </qlevel>
      <qlevel name=“m-thumbnail” ...>
        ...
      </qlevel>
    </channel>
  </access-streamer>
  <access-streamer name=“AS2” is-home=“false”>
    <!-- quality levels available from AS2 -->
    ...
  </access-streamer>
</channel-state>
```

Box 4: Channel state.

A channel state is the result of an on-the-fly negotiation procedure (e.g., based on SIP [49], RTSP [48] or RTCP [41]) that takes the availability of resources on the client and in the access streamers into account. As a result, some of an access streamer’s supported quality level may not appear in a channel state (e.g., ‘m-TV’ in Box 4).

A channel state generally involves of a mixture of home and foreign access streamers. For example, the channel state in Box 4 involves one home access streamer (AS1) and one foreign access streamer (AS2). The quality levels available from a home access streamer are constrained by the delivery agreement with the end-user. For example, quality level ‘m-videophone’ (AS1) will not appear in the channel state of Box 4 if it does not appear in the user’s delivery agreement (see Box 2).

For foreign access streamers, the channel state is furthermore based on *roaming agreements*. A home access streamer establishes a roaming agreement with a foreign access streamer so that its clients can use the foreign access streamer to receive channels. A roaming agreement specifies a mapping between the quality levels of a home access streamer and those of a foreign access streamer. This mapping ultimately influences the list of quality levels the client can receive from the foreign access streamer.

Box 5 illustrates what a roaming agreement might look like. In this case, the ‘m-TV’ quality level of the home access streamer AS1 maps to the same level at the foreign access streamer AS2. The ‘m-videophone’ and ‘m-thumbnail’ quality levels both map to the thumbnail levels of the foreign access streamer.

```
<roaming-agreement access_streamer1=“AS1” access_streamer2=“AS2”>
  <qlevel-mappings from=”AS1” to=”AS2”>
    <!-- ‘to’ is the quality level in AS2 -->
    <qlevel-map from=“m-TV” to=“m-TV”> ... </qlevel-map>
    <qlevel-map from=“m-videophone” to=“m-thumbnail”> ... </qlevel-map>
    <qlevel-map from=“m-thumbnail” to=“m-thumbnail”> ... </qlevel-map>
  </qlevel-mappings>
  ...
</roaming-agreement>
```

Box 5: Roaming agreement.

Observe that the entire roaming agreement of Box 5 can either be established statically (i.e., pre-negotiated) or dynamically. In addition, it can be established bilaterally or through a roaming broker (cf. [36]).

A *connectivity state* indicates which subnets (e.g., 802.11 and UMTS subnets) and ISPs are reachable from the client’s current location. For each reachable subnet, the connectivity state also specifies the amount of IP-level resources that are currently available to receive the selected channel, either through unicast or through multicast. The channel state is derived from the user’s connectivity agreement based on the available resources in the subnets and on the client.

Box 6 shows an example of a connectivity state based on the connectivity agreement of Box 3. Observe that the maximum available bandwidth for the UMTS subnet is lower than in the connectivity agreement, for instance because of a policy of the ISP or because of other traffic.

```
<connectivity-state user=“tom@domain” client=“C1”>
  <isp name=“I1”>
    ...
  <isp>
  <isp name=“I2”>
    <subnet address=“1.2.3.0” mask=“0.0.0.255” type=“802.11” ...>
      <bandwidth max=“500”> ... </bandwidth>
    </subnet>
    <subnet address=“1.2.4.0” mask=“0.0.0.255” type=“UMTS” ...>
      <!-- max bandwith was 500 in connectivity agreement -->
      <bandwidth max=“200”> ... </bandwidth>
    </subnet>
  <isp>
</connectivity-state>
```

Box 6: Connectivity state.

In our framework, roaming agreements are also set up between ISPs (cf. [21, 36, 37]). The purpose of these agreements is similar to streamer-level roaming agreements, but they pertain to network-level roaming issues (e.g., bandwidth).

4.4 Usage

The usage phase starts with the automatic selection of a quality level and a subnet. This may for instance be done based on user profiles (e.g., containing the user’s preferred quality levels). As an example, suppose that the end-user has selected channel ‘world news’ and that the selected quality level is ‘m-videophone’ of access streamer AS1 (Box 4). In this case, the client must subscribe to multicast groups 224.5.6.7 and 224.5.6.8 to receive the channel at videophone quality in MPEG4 format. The result of the usage phase is that channel is being forwarded to the client at the selected quality level via the selected subnet.

The channel and connectivity states may *change* dynamically during the usage phase. There are various reasons for such a change. For example, the channel state shown in Box 4 may change because AS1 no longer supports ‘m-videophone’ as a result of AS1’s policies. Another reason may be that the available bandwidth between the client and access streamer AS1 drops to a level that 'm-videophone' can no longer be supported. These events require a re-establishment of the state of Box 3 and might therefore require a renegotiation. The net result may be that another quality level will be selected and that a client must switch to this level (e.g., from 'm-videophone' to the 'm-thumbnail' level) or to another set of streams of the same level (e.g., from m-videophone's MPEG4 streams to its H261 streams). Observe that the newly selected quality level may belong to another access streamer (e.g., AS2).

4.5 Summary

Figures 4 and 5 summarize the agreements, states, and roles of our framework in UML [53] class diagrams. For simplicity, we have omitted class attributes and methods, as well as the multiplicities of associations.

Figure 4 depicts the relation between users and their home access streamers and home ISPs during the subscription phase (Section 4.1). The delivery agreement of Box 2 illustrates an instance of the association class `DeliveryAgreement`. User tom@domain and access streamer AS1 are references to instances of classes `User` and `AccessStreamer`, respectively. Similarly, the connectivity agreement of Box 3 illustrates an instance of `ConnectivityAgreement`, while ISP1 is a reference to an instance of class `ISP`.

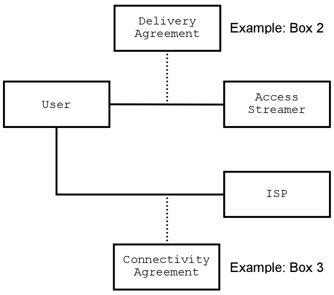

Figure 4: Class structure during the subscription phase.

Figure 5 shows the relations between clients, access streamers, and ISPs during the establishment phase (Section 4.3). The channel state of Box 4 illustrates an instance of class `ChannelState`. It involves a reference to an instance of `Client` (C) and two references to instances of `AccessStreamer` (AS1 and AS2). The roaming agreement of Box 5 illustrates an instance of `AS-RoamingAgreement` and also involves references to two instances of `AccessStreamer` (AS1 and AS2).

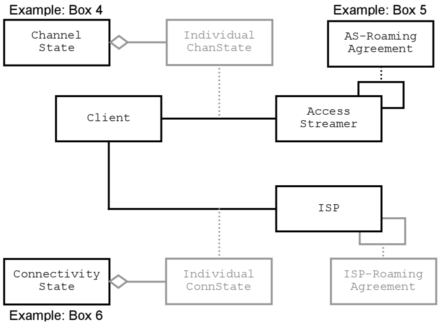

Figure 5: Class structure during the establishment phase.

The class diagram of the usage phase (Section 4.4) is similar to that of the establishment phase.

5. Movement

In this section, we address the case in which a mobile client moves while it receives a channel (i.e., it is in the usage phase). We concentrate on handoffs at the level of access streamers as a result of handoffs between subnets.

Since our framework does not require any IP-level mobility functions, we handle mobility at the application-level [24-29]. We use client-controlled handoffs [50]. In this paper, we assume that clients automatically discover which subnets they can reach (e.g., through DHCP) and update the connectivity state accordingly.

The set of available access streamers in the channel state can change when the set of reachable subnets in the connectivity state changes. For example, when the client of Figure 2c moves out of subnet N1, N1 will disappear from the connectivity state while access streamer AS1 and its quality levels will disappear from the channel state. When C moves back into N1, N1 and AS1 will reappear.

In our framework, a client can automatically select a new quality level when the channel state changes (e.g., based on the user's preferred quality levels). The target quality level may be provided by a different access streamer, thus requiring the client to handoff to that access streamer (application-level handoff). To receive the target quality level, the client drops the streams of the old quality level and begins to receive those of the new quality level [47] (adaptation). Experiments with a rudimentary version of our platform [47] showed that it takes several seconds before the streams of the old quality level can be dropped and those of the new quality level can be displayed.

To provide continuous service to clients, it may be crucial that the channel state is updated as fast as possible. For example, when client C receives channel 'world news' from access streamer AS1 via subnet N1 (Figure 2c) and moves out of N1's coverage area, it must handoff to access streamer AS2. For a smooth handoff, AS2 must therefore appear in the channel state of 'world news' as soon as possible. This requires fast discovery and negotiation procedures. To execute the handoff in a smooth manner, it also requires fast initialization procedures (e.g., when the streams of the target quality level are based on a different packet format).

In our framework, the adaptation from one quality level to another is based on policies. In the past, we have developed such adaptation policies in overlay situations using packet loss characteristics [47]. These policies define when a handoff should start and when it should complete. Similar policies based on available bandwidth are described in [23]. [51] considers network-level policies based on aspects such as available bandwidth, power consumption, and connection set up delay.

For reasons of security, we assume that adaptation policies are provided by clients. In a more programmable environment, they can however also be provided by access streamers [52].

6. Conclusions and Future Work

We presented a comprehensive role-based framework for the design of an application-level platform that supports broadcast applications running in a heterogeneous mobile Internet. The framework's strict separation between application and network-level roles introduces additional complexity due to the duplication of network-level concepts at the application-level (e.g., home locations, roaming agreements, and handoffs). However, standardization efforts (e.g., the OPES and CDI working groups of the IETF) and business cases (e.g., in the form of companies such as Akamai and Digital Island) indicate that this separation is desirable. We are unaware of other work that uses such a model to support unrestricted mobility in a heterogeneous mobile Internet.

We have developed a rudimentary version of our platform that allows a mobile client to handoff between two quality levels of two different access streamers in an overlay situation and analyzed its handoff behavior [47]. We are currently implementing a more advanced version of the platform, using the framework outlined in this paper as a starting point.

7. References

[1] M. Haardt, W. Mohr, "The Complete Solution for Third-Generation Wireless Communications: Two Modes on Air, One Winning Strategy", IEEE Pers. Comm., Dec. 2000

[2] H. Yumiba, K. Imai, M. Yabusaki, "IP-Based IMT Network Platform", IEEE Pers. Comm., Oct. 2001

[3] N. Drew, M. Dillinger, "Evolution Toward Reconfigurable User Equipment", IEEE Comm. Mag., Feb. 2001

[4] F. Vakil, A. Dutta, J-C. Chen, M. Tauil, S. Baba, N. Nakajima, Y. Shobatake, H. Schulzrinne, "Supporting Mobility for Multimedia with SIP", Internet Draft, draft-itsumo-sipping-mobility-multimedia-01.txt, Dec. 2001

[5] E. Amir, S. McCanne, R. Katz, "An Active Service Framework and its Application to Real-time Multimedia Transcoding", Proc. of ACM SIGCOMM'98, Vancouver, Canada, Sept. 1998

[6] K. Jonas, M. Kretschmer, J. Moedeker, "Get a KISS — Communication Infrastructure for Streaming Services in a Heterogeneous Environment", Proc. of ACM Multimedia'98, Bristol, UK, Sept. 1998

[7] L. Wu, R. Sharma, B. Smith, "Thin Streams: An Architecture for Multicast Layered Video", 7th Intl. Workshop on Network an Operating Systems Support for Digital Audio and Video (NOSSDAV97), St. Louis, USA, May 1997

[8] I. Kouvelas, V. Hardman, J. Crowcroft, "Network Adaptive Continuous-Media Applications Through Self Organised Transcoding", Proc. Network and Operating Syst. Support for Digital Audio and Video (NOSSDAV'98), July 1998, Cambridge, UK

[9] S. Cheung, M. Ammar, X. Li, "On the use of destination set grouping to improve fairness in multicast video distribution", proceedings IEEE Infocom '96, San Francisco, USA, March 1996, pp. 553-560

[10] S. McCanne, V. Jacobson, M. Vetterli, "Receiver-driven Layered Multicast", Proc. of ACM SIGCOMM, Stanford, USA, August 1996

[11] M. Vernick, S. Bryden, M. Condry, G. Disher, J. Straley, W. Walkoe, "Requirements for End-To-End Delivery of Broadband Content", Broadband Content Delivery Forum, Oct. 2001

[12] A. Dutta, W. Chen, H. Schulzrinne, O. Altintas, "Mobility Support for Wireless Streaming in MarconiNet", Proc. of IEEE Broadband Wireless Summit, Interop 2001, Las Vegas, USA, May 2001

[13] TINA Consortium, "Business Model and Reference Points", May 1997, http://www.tinac.com/specifications/documents/bm_rp.pdf

[14] DOLMEN Consortium, "Open Service Architecture for Mobile and fixed environments (OSAM)", Final Release, Version 4.0, July 1998

[15] G. Tomlinson, R. Chen, M. Hofmann, "A Model for Open Pluggable Edge Services", Internet Draft draft-tomlinson-opes-model-00.txt, July 2001

[16] M. Day, B. Cain, G. Tomlinson, P. Rzewski, "A Model for Content Interworking (CDI)", Internet Draft, draft-day-cdnp-model-07.txt, Oct. 2001

[17] K. Pahlavan, P. Krishnamurthy, A. Hatami, M. Ylianttila, J. Makela, R. Pichna, J. Vallström, "Handoff in Hybrid Mobile Data Networks", IEEE Personal Communications, April 2000

[18] M. Stemm, R. Katz, "Vertical Handoffs in Wireless Overlay Networks", ACM Mobile Networking, Special Issue on Mobile Networking and Internet, Spring 1998

[19] J. Meggers, A. Park, R. Ludwig, "Roaming between GSM and Wireless LAN", ACTS Mobile Summit, Granada, Spain, Nov. 1996

[20] E. Brewer, R. Katz, Y. Chawathe, S. Gribble, T. Hodes, G. Nguyen, M. Stemm, T. Henderson, E. Amir, H. Balakrishnan, A. Fox, V. Padmanabhan, S. Seshan, "A Network Architecture for Heterogeneous Mobile Computing", IEEE Personal Communications, Oct. 1998

[21] G. Markoulidakis, G. Lyberopoulos, D. Tsirkas, E. Sykas, "Inter-Operator Roaming Scenarios for Third Generation Mobile Telecommunication Systems", 2nd IEEE Symposium on Computers and Communications (ISCC'97), Alexandria, Egypt, July 1997

[22] N. Yeadon, F. Garcia, D. Hutshison, D. Shepherd, "Filters: QoS Support Mechanisms for Multipeer Communications", IEEE Journal on Selected Areas in Comm., Sept. 1996

[23] R. Liao, A Campbell, "On Programmable Universal Mobile Channels in a Cellular Internet", 4th ACM/IEEE International Conference on Mobile Computing and Networking (MOBICOM'98), Dallas, October, 1998

[24] E. Wedlund, H. Schulzrinne, "Mobility Support Using SIP", 2nd ACM/IEEE Int. Conf. on Wireless and Mobile Multimedia (WoWMoM'99), Seattle, USA, Aug. 1999

[25] W. Liao, "Mobile Internet Telephony Protocol (MITP): an Application-Layer Protocol for Mobile Internet Telephony Services", Proc. IEEE ICC'99, Vancouver, Canada, June 1999

[26] W. Liao, "Mobile Internet Telephony: Mobile Extensions to H.323", Proc. IEEE INFOCOM'99, New York, USA, March 1999

[27] D. Park, W. Yoon, D. Lee, “An Efficient Handoff Management for Mobility Support to H.323”, International Workshop on Mobile Multimedia Communications (MoMuC’00), Tokyo, Japan, Oct. 2000

[28] B. Landfeldt, T. Larsson, Y. Ismailov, A. Seneviratne, “SLM, a Framework for Session Layer Mobility Management”, In Proc. IEEE ICCCN, October 1999

[29] R. Singh, Y.C. Tay, W.T. Teo, S.W. Yeow, “RAT: A Quick (And Dirty?) Push for Mobility Support”, Proc. of the 2nd IEEE Workshop on Mobile Computers and Applications, New Orleans, USA, Feb. 1999

[30] J. Solomon, “Mobile IP — The Internet Unplugged”, Prentice Hall, 1998

[31] A. Helmy, “A Multicast-based Protocol for IP Mobility Support”, ACM Second International Workshop on Networked Group Communication (NGC 2000), Palo Alto, USA, November 2000

[32] C. Tan, S. Pink, K. Lye, “A Fast Handoff Scheme for Wireless Networks”, Proc. 2nd ACM Workshop on Wireless Mobile Multimedia (WoWMoM’99), Seattle, Aug. 1999

[33] J. Mysor, V. Bharghavan, “A new multicasting-based architecture for internet host mobility”, Proc. ACM/IEEE Mobicom, Sept. 1997

[34] A. Campbell, J. Gomez, S. Kim, A. Valkó, C. Wan, Z. Turányi, “Design, Implementation, and Evualation of Cellular IP”, IEEE Personal Communications, August 2000

[35] R. Ramjee, T. La Porta, S. Thuel, K. Varadhan, “HAWAII: A Domain-based Approach for Supporting Mobility in Wide-area Wireless Networks”, Seventh International Conference on Network Protocols, Toronto, Canada, 1999

[36] 3GPP, “Automatic Establishment of Roaming Relationships”, Technical Report 22.971, version 3.1.1, April 1999

[37] 3GPP, Technical Report 22.011 (no title), version 4.3.0, March 2001

[38] C. Hesselman, H. Eertink, “Broadcasting Multimedia Channels in Future Mobile Systems”, Position Paper, Proc. 6th Int’l Conference on Protocols for Multimedia Systems (PROMS'01), Enschede, The Netherlands, Oct. 2001

[39] A.Watson, M. Sasse, “Measuring Perceived Quality of Speech and Video in Multimedia Conferencing Applications”, Proc. ACM Multimedia ‘98, Sept. 1998, Bristol, England

[40] OMG Telecommunications Domain Task Force, “Control and Management of A/V Streams”, ftp://ftp.omg.org/pub/docs/formal/00-01-03.pdf

[41] H. Schulzrinne, S. Casner, R. Frederick, V. Jacobson, “RTP: A Transport Protocol for Real-Time Applications”, RFC 1889, Jan. 1996

[42] H. Schulzrinne, “RTP Profile for Audio and Video Conferences with Minimal Control”, RFC 1890, Jan. 1996

[43] A. Balachandran, A. Campbell, M. Kounavis, “Active Filters: Delivering Scaled Media to Mobile Devices”, 7th Int. Workshop on Network and Operating System Support for Digital Audio and Video (NOSSDAV'97), St. Louis, USA, May 1997

[44] Kutscher, Ott, Bormann, “Session Description and Capability Negotiation”, Internet Draft, draft-ietf-mmusic-sdpng-04.txt, March 2002

[45] B. Zenel, D. Duchamp, “A General Purpose Proxy Filtering Mechanism Applied to the Mobile Environment”, Proc. 3rd ACM/IEEE International Conference on Mobile Computing and Networking, Budapest, Hungary, Sept. 1997

[46] E. Amir, S. McCanne, H. Zhang, “An Application Level Video Gateway”, Proc. of ACM Multimedia, San Fransisco, USA, Nov. 1995

[47] C. Hesselman, H. Eertink, A. Peddemors, “Multimedia QoS Adaptation for Inter-tech Roaming”, Proc. of the 6th IEEE Symosium on Computers & Communication (ISCC 2001), Hammamet, Tunisia, July 2001

[48] H. Schulzrinne, A. Rao, R. Lanphier, “Real Time Streaming Protocol (RTSP)”, RFC 2326, April 1998

[49] M. Handly, H. Schulzrinne, E. Schooler, J. Rosenberg, “SIP: Session Initiation Protocol”, RFC 2543, March 1999

[50] N. Tripathi, J. Reed, H. VanLandingham, “Handoff in Cellular Systems”, IEEE Personal Communications, Dec. 1998

[51] H. Wang, R. Katz, J. Giese, “Policy-Enabled Handoffs Across Heterogeneous Wireless Networks”, 2nd IEEE Workshop on Mobile Computing and Applications (WMCSA 1999), New Orleans, USA, February 1999

[52] R. Liao, A. Campbell, “A Utility-Based Approach for Quantitative Adaptation in Wireless Packet Networks”, ACM Journal on Wireless Networks (WINET), Vol. 7, No. 5, pg. 541-557, Sept. 2001

[53] H.-E. Eriksson, M. Penker, “UML Toolkit”, John Wiley & Sons, 1998