Unified Layer-2 Triggers and Application-Aware Notifications

Marten Bandholz, Janne Riihijärvi and Petri Mähönen
Department of Wireless Networks, RWTH Aachen University
Kackertstrasse 9, D-52072 Aachen, Germany
{mba, jar, pma}@mobnets.rwth-aachen.de

ABSTRACT

We define a notification hierarchy including a set of unified link-layer triggers to be used for cross-layer optimization in heterogeneous wireless networks. Unlike previous suggestions, our approach is not specific to IEEE 802 -technologies, and can be used by applications as well as network and transport protocol entities. We also present full testbed implementation of our system together with the corresponding software architecture focussing on IEEE 802.11 WLAN networks. As a test case, we illustrate the benefits of our approach with a video streaming application using the retransmission ratio as a link quality indicator.

Categories and Subject Descriptors

C.2.1 [**Computer-Communication Networks**]: Network Architecture and Design—*Wireless communication*

General Terms

Design, Performance

Keywords

L2-trigger, Cross-Layer Adaption, Unified Link Layer, IEEE 802.11, RealPlayer

1. INTRODUCTION

Wireless channels are inherently dynamic in nature. Mobility of users and interferers can cause sudden changes in the quality of links used in short time periods. Even the most carefully designed wireless technologies, such as the 3G cellular systems are still prone to packet losses and widely varying latencies. Situation is even more difficult when ISM-bands are being used, inter- and intra-technology interferences causing serious problems with the perceived quality of service.

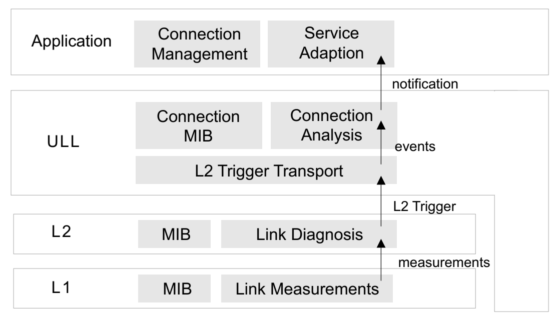

Figure 1: Notification Hierarchy using unified L2 triggers. Here ULL stands for Unified Link Layer.

While wireless technologies will certainly continue to improve, it is practically impossible to give meaningful, hard quality of service guarantees over wireless channels. We therefore consider it essential to offer applications and higher-layer protocols information about link-layer conditions that are *relevant* to the particular application. In this paper we define a hierarchical, adaptive mechanism for cross-layer optimisation based on unified approach to link-layer triggers, or notifications on changes in link conditions.

Naturally, the concept of link-layer triggers, or L2 triggers, as they are also called, is not a new one. However, none of the previous work has really been widely accepted. In our view the main reasons for this have been either the domain-specific nature of the triggers developed, or technology specific approach adopted. An example of the work of the first type is the transport and mobility trigger work carried out in various working groups of the IETF [11]. The IEEE is also working on standardizing L2 trigger definitions in the IEEE 802.21 working group [9]. However, also their work focuses on mobility and handover inter-operability, and thus targets network protocol agents rather than being applicable for a wider variety of applications. There is a clear need for more general approach, as triggers are particularly important for many future event-driven applications, especially in the sensor and embedded networking domain.

2. GOALS AND APPROACH

The main challenge in the unified approach advocated is the disparity between application interests, and quality metrics offered by various wireless technologies. (For simplicity, we take the word "applications" to include network and transport protocol entities as well.) Applications would typically prefer to operate on concepts such as bandwidth, end-to-end delay, or, indeed, the cost of using a link. However, information actually available on individual links is often oriented towards the physical layer, including measures such as bit error rate or received signal strength.

The system presented in this paper resolves this discrepancy by offering transparent, technology-independent notifications for applications by mapping technology-specific measurements into unified link-layer triggers. We adopt a hierarchical notification scheme, where applications indicate their needs with respect to connectivity, and only changes in link conditions that are significant with respect to these needs result in evaluation of the new link conditions.

The model of the notification hierarchy is shown in figure 1. Link specific measurements supplied by the physical layer are abstracted with unified L2 trigger by the means of a link diagnosis. The L2 trigger might be consumed locally or sent via the network. In order to support *remote triggers* a suitable transport protocol was proposed by Yegin *et al.* in [15]. From the point of view of higher layer entities, L2 triggers are recognized as a sort of connectivity related event, similar to events generated by the routing and transport layers, such as new_network_available or connection_timeout. All connectivity related events are mapped to application notifications by a centralized entity, called Connectivity Support (CS). This includes also a translation of the information attached to events, also called *hints* [5]. For instance, the first hop link layer addresses of the L2 triggers are mapped to end-to-end connections used by the application. The actual service adaption is then performed by the application itself and might include communication to a remote server.

In the following subsections we will discuss in greater detail some of the requirements for different classes of link and network aware applications, and then recall briefly which kinds of quality metrics are available in common link-layer technologies.

2.1 Application Requirements

The requirements between applications differ mainly in the levels of abstraction and detail the application writers wish to operate on. For a simple download client, for example, notifications of interest would mostly be limited on appearance and disappearance of links, and to the bandwidth offered by them. Other extreme would be applications of Connection Manager type, responsible for configuring network and transport protocols, managing mobility and handovers, and even helping in selection of which link to use given application requirements. Obviously an application like this would have little interest in transparency, as more the details it has available to it, more informed the decisions it can take. We shall deal with the role of connection manager in our architecture in greater detail in subsection 3.4.

Interesting middle ground between these two extremes is formed by various adaptive streaming applications. In addition to bandwidth, latencies and higher order quantities such as jitter are of considerable interest to them. Also knowledge of error characteristics can be of interest, as adaptive coding and content protection schemes can efficiently be used to enhance user experience. Streaming applications are also typically rather asymmetrical in nature, differing greatly in terms of uplink and downlink use. This brings in a strong requirement to support triggers and notifications in a directed fashion.

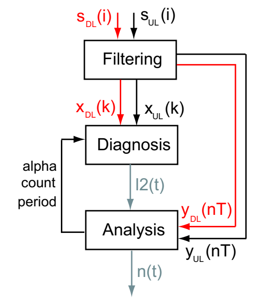

Figure 2: Model of the signal flow for uplink (UL) and downlink (DL) direction. Here following notation is used: s - measurement samples, x,y - link layer metrics, l2 - discrete L2 trigger, n - application notifications.

2.2 Link Layer Metrics

The collection of quality-related metrics monitored in different link-layer technologies is extremely heterogeneous. Cellular systems typically keep track of very complete set of physical layer characteristics, even though accessing the information on a typical terminal or user equipment relies on manufacturer-specific and often undocumented interfaces. Information tracked by, for example, WLAN (IEEE 802.11) or WPAN (IEEE 802.15.1 and IEEE 802.15.4) wireless interfaces is more readily accessible, but unfortunately rather limited in nature. As these IEEE technologies are very relevant to the present discussion, we shall discuss the available link quality indicators in slightly greater detail.

The main signal domain metrics available in both IEEE 802.11 and the IEEE 802.15 -families are the received signal strength (RSSI), and signal-to-noise-and-interference ratio (SNIR) for the downlink. Both of these are limited in time resolution by the sampling granularity induced by received frames. Naturally, the SNIR is more expressive metric, as it includes the effect of interferers. Shadowing and fading effects can cause both of these metrics to fluctuate rapidly, so filtering or averaging techniques should be employed to obtain quality measures relevant to the applications.

Third quality metric in the downlink relevant to the IEEE-technologies in question is the frame error rate (FER). Unlike in cellular systems, for example, bit error rate is not available. The FER is easily calculated, but does require sampling over substantial period of time for statistical accuracy.

Short term indications can be derived from the retransmission ratio (RR), which is commonly used in the uplink for the adaption of the transmission rate. However, it is substantially useful in the downlink direction as well, as we will see in section 5.

3. NOTIFICATION HIERARCHY BASED ON UNIFIED L2 TRIGGER

The following subsections describe the layered approach in more detail discussing the signal flow as shown in figure 2.

The input signals $s_{DL}(i)$ and $s_{UL}(i)$ represent measurement samples, which are not taken at a constant rate, but on a frame basis. Usually, simple filters on these input vectors are sufficient to gain some interesting link layer metrics $x_{DL}(k)$ and $x_{UL}(k)$. These inputs are subject to further link specific diagnosis. For instance, thresholds with a hysteresis can be applied to detect a significant change in the link layer metrics. In case the condition for a L2 trigger is fulfilled, the signal $l2(t)$ triggers the link analysis.

Furthermore link layer metrics, which are not subject to link diagnosis can be used for link comparison or qualification. The signals $y_{DL}(k)$ and $y_{UL}(k)$ capture this need of the Connectivity Support (CS) entity for an insight on link layer metrics.

3.1 Measurements

Both, physical and link layer implementations usually provide a vast amount of technology specific measurements. This includes metrics referring to signal strength, modulation or error protection schemes. Most of these real-time measurements are taken in the hardware or firmware and accumulated in counters or provided in special registers. Some of them might be available only during transmission or reception of a frame. Also, some parameters might be available for the uplink- or downlink direction only. A small overview on the available measurements within different technologies is given in [8].

3.2 Filtering

In order to provide a first abstraction of the current link state these samples are processed by the means of simple filters. The resulting link layer metrics characterize the link in a technology specific way and indicate trends or emphasize that the link is in a particular state, for instance under line-of-sight conditions or in a fading spot.

3.3 Link Diagnosis

The next level in notification hierarchy is the mapping of link-layer metrics and their changes into the unified L2 triggers. Three triggers are used in the present architecture:

- The **link_up** trigger is fired when data frames can be send and received via this link.
- The **link_down** trigger is fired when no more data frames can be send or received via this link.
- In case the link quality is deemed to have changed significantly, the **link_quality_changed** trigger is fired.

Motivated by the notion of links in [14], each trigger is fired for a particular link identified by an ordered pair of link layer addresses. The order of these addresses determines the direction sense (uplink vs. downlink direction), and is passed alongside with the L2 trigger as a hint.

In previous work, such as [6], a simple state model of link conditions with transitions based directly on measurements has been used to decide when these triggers should be fired. However, as reported in [1], this results in unreliable L2 triggers. Instead, our scheme uses a (usually linear) combination of multiple measurements, together with a free parameter called *significance* (α), which can used by higher layers to control the sensitivity of the link diagnosis. Changing the significance parameter α results in more or fewer L2 triggers. Without any implications about particular measurement-thresholds, the significance parameter abstracts link layer specific away, while leaving a form of closed-loop control for the consumer of the L2 trigger. This widens the applicability of our approach from supporting Mobile IP handovers only towards all applications sensitive to link conditions.

3.4 Connectivity-Support (CS)

If delivered locally the term L2 trigger is exchangeable with events, because no transport protocol is utilized. The events themselves are mapped to application notifications with a change in the information given as a hint, such as an end-to-end address instead of a link layer address.

Therefore the application registers for notification support indicating its remote host and the application type, such as streaming application, download client, and the notification conditions, such as bandwidth smaller then 300 kbps. The CS entity looks up the current routing tables to figure out which links are in use for these connections[1]. Events for these links are then taken into account for the application notifications: link_up and link_down map directly to connection_up and connection_down with two IP-addresses as a hint. The registered condition is evaluated upon the link_quality_changed event, so that not each event results in a notification to the application. For instance, if too many evaluations turned out to be unnecessary, the significance α is lowered. This trades an increased probability for getting an interesting link change too late for reduced computational overhead.

If a hint goes along with the application notification, for instance the bandwidth, the actual bandwidth queried from the link layer needs to be scaled to forward the bandwidth the application perceives. The scaling factor can be estimated from the overhead of the currently running protocols for this connection.

3.5 Application Notification

Service Adaption is a task performed by the application upon reception of a notification for its connection. If the application is using multiple connections it needs to register notification support for all of them individually, since they can potentially route via different links. This is especially relevant for ad-hoc networks. A simple example for service adaption is a download client starting or stopping its file transfer upon connection_up and connection_down. A more advanced application would be a Connection Manager, which starts its link selection algorithms upon major changes in the quality of an active connection.

4. TESTBED IMPLEMENTATION

The overall implementation architecture of the Linux testbed is shown in figure 3. Starting at the lowest level, it will be discussed in greater detail in the next subsections.

4.1 Measurements

Basically, measuring wireless channels puts real-time requirements on the implementation supplying this information. Hence, this task should be performed by the hardware (HW) or firmware (FW) for the sake of accurate and OS independent samples. In the testbed setup the used hardware was a Netgear MA 401 card with a Prism2.5 chipset

[1] In some future time the maintenance of links might be performed by a Connection Manager reducing the feature set of the CS entity.

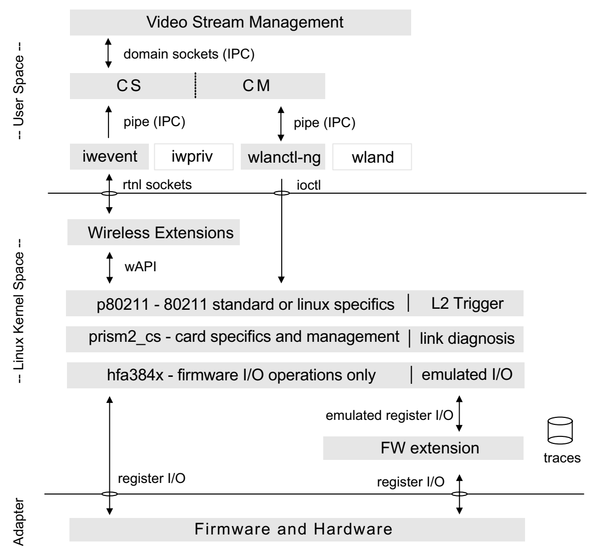

Figure 3: Overall Implementation Architecture of Linux testbed.

(v 1.0.0) and a hfa3873 series baseband controller. The FW (pri: 1.0.7, sta: 1.3.6) stores the samples in registers to be read be the wlan-ng driver [2]. The appropriate registers were identified by the means of a recent driver programmers manual [10].

Hence, the incoming measurement signals can be identified as the time-dependent vectors given by $s_{DL}(i) = (\mathit{rate}, \mathit{status}, \mathit{bytes}, \mathit{lost\_segment}, \mathit{lost\_fragment}, \mathit{link}, \mathit{noise})$ and $s_{UL}(i) = (\mathit{rate}, \mathit{status}, \mathit{bytes})$. Additional measurements were available, but did not very well characterize the conditions of the link. For instance, a high inter-packet-arrival or -transmission time does not necessarily indicate a fading condition since also low traffic on the network introduces this pattern to the link layer metric. Also, the signal strength turned out to be an unreliable indicator of link quality during the presence of a high level of noise.

Although new samples are needed upon reception and transmission of each individual frame to assign them to different links, neither the FW nor the driver maintained this information. An additional FW emulation module in the Linux kernel space was developed to take these updates on a per-frame basis and supply the value in a virtual register. The modified driver then uses these samples to assigned them to a per-link diagnosis.

In addition to the basic functionality described above, a logging interface was developed that allows overwriting these samples with values from a trace file. Thus the testbed provides an emulation platform for future experiments.

4.2 Filtering

For this example implementation on IEEE 802.11b the following link layer metrics were derived from the samples by the means of simple filters: SNIR[2], Retransmission Ratio (RR), Frame Error Rate (FER), and bitrate (rate). These are defined by the following formulas:

[2]The *noise* samples include interference power as well.

$$\begin{aligned}
\mathit{rate}(k) &= \mathit{rate}(i) \\
\mathit{RR}(k) &= \frac{\sum_{i=0}^{n} \mathit{ISRETRANS}(\mathit{status}(i))}{\sum_{i=0}^{n} \mathit{ISOK}(\mathit{status}(i))} \\
\mathit{FER}(k) &= \frac{\sum_{i=0}^{n} \mathit{ISFCSERR}(\mathit{status}(i))}{\sum_{i=0}^{n} \mathit{ISOK}(\mathit{status}(i))} \\
\mathit{SNIR}(k) &= \frac{\mathit{level}}{\mathit{noise}}
\end{aligned}$$

Additionally, the link layer Bandwidth (BW), is frequently measured as the amount of bytes received within the sample period Δt:

$$\mathit{BW}(k) = 8 \cdot \frac{\sum_{i=0}^{n} \mathit{bytes}(i)}{\Delta t}.$$

4.3 Link Diagnosis

The quality of a link is intentionally not modeled with states, because link level measurements have shown that links with intermediate loss rates and no sharp transition between high and low error rate are quiet common [3]. Additionally, asymmetric link behavior has been observed emphasizing the need for a direction sense, modeled by the order of the two MAC addresses of the peers [4].

In order to provide indications for both directions, different link layer metrics are used for uplink and downlink. The robustness of the link indications as outlined by the IAB guidelines, is also taken into account. For instance in the downlink, the $\mathit{SNIR}(k)$ diagnosis takes the persistence of the link quality change into account to smoothen out fading spikes. For the FER no validation of the mechanism could be worked out, since no significant FER was encountered during the indoor experiments, as also reported in [7].

Basically, the link_quality_changed trigger is fired upon a fulfilled condition based on the RR, SNIR, or FER as expressed by the following formula:

$$l2(t) = \begin{cases} 0, & \\ 1, & [(1-\alpha) \cdot \mathit{RR}_{\mathrm{ref}} < \mathit{RR}(k) < \mathit{RR}_{\mathrm{ref}} \cdot (1+\alpha)] \vee \\ & [((1-\alpha) \cdot \mathit{SNIR}_{\mathrm{ref}} < \mathit{SNIR}(k) \cdot (1+\alpha)) \wedge \\ & (t - t_{\mathrm{last\_SNIR\_downshift}} > t_0)] \vee \\ & [\mathit{rate}(k) \neq \mathit{rate}_{\mathrm{ref}} \wedge (t - t_{\mathrm{last\_rate\_shift}}) > t_0] \vee \\ & [(\mathit{FER} < \frac{\mathit{FER}_{\mathrm{ref}} \cdot \alpha}{100}) \wedge (\mathit{FER} > \frac{\mathit{FER}_{ref} \cdot \alpha}{100})] \end{cases}$$

The used parameters for link diagnosis, namely the significance α, sample count c and diagnosis period t_0 are set by CS. Some meaningful measurements about the number of lost segments and lost fragments turned out to be poorly supported by the FW.

The link_up and link_down triggers are mapped to the Prism2 specific FW states in figure 4.

How well the diagnosis integrates with the classic internal wlan-ng driver structure is shown in figure 3. The internal division of work encapsulates the register I/O to the hfa384x

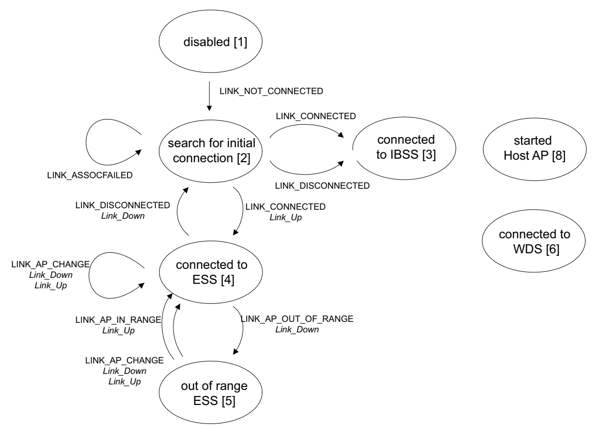

Figure 4: How Prism2 firmware states map to the link_up and link_down trigger.

and assigns the link diagnosis to the prism2sta component. Both are implemented in the prism2_cs module used by the p80211 module propagating the L2 triggers via the OS specific wireless API. The L2 triggers are forwarded via the in-kernel API of the wireless extensions [13]. The kernel-to-user space communication is realized by the means of RTnetlink sockets. User space applications then link against the iwlib library to be able to listen on the appropriate wireless extensions channel.

4.4 Connection Analysis

Figure 5 shows a simplified message sequence chart (MSC) of the overall communication within the system for the example of a varying link quality. In the figure the link oriented addresses (MAC addresses) are highlighted in blue, whereas the connected oriented addresses (IP addresses) are marked red. The Connectivity Support (CS) entity maps the L2 trigger to a connection based on pre-registered service requests. These requests include an indication of the application type and an appropriate threshold for the application, such as the bandwidth.

The lack of a fully implemented Connection Manager on Linux was solved with a Connection Manager stub implementing the used features only. Inter Process Communication (IPC) between CS and application was realized with domain sockets.

4.5 Application Notification and Adaptive Stream Management (ASM)

As a demonstrator application using the notification hierarchy the RealPlayer was modified. The standard version uses the ASM technology to adapt the video stream to changes in the connectivity [12]. It defines an application layer protocol based on ASM rules. These rules are exposed by the streaming server before a stream is setup and define the different pre-coded video streams ready for download. They are commonly using bandwidth and latency indications, but also custom ASM rules can be defined using the video frame error rate or other parameters to characterize the requirements of the stream. The RealPlayer uses fast-buffering techniques to estimate the bandwidth and to select a suitable stream at startup. In case the network quality changes and does not meet the requirements of the

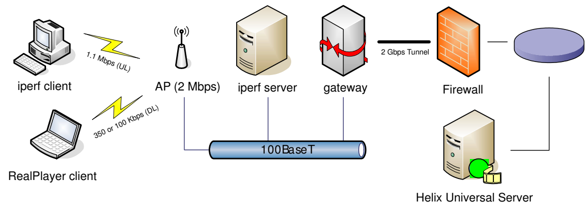

Figure 6: Demonstration Setup

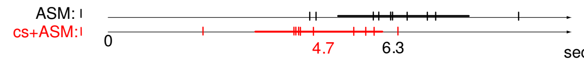

Figure 7: Margin for ASM stream switching advance.

currently subscribed ASM rule any more, RTSP messages are exchanged with the server to switch to another stream.

The modified RealPlayer version is using the provided application notifications as another source for stream switching.

5. RESULTS

In order to demonstrate the usability of the notification hierarchy, it was tested with the adaptive video streaming scenario illustrated in figure 6. Two nodes were sharing the same access point (AP). While the already described modified RealPlayer was streaming a video, the second node was uploading data, causing cross-traffic and frame collisions on the wireless link. Using the common stream switching scheme, which is based on transport layer measurements, a significant delay was experienced before the stream was downshifted. Taking also the L2 trigger information based on the RR into account, the detection time of a significantly reduced bandwidth could be reduced by approximately 25% as shown in figure 7.

The timing advance can also be seen in figure 8. Here, the link bandwidth was taken as a base for an estimation of the stream bandwidth, which implies that there is only a single video stream and no other applications using the same link from the mobile node.

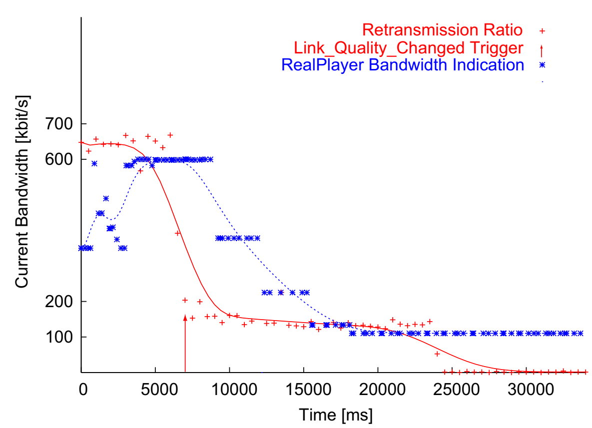

Figure 8: Timing advance for ASM with L2 trigger based on RR.

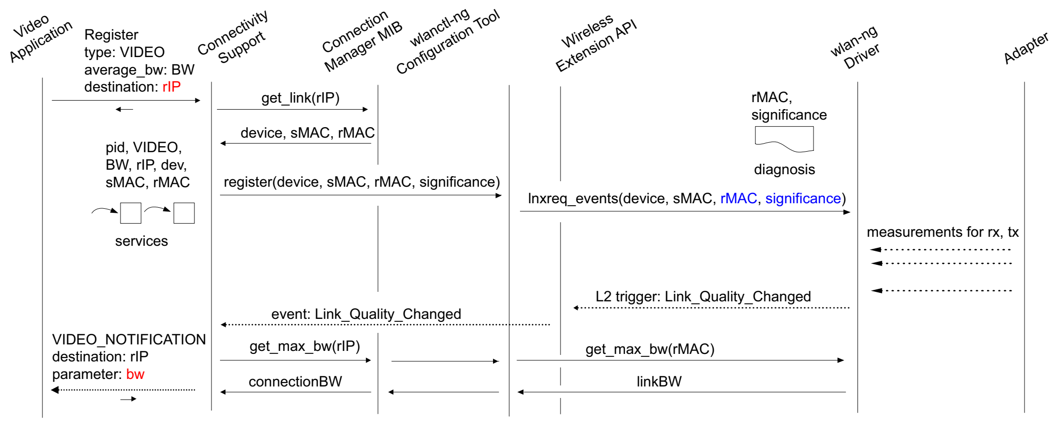

Figure 5: MSC for the example of a varying link quality.

6. CONCLUSIONS AND FUTURE WORK

The presented approach supports network aware applications with unified link layer triggers, while requiring only minor changes to the applications. A notification hierarchy was presented, which is compliant with the IETF requirements on L2 triggers in terms of reliability and robustness. Efficiency is achieved by parameterized link specific diagnosis. In our future work we plan to evaluate an alternative architecture for link aware applications, such as the Connection Managers or configuration tools. They demand the possibility of the registration of link specific thresholds, as for instance "average SNIR drops below 54dB." The benefit would be a less complex driver freed from link diagnosis. However, the challenges for rapid notifications in this case are the high frequent polling for the actual measurements from the driver and the specification of a powerful API to accommodate for complex queries.

7. ACKNOWLEDGMENTS

We would like to thank DFG (Deutsche Forschungsgemeinschaft), European Union (the IST GOLLUM-project) and RWTH Aachen University for financial support. We would also like to thank all the members in the GOLLUM research team, and especially Andreas Ibing and Dr. Alain Gefflaut for enlightening discussions.

8. REFERENCES

[1] B. Aboba. Architectural implications of link indications. IETF Informational Internet Draft http://www.ietf.org/internet-drafts/draft-iab-link-indications-03.txt [Visited Sept. 24, 2005], August 2005.

[2] Absolute Value Systems Inc. (AVS). wlan-ng driver implementation. http://www.linux-wlan.org [Visited Sept. 24, 2005].

[3] D. Aguayo, J. Bicket, S. Biswas, G. Judd, and R. Morris. Link-level measurements from an 802.11b mesh network. In *Proc. ACM SIGCOMM Portland, Oregon*, pages 121–132, August/Sept. 2004.

[4] A. Aguiar and J. Klaue. Bi-directional wlan channel measurements in different mobility scenarios. In *Proc. of Vehicular Technology Conference (VTC Spring), Milan, Italy*, pages 64–68, May 2004.

[5] P. Bertin, T. Noel, and N. Montavont. Parameters for link hints. IETF Internet Draft of DNA Network Working Group, http://ftp.shlink.de/dokumente/internet-drafts/draft-bertin-hints-params-00.txt, expired February 2004 [Visited Sept. 24, 2005], August 2003.

[6] L. Dimopoulou, G. Leoleis, and I. S. Venieris. Fast handover support in a wlan environment: Challenges and perspectives. *IEEE Network*, 19(3):14–20, May 2005.

[7] D. Eckhardt and P. Steenkiste. Measurement and analysis of the error characteristics of an in-building wireless network. In *Proc. ACM SIGCOMM, Stanford, CA*, pages 243–254, October 1996.

[8] A. Festag and A. Wolisz. Optimization of handover performance by link layer triggers in IP-based networks. *Technical Report, Telecommunication Networks Group, Technische Universität Berlin*, TKN-02-014, July 2002.

[9] V. Gupta and D. Johnston. A generalized model for link layer triggers. http://www.ieee802.org/handoff/march04_meeting_docs/Generalized_triggers-02.pdf [Visited Sept. 24, 2005], March 2004.

[10] Intersil Corp. Prism driver programmers manual rm025.3, version 3.0. http://calradio.calit2.net/calradio1/rfboard/datasheets/RM025-prism-dpm-ver3d0.pdf [Visited Sept. 24, 2005], July 2003.

[11] K. Mitani, R. Shibui, and F. Teraoka. Low latency handoffs in mobile IPv4. IETF Internet Draft in WG MIPv4, http://ietfreport.isoc.org/all-ids/draft-koki-mobopts-l2-abstractions-02.txt [Visited Sept. 24, 2005], February 2005.

[12] Real Networks Inc. Helix client and server software developer's guide, chapter 11: Adaptive stream management. SDK Version R5, https://common.helixcommunity.org [Visisted Sept. 24, 2005], December 2003.

[13] J. Tourrilhes. Wireless extensions for linux. http://www.hpl.hp.com/personal/Jean_Tourrilhes/Linux/Linux.Wireless.Extensions.html [Visited Sept. 24, 2005], January 1997.

[14] J. Tourrilhes and C. Carter. P-handoff: A protocol for fine grained peer-to-peer vertical handoff. HP Lab Report HPL-2002-61, 2002.

[15] A. E. Yegin. Link-layer triggers protocol. IETF Informational Internet Draft http://yegin.org/alper/draft-yegin-l2-triggers-00.txt, expired December 2002 [Visited Sept. 24, 2005], June 2002.