

## Weekly Report (23.5 – 27.5)

### Ingenieurpraxis

**Phase 1:** Literature research, evaluation of configurable router alternatives  
(week 1/1)

Literature research focused on 3 main areas: “General”, “Keyframe detection” and “Frame selection”. Most literature that includes frame dropping in any way treats it in a very glancing manner, focusing rather on other aspects of the proposed algorithm. Frame dropping is relatively common in time-sensitive video applications where the client’s bandwidth is constrained. Server-side dropping of low priority frames then lower the load on the client-side.

9.	is deadline of frame $i$ violated?
10.	For $j = i_0 + 1$ to $i$
11.	Compute the gain $\Delta_j^i$
12.	Choose frame $k$ with largest gain $\max \Delta_i$
13.	Discard frame $k$ and include frame $i$ , i.e.,
14.	$S^\# := (S^\# \cup \{i\}) \setminus \{k\}$
15.	Update buffer occupancy at $B_i$ , i.e.,
16.	$B_i := B_{i-1} + C + \max \Delta_i - f_i$

Notice line 12. & 13.

I did a bit of digging in the Keyframe detection side, thinking that maybe an implementation of the algorithm would be within the scope of this work, but with the dummy data being clearly marked, implementation of an independent algorithm for the raw frames does not seem to be necessary. It is good, however, to know there is a wealth of research out there that may have some answers for problems that may come up in the future.

Additionally, I did some reading into Click itself and associated alternatives. See below.

Regarding Click vs. Alternatives

Alternatives seem to be relatively scarce. A paper I found listed a couple of alternatives to click: Ensemble, x-kernel and Scout.

Ensemble I haven't even been able to find unequivocally. The name doesn't really help, with multiple companies providing video or networking services with exactly the same moniker.

x-kernel (<http://www.cs.arizona.edu/projects/xkernel/>) seems to be a project mostly performed by researchers from the University of Arizona. Its an "Object-based framework for implementing network protocols", so mostly aimed at TCP/IP, which means it lacks the stated flexibility that Click does, at least at a glance.

I can't find a single reference or paper in their page that is post-2000. This is not to say of course that Click is the bloodiest of edges, with its last release in 2011, but at least it has been active during this Millennium.

X-kernel seems unviable.

A spinoff of x-kernel, Scout (<http://www.cs.arizona.edu/projects/scout/>) is a "communication-oriented operating system targeted at network appliances". It suffers from the same problem of being woefully outdated, with not one publication this side of Y2K. The fact that it's designed as its own OS also seems like it doesn't really adhere to our specifications.

Its website also features and impressive amount of dead or broken links, which doesn't bode well for any kind of support.

Scout is also unviable.

Another paper ("AiroLAB: Leveraging on Virtualization to Introduce Controlled Experimentation in Operational Multi-hop Wireless Networks", Doriguzzi et al, 2010) considered using OpenVZ (<https://openvz.org/>), a "modified Linux kernel tree that supports virtualization, isolation and resource management (...)", but found it too limiting compared to Click because of the "impossibility to run customized kernel images in different slivers".

OpenVZ is mostly used for virtualization of Linux containers.

"Click vs. Linux: Two Efficient Open-Source IP Network Stacks for Software Routers", (Bianco et al, 2005) aims to prove the "feasability of building a high performance IP router (...)" with the use of IP Network stacks and compares Click favorably in terms of reception/transmission (bit and packet) rate, among other measures, in a series of evaluations against the standard Linux network stack implementation. It especially attributes the differences to elements such as better buffer management in Click.

In the end, Click seems to be the best option for the specified task. Next week will be focused on familiarizing myself with Click and getting the basics down for the work that is yet to come.

Arturo Buitrago Méndez

**Consulted literature:**

- Bachhuber, C., Steinbach, E., Universit, T., & Reisslein, M. (n.d.). Low Delay Video Transmission Strategies Chair for Media Technology.
- Bianco, A., Birke, R., Bolognesi, D., Finochietto, J. M., Galante, G., Mellia, M., ... Neri, F. (2005). Click vs. Linux: two efficient open-source IP network stacks for software routers. HPSR. 2005 Workshop on High Performance Switching and Routing, 2005., 18–23. <http://doi.org/10.1109/HPSR.2005.1503186>
- Braem, B., & Voorhaen, M. (2011). Click Concepts Click Modular Router Concepts and Philosophy October 2011. October, (October), 1–25.
- Chebrolu, K., & Rao, R. R. (2004). Selective frame discard for interactive video. Communications, 2004 IEEE, 7(c), 4097–4102. <http://doi.org/10.1109/ICC.2004.1313320>
- Doriguzzi Corin, R., Riggio, R., Miorandi, D., & Salvadori, E. (2011). AiroLAB: Leveraging on virtualization to introduce controlled experimentation in operational multi-hop wireless networks. Lecture Notes of the Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering, LNICST, 46, 331–344. [http://doi.org/10.1007/978-3-642-17851-1\\_27](http://doi.org/10.1007/978-3-642-17851-1_27)
- Hemy, M., Hengartner, U., Steenkiste, P., & Gross, T. (n.d.). MPEG System Streams in Best-Effort Networks, 1–11.
- Kohler, E., Morris, R., Chen, B., Jannotti, J., & Kaashoek, M. F. (2000). The click modular router. ACM Transactions on Computer Systems, 18(212), 263–297. <http://doi.org/10.1145/354871.354874>
- Liu, T., & Choudary, C. (2004). Real-time content analysis and adaptive transmission of lecture videos for mobile applications. ACM Multimedia 2004 - Proceedings of the 12th ACM International Conference on Multimedia, 400–403. <http://doi.org/10.1145/1027527.1027622>
- Usach-Molina, P., Sastre, J., Naranjo, V., Vergara, L., & Muñoz, J. M. L. (2010). Content-based dynamic threshold method for real-time keyframe selecting. IEEE Transactions on Circuits and Systems for Video Technology, 20(7), 982–993. <http://doi.org/10.1109/TCSVT.2010.2051276>
- Youm, S., & Kim, W. (2003). Dynamic threshold method for scene change detection. Proceedings - IEEE International Conference on Multimedia and Expo, 2, II337–II340. <http://doi.org/10.1109/ICME.2003.1221622>
- Zhang, Z.-L., Nelakuditi, S., Aggarwal, R., & Tsang, R. P. (1999). Efficient Selective Frame Discard Algorithms For Stored Video Delivery Across Resource Constrained Networks. Proceedings of the IEEE International Conference on Computer Communications (INFOCOM), 472–479.