

# Entwicklung einer Messmethode für die Verarbeitungszeit von UI-Frameworks

Thema:

Entwicklung einer Messmethode für die Verarbeitungszeit von UI-Frameworks

Art:

MA

Betreuender:

Andreas Schmid

Erstgutachter/in:

Raphael Wimmer

Status:

Entwurf

Stichworte:

Latenz, Graphik, Framework

angelegt:

2019-08-04

## Hintergrund

Bei der Interaktion mit Computersystemen spielt die Latenz zwischen Ein- und Ausgabe eine wichtige Rolle, da sich zu lange Wartezeiten auf Performanz und User Experience auswirken können [1]. Die Gesamtlatenz (Ende zu Ende) eines Systems setzt sich aus mehreren Teillatenzen bei Eingabe, Verarbeitung und Ausgabe zusammen [2]. Um die Gesamtlatenz eines Systems messen, verstehen und schließlich verbessern zu können, ist es sinnvoll, die Teillatenzen einzeln zu betrachten.

Mittlerweile existieren zahlreiche UI-Frameworks, die Funktionalität zur Ein- und Ausgabe abstrahieren und somit das Erstellen von graphischen Anwendungen erleichtern. Diese Frameworks reichen von hardwarenahen Programmierschnittstellen wie OpenGL [3], GUI-Toolkits wie Qt [4], 3D-Engines wie Unity [5] bis hin zu browserbasierten Ansätzen. Mit der Programmiererfreundlichkeit des UI-Frameworks steigt in der Regel auch dessen Abstraktionsgrad, was auch einen Effekt auf die Verarbeitungszeit des Frameworks und somit die Gesamtlatenz des Systems haben kann.

## Zielsetzung der Arbeit

Da bei der Auswahl eines UI-Frameworks auch dessen Latenz eine Rolle spielt, soll in dieser Arbeit eine Messmethode entwickelt werden, mit der die Verarbeitungszeit von UI-Frameworks gemessen und verglichen werden kann. Dazu ist es wichtig, den Renderingprozess eines Systems [6] zu verstehen, um die korrekten Messpunkte zu bestimmen und Störfaktoren auszuschließen [7]. In dieser Arbeit sollen zuerst die Architekturen gängiger UI-Frameworks untersucht und verglichen werden. Anschließend soll ein Mess-Framework implementiert werden, mit dem die Latenz von UI-Frameworks reproduzierbar gemessen und verglichen werden kann. Damit soll dann eine Auswahl an verbreiteten UI-Frameworks getestet werden. Die Ergebnisse sollen dokumentiert und eingeordnet werden. Zielplattform des Mess-Frameworks ist Linux, optional auch zusätzlich andere Betriebssysteme

## Konkrete Aufgaben

- Aufbereiten des Themas (Literaturrecherche) (2 Wochen)
- Einarbeiten in die Rendering-Pipeline von Linux (2 Wochen)
- Aufstellen eines Überblicks über verschiedene UI-Frameworks (2 Wochen)
- Implementieren einer Anwendung zum Messen der Latenz von UI-Frameworks (4 Wochen)
- Implementieren von Testanwendungen mit verschiedenen UI-Frameworks (2 Wochen)
- Messen der Latenz von UI-Frameworks (2 Wochen)
- Auswertung der Ergebnisse (2 Wochen)
- Fertigstellen der schriftlichen Ausarbeitung (4 Wochen)

## Erwartete Vorkenntnisse

- grundlegende Linuxkenntnisse
- gute Programmierkenntnisse in C/C++
- technisches Interesse und sorgfältige Arbeitsweise werden vorausgesetzt

## Weiterführende Quellen

[1] I. Scott MacKenzie and Colin Ware. "Lag as a Determinant of Human Performance in Interactive Systems." Accessed February 23, 2019. <https://www.yorku.ca/mack/CHI93b.html>.

[2] Raphael Wimmer, Andreas Schmid, and Florian Bockes. "On the Latency of USB-Connected Input Devices." In Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems, 420:1–420:12. CHI '19. New York, NY, USA: ACM, 2019. <https://doi.org/10.1145/3290605.3300650>.

[3] <https://www.opengl.org/>

[4] <https://www.qt.io/>

[5] <https://unity.com/>

[6] <https://blogs.igalia.com/itoral/2014/07/29/a-brief-introduction-to-the-linux-graphics-stack/>

[7] <https://easyperf.net/blog/2019/08/02/Perf-measurement-environment-on-Linux>

From:  
<https://wiki.mi.ur.de/> - **MI Wiki**

Permanent link:  
[https://wiki.mi.ur.de/arbeiten/framework\\_latency?rev=1564999762](https://wiki.mi.ur.de/arbeiten/framework_latency?rev=1564999762)

Last update: **05.08.2019 10:09**

