|  |
| --- |
| FH Joanneum |
| **Screensharing** |
| Seminararbeit Heterogene Systeme |

|  |
| --- |
| Kevin Pfeifer  [Datum] |

Inhalt

[1. Abstract 3](#_Toc442289908)

[2. Einleitung 4](#_Toc442289909)

[3. Airplay 5](#_Toc442289910)

[4. Intel WIDI 6](#_Toc442289911)

[5. AMD AWD 7](#_Toc442289912)

[6. Miracast 8](#_Toc442289913)

[7. Chromecast, Firetv 9](#_Toc442289914)

[8. Fazit 10](#_Toc442289915)

# Abstract

**Entweder den abgegebenen oder nach der seminararbeit neu formulieren**

**Und links 3,5 cm abstand, sonst 3 - erst am schluss machen für verlängernden effekt ;)**

Im Rahmen dieser Seminararbeit werden verschiedene Screenshare-Techniken behandelt und verglichen. Der Vergleich erfolgt zum einen nach der zu Grunde liegenden Technologie und Funktionsweise aber auch in Bezug auf die Interoperabilität zwischen verschiedenen Geräten und Plattformen. Weiters werden die Punkte Kosten und Einsatzgebiete behandelt.

Es soll ermittelt werden welche Software nur auf einer einheitlichen Plattform läuft, welche auf verschiedenen Plattformen laufen und welche Einschränkungen es dabei gibt (Performance, Funktionalität, Sicherheit, Workarounds). Dabei werden Informationen aus den einzelnen Spezifikationen, Erfahrungsberichten sowie Tests gewonnen.

Zusätzlich soll speziell für mobile Plattformen (Android und iOS) ermittelt werden welche Möglichkeiten zum Screenshare auf Desktop-, Mobile- sowie TV-Geräte vorhanden sind. Spezielles Augenmerk liegt dabei auf der Performance sowie den Einschränkungen der einzelnen Methoden auf mobilen Plattformen.

Diese Arbeit liefert eine Zusammenfassung und Übersicht über alle bearbeiteten Technologien, deren Vor- und Nachteile sowie deren effektiven Nutzen.

# Einleitung

Allgemeiner bullshit

Warum wir des thema behandeln

Welche software wir genau untersuchen

Auf was wir sie untersuchen

Was wir davon eigentlich wollen (gscheider vergleich, sinnvolle software)

Im prinzip eine erweiterung und präzisierung des abgegebenen abstracts

# AirPlay

Proprietäres protokoll für apple geräte

Zwischen mac, iphone, ipad etc.. für tv: spezielle set-top box

Airserver: für windows.. kann airplay und miracast streamen

AirPlay ist ein speziell von Apple entwickeltest Protokoll, welches das streamen von Audio und Video Daten zwischen unterstützten Geräten ermöglicht. Im Vergleich zu Bluetooth hat AirPlay eine weitaus höhere Datenübertragungsgeschwindigkeit (120 MBit/s) welche bei verlustfreien Audio- und Videoformaten benötigt werden. AirPlay löste 2010 das Vorgängerprojekt AirTunes ab. (GRAVIS 2011; Hoffman 2013)



Abbildung X: iPad und iPhone AirPlay Auswahl (Apple 2016)

Prinzipiell ist es ohne weitere Zusatzsoftware mögliche von allen mobilen Apple Geräten zu einem Apple TV oder eine AirPlay fähigen Gerät Videos bzw. Audio zu senden. Dies geschieht entweder über ein LAN oder über eine P2P-Verbindung. Lautsprechersysteme oder Fernsehapparate können diese AirPlay-Funktionalität auch direkt implementiert haben um kein weiteres Gerät, wie einen AirPort Express o.Ä. kaufen zu müssen. (Apple 2016)

Andere Betriebssysteme wie Windows müssen Zusatzsoftware wie 5KPlayer installieren um auf AirPlay-fähigen Geräten zu streamen oder als AirPlay-Empfänger zu dienen.

Durch zusätzliche Applikationen wie AirFloat können auch mobile Apple Geräte als AirPlay Empfänger dienen.

# Intel WIDI

## Ursprünge

Intels Wireless Display (WiDi) wurde ursprünglich entwickelt, um den Bildschirm eines Notebooks beziehungsweise einzelne Inhalte kabellos zu einem Monitor oder Projektor zu übertragen. Die erste Version wurde 2010 vorgestellt und zielte vor allem darauf ab, bei Präsentationen (Powerpoint, Slideshows) eingesetzt zu werden (vgl. Müssig 2010).

Wie sich bereits aus dem Namen erschließen lässt, wurde diese Technologie von Intel entwickelt. Um WiDi nutzen zu können wurde ursprünglich eine CPU der Arrendale Generation mit integrierter Grafikeinheit sowie ein Intel Centrino 6000 Wlan Modul benötigt um WiDi nutzen zu können. Intel integrierte nur in diese Treiber die entsprechenden Funktionen für WiDi. Durch diese Einschränkungen war es nicht möglich WiDi zu nutzen, wenn der Sender andere Prozessoren beziehungsweise Grafikeinheiten wie Nvidia oder AMD nutzte (vgl. Müssig 2010).

Um Daten per WiDi zu empfangen war eine eigene Set-Top Box notwendig. Diese dekodierte das per WiDi gesendete Signal und leitete dies per Kabel an das Endgerät weiter.

Die Probleme dieser ersten Versionen von WIDI waren zahlreich. Zum einen war die Anzahl an WIDI fähigen Sendern sehr begrenzt und zum anderen gab es nur wenige Empfänger, die zudem auch noch sehr teuer (damals ca. 100USD) waren. Zusätzlich war WiDi nur für Windows 7 und darüber verfügbar. Weiterhin war es nicht möglich, kopiergeschützte Daten wie Filme von BluRays zu übertragen, da WiDi High-Bandwith Digital Content Protection (HDCP) nicht unterstützte. Die Übertragungsqualität war außerdem auf 720p und Stereo Sound beschränkt. Zudem entstand bei der Nutzung von WiDi eine Latenzzeit im Bereich von mehreren Sekunden. Die Latenzzeit entspricht der verzögerten Darstellung beim Empfänger aufgrund der zum Übertragen benötigten Zeit. Eine hohe Latenzzeit ist vor allem bei Anwendungen welche Benutzereingaben verlangen störend (vgl. Müssig 2010; Ziesecke o.J.).

Da WiDi die Grafikeinheit der CPU nutzt, wird diese sehr stark belastet. Vor allem bei schwächeren Prozessoren (niedrige Taktrate, kleine Caches) kann dies zu Problemen führen (vgl. Paine 2014).

## Entwicklung

In den darauffolgenden Generationen wurden viele dieser Probleme beseitigt. So wurde die Latenzzeit verbessert. Dadurch wurde sowohl das Arbeiten mit WiDi vereinfacht als auch das Anzeigen von Spielen ermöglicht. Zusätzlich unterstützt WiDi ab der zweiten Generation auch HDCP, wodurch das Abspielen von kopiergeschütztem Material ermöglicht wird. Auch in Sachen Qualität wurde WiDi verbessert. Neben 5.1 Surround Sound wird auch die Übertragung von höheren Auflösungen ermöglicht (vgl. Ziesecke o.J.).

WiDi ist weiterhin nur für Windowsgeräte mit entsprechender Intel Hardware verfügbar, wobei jetzt wesentlich mehr Prozessoren und Grafikeinheiten unterstützt werden. Ob ein Gerät WiDi unterstützt kann mit einem Tool[[1]](#footnote-1) festgestellt werden.

Ab der Version 3.5 unterstützt WiDi den Screensharing Standard Miracast und gilt als eine konkrete Implementierung dessen. Ab Windows 8.1 benützt WiDi Prozesse des Betriebssystems um Miracast Verbindungen aufzubauen (vgl. Länger 2015).

Der Vorteile der Implementierung von Miracast liegt in der breiteren Unterstützung von Empfänger Geräten. Da für Miracast wesentlich mehr Empfänger verfügbar sind, fällt es den Benutzern leichter, ein passendes Empfangsgerät zu finden. Allerdings implementiert nicht jedes Gerät dieselben Aspekte von Miracast, was oft zu Inkompatibilität führt (Siehe Kapitel 6).

Zusätzlich zu normalem WiDi hat Intel auch noch eine Pro Version speziell für Business Kunden entwickelt. Diese inkludiert zusätzliche Sicherheitsfeatures sowie Kollaborationstools (vgl. Paine 2014).

Generell ist WiDi eher dafür konzipiert, einen Notebook- oder Tabletbildschirm auf einen größeren Monitor oder Beamer mit entsprechendem Receiver (integriert oder extern) zu übertragen, nicht aber um Bildschirme zwischen beispielsweise zwei Notebooks zu übertragen.

Da die Verfügbarkeit auf Windows Geräte beschränkt ist, und die Implementierung von Miracast nicht zwangsläufig Kompatibilität mit allen Miracast Receivern bedeutet, ist WiDi in Bezug auf heterogene Systeme eher ungeeignete Lösung zu betrachten.

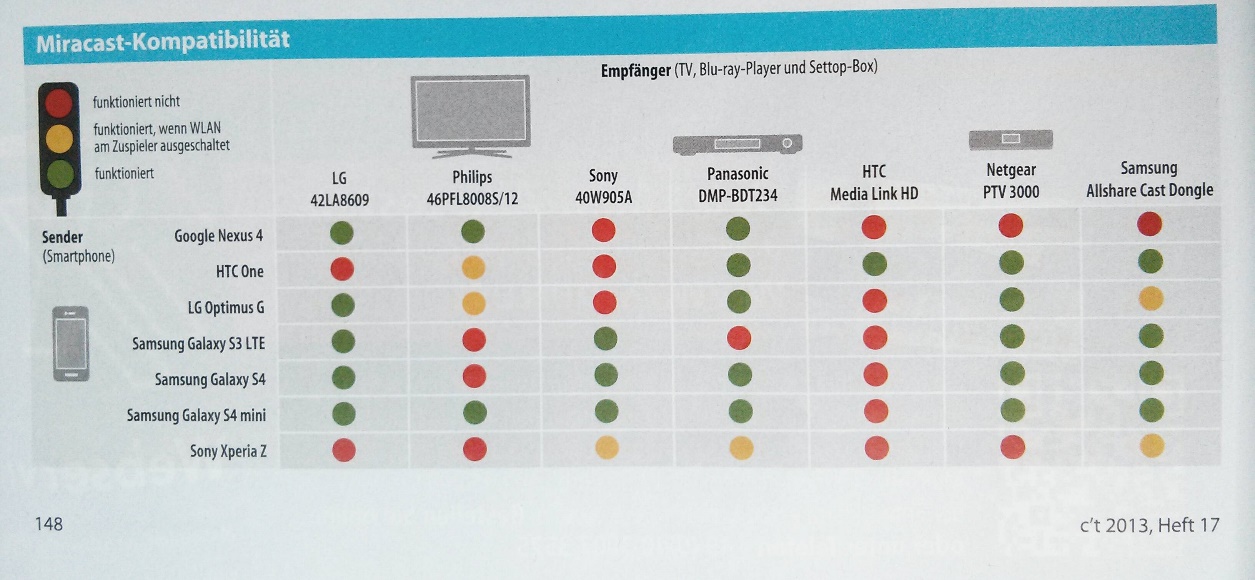
# AMD AWD

Ähnlich wie widi

Implementiert miracast standard

# Miracast

Awesome picture:



(vgl. Czerulla und Hansen 2013, S. 148)

# Chromecast, Firetv

Nicht direkt screensharing

Mit entsprechenden geräten werden diese „erweitert“

Handy als fernbedienung, etc

# Fazit

Problem: technologie noch immer unausgereift

Weiterhin implementieren unterschiedliche devices miracast auf unterschiedliche art, was zu inkompatibilität führt (siehe tests)

Für nicht smarte geräte werden zusatzdevices benötigt

Weiterhin probleme mit qualität, latenz, etc bei schwachen netzwerken

Für beste ergebnisse: try and error oder kabel verwenden :P

Literaturverzeichnis

Apple (2016) *Mit AirPlay drahtlos Inhalte von Ihrem iPhone, iPad oder iPod touch streamen* [Online]. Verfügbar auf https://​support.apple.com​/​de-at/​HT204289.

Czerulla, H. A. and Hansen, S. (2013) ‘Miracast-Check: Was die AirPlay-Alternative derzeit leistet’, *c't*, no. 17.

GRAVIS (2011) *AirPlay: 10 Fragen, 10 Antworten* [Online]. Verfügbar auf http://​www.gravis.de​/​blog/​airplay-10-fragen-10-antworten/​.

Hoffman, C. (2013) *Wireless Display Standards Explained: AirPlay, Miracast, WiDi, Chromecast, and DLNA* [Online]. Verfügbar auf http://​www.howtogeek.com​/​177145/​wireless-display-standards-explained-airplay-miracast-widi-chromecast-and-dlna/​.

Länger, K. (2015) *HDMI ohne Kabel: Miracast & Co* [Online]. Verfügbar auf http://​www.tecchannel.de​/​pc\_mobile/​peripherie/​3199755/​hdmi\_ohne\_kabel\_miracast\_co/​index2.html.

Müssig, F. (2010) *Hands-On: Intel Wireless Display* [Online]. Verfügbar auf http://​www.heise.de​/​newsticker/​meldung/​Hands-On-Intel-Wireless-Display-900035.html.

Paine, S. (2014) *WiDi – Wireless Display Overview, Specifications, Testing and Demos* [Online]. Verfügbar auf http://​www.umpcportal.com​/​2014/​02/​widi-wireless-display-overview-specifications-testing-and-demos/​.

Ziesecke, D. (o.J.) *Intel WiDi* [Online]. Verfügbar auf http://​www.voip-information.de​/​intel-widi.php.

1. <http://www.intel.de/content/www/de/de/support/emerging-technologies/000014931.html?wapkw=widi&_ga=1.144481589.1454502059.1454616028> [↑](#footnote-ref-1)