

Evaluation von Bluetooth, NFC und USB im Rahmen von PharoThings und Android als p2p Verbindung

Bachelorarbeit Sommersemester 2019

Jan Phillip Kretzschmar (*jan@2denker.de*)

Betreuer (Zweidenker GmbH):
Anton Borries (*anton.borries@2denker.de*)

Betreuer (TH Köln):
Prof. Christian Kohls

1. Juli 2019

Contents

1	Einleitung	2
2	Evaluationsziel	2
3	Evaluationsmethodik	4
3.1	Einschränkungen	5

1 Einleitung

Die Verbindungskonfiguration von IoT Geräten wurde bereits auf Basis von Wi-Fi Direct untersucht¹. Da die Implementierung dieser Technologie einige Unzulänglichkeiten besitzt, soll nun ein Vergleich von weiteren p2p Technologien vorgenommen werden. Hierunter fallen Bluetooth, NFC und USB, da diese alle von Android Smartphones nativ unterstützt werden. Da die Vor- und Nachteile sowie Funktionsweisen dieser p2p Schnittstellen bereits bekannt sind, sollen sie im Rahmen dieser Arbeit in die Verbindungskonfiguration mit PharotThings eingebunden werden, um auch hier Probleme aufzudecken und zu einer optimalen Lösung des Problems zu gelangen.

Die momentane Lösung² besteht aktuell aus Service Discovery und einer p2p Verbindung. Ob Service Discovery in seiner aktuellen Form beibehalten werden soll, ist ebenfalls zu evaluieren, da sich eine erweiterte Lösung auch nicht auf nur eine Technologie beschränken muss. Es ist daher zu überprüfen, ob eine Hybridlösung in Verbindung mit Wi-Fi Direct oder Bluetooth LE sinnvoll ist, sodass lediglich die p2p Verbindung über eine andere Technologie stattfindet.

Softwarequalität muss definiert werden.³

2 Evaluationsziel

Im vorausgegangenen Praxisprojekt wurde deutlich, dass Wi-Fi Direct einige gravierende Mängel im Hinblick auf die Funktionalität und Robustheit der bereitgestellten Bibliotheken aufweist. So ist die Dokumentation der meisten Nachrichten und Events unvollständig oder fehlt und es ist unklar wie eine Verbindung aufgebaut werden muss. Gleichzeitig sind Fehlermeldungen entweder nicht aussagekräftig oder führen dazu, dass der laufende Daemon abstürzt. Daraus lassen sich für diese Evaluation zwei Kriterien im Bereich der Robustheit ableiten. Zunächst kann getestet werden wie aussagekräftig Fehlercodes oder Fehlermeldungen sind. Das eine Ende des Spektrums bildet hierbei die Darstellung des Erfolgs eines Aufrufs durch eine boolesche Variable, auf der anderen Seite wird ein genauer Fehlercode in Verbindung mit einer Fehlermeldung ausgegeben. Die Fehlercodes sind dabei für jede genutzte Methode dokumentiert und bieten Aufschluss auf die Gründe der erhaltenen Fehlschläge. Gleichzeitig soll evaluiert werden, in wie weit sich von aufgetretenen Fehlern erholt werden kann, um dennoch eine erfolgreiche Verbindung aufbauen zu können. Hierbei wird der messbare Bereich auf der einen Seite durch den Absturz des verwendeten Moduls beschränkt. Es besteht somit keine Möglichkeit zur Erholung von Fehlern und es muss darüber hinaus sogar das verwendete Modul neu gestartet werden. Dem gegenüber steht die gute Dokumentation von Fehlercodes, welche es erlaubt, auf nicht kritische Fehler reagieren zu können. Ebenso können die Fehler bereits einen Codeblock beinhalten, der es erlaubt, den Fehler, so fern dieser unerwünscht

¹Kretzschmar: *Verbindungskonfiguration von PharotThings auf Raspberry Pi durch Android App* TH Köln Praxisprojekt Sommersemester (2019).

²<https://github.com/janphkre/iot-connectivity>

³TODO

gewesen ist, zu beseitigen. Ebenso sollte die Anbindung einer eingesetzten Technologie testbar sein. Die Testbarkeit der Software ergibt sich zum einen aus einem hohen Abstraktionsgrad, so dass sich die genutzten Technologien mit wenig Aufwand durch Mocks und Stubs auswechseln lassen. Ebenso ist eine Testbarkeit erst dann gegeben, wenn die genutzte Technologie eine hohe Stabilität im Hinblick auf Wiederholbarkeit bietet.

Die Funktionalität von Wi-Fi Direct wird aktuell durch mehrere Faktoren eingeschränkt. Zunächst mangelt es an einem internen Zustandsdiagramm des genutzten Moduls. Dadurch ist es nicht möglich, festzustellen, welche Methoden in einer bestimmten Reihenfolge aufgerufen werden müssen, um den erwünschten Zustand einer Verbindung zu erreichen. Zudem sind hierbei interne Zustandsübergänge durch ankommende Events oder weitere unbekannte Gründe nicht dokumentiert. Dies führt dazu, dass nicht klar ist, wann Methoden aufgerufen werden müssen um den aktuellen Zustand der p2p Schnittstelle beizubehalten. Als Evaluationskriterium kann hierbei wieder die Dokumentation im Hinblick auf die Funktionalität dienen, denn es kann festgestellt werden, wie gut der Verbindungszustand und Modulzustand dokumentiert sind. Es besteht zum Einen die Möglichkeit, dass keine Dokumentation vorliegt und das genutzte Modul lediglich als Blackbox genutzt werden kann. Als Optimum sollte jedoch die Dokumentation soweit vorhanden sein, dass interne sowie externe Events dokumentiert sind und gemeinsam mit einem Zustandsdiagramm sowohl der Verbindung als auch der Software dazu genutzt werden können, genau nachzuvollziehen, wann sich der aktuelle Zustand ändert und wie dieser wiederhergestellt werden kann.

Die Qualität der eingesetzten Technologie kann zudem auf Effizienz überprüft werden. Da die genutzten Module in das Gerät fest integriert sind, lässt sich Energieverbrauch nicht messen und ist außerdem von dem im Gerät konkret verbauten Chip abhängig. Es kann jedoch eine Aussage über Übertragungsraten getroffen werden, welche eine numerische Skala abbilden. Hierdurch kann festgestellt werden, in wie weit manche Anwendungsfälle mit der entsprechenden Technologie möglich sind, oder die Nutzbarkeit dadurch eingeschränkt werden. Die Konfiguration über eine REST-Schnittstelle benötigt keine hohen Datenraten zur Kommunikation, falls der Nutzer jedoch beispielsweise eine Remote-Verbindung durch Telepharo über die p2p Schnittstelle zwischen seinem persönlichen Computer und dem eingesetzten IoT-Gerät aufbauen will, benötigt er eine relativ latenzfreie und hochvolumige Datenübertragung.

Letztlich kann überprüft werden, wie leicht sich die Technologie implementieren lässt, sodass sie für das bestehende Projekt genutzt werden kann. Da die Schwierigkeit von Aufgaben im Rahmen von Softwareentwicklung immer eine Kombination aus Zeitaufwand und Komplexität sind, lässt sich hierbei immer nur eine Schätzung oder persönliche Meinung auf Basis der eigenen Präferenz angeben.

3 Evaluationsmethodik

Um die genannten Evaluationsziele überprüfen zu können, soll jede der Technologien Bluetooth (LE), NFC und USB in einem vergleichbaren Maße zu Wi-Fi Direct im Hinblick auf die Vollständigkeit ihrer Dokumentation und Nutzbarkeit ihrer Implementierung überprüft werden. Unter dem Aspekt der Softwarequalität lassen sich die Technologien gegenüberstellen. Dabei soll auf Grundlage einer beispielhaften Implementierung im Rahmen des bestehenden Projektes die Vor- und Nachteile verdeutlicht werden und zudem eine Evaluation der Komplexität und des Aufwandes einer Implementierung vorgenommen werden. Metriken, um die Qualität der Software zu messen, sind als ISO-Standard 9126 ⁴ definiert. Hierbei soll auf die folgenden Punkte eingegangen werden und in wie weit diese extern gemessen werden können, da nicht einsehbar ist, wie ausgelieferte Implementierungsartefakte intern getestet werden und dokumentiert sind.

1. *Zuverlässigkeit*: Um die Robustheit der Technologien feststellen und vergleichen zu können, muss festgestellt werden können, wie viele Fehler in der genutzten Implementierung existieren. Da dieser Punkt jedoch unter Anderem stark von der genutzten Hardware abhängt soll sich zunächst auf das Android Gerät Samsung Galaxy S7 und einen Raspberry Pi 3 B+ beschränkt werden. Gleichzeitig sollte hierbei zur Evaluation die Qualität der Dokumentation bewertet werden. Dazu zählt die Dokumentation und Verfügbarkeit des Sourcecodes im Hinblick auf die Schicht, welche als Schnittstelle bereit gestellt wird.
2. *Wartbarkeit*: Um Fehler bei der Benutzung einer der Technologien beheben zu können, ist es nötig, dass die Technologie auf der einen Seite durch Dokumentation und Quellcode leicht zu verstehen ist und gleichzeitig es erlaubt, reproduzierbar Fehler zu testen. Um hierbei eine aussagekräftige Bewertung vornehmen zu können, sollte keine Metrik wie die Häufigkeit von abweichenden Ergebnissen bei gleichen Eingabeparametern genutzt werden, da dies von nicht überschaubaren Faktoren abhängt. Es kann daher nur auf den Abstraktionsgrad der angebotenen Schnittstellen eingegangen werden, da diese für eigene Tests ersetzt werden müssen. Jedoch kann dieser Punkt auch in Verbindung mit der Zuverlässigkeit gesehen werden, da Fehler bei einer sauberen Dokumentation kein großes Hindernis für Wiederholbarkeit und damit Testbarkeit darstellen.
3. *Funktionalität*: Die Abdeckung der Implementierungen der erwarteten Funktionalität kann nur sehr eingeschränkt überprüft werden, da aus einer externen Sicht die Menge der Funktionalitäten sich auf den Verbindungsaufbau und -abbau, sowie ein Senden und Empfangen von verbindungslosen Daten beschränken und alle Technologien diese Funktionalitäten anbieten. Es muss somit darauf geachtet werden in wie weit die internen Zustände dokumentiert sind und erreicht werden können.

⁴Quelle

4. *Effizienz*: Die Effizienz der Technologien kann durch einen Vergleich von Datendurchsatzraten und Antwortzeiten festgestellt werden. Diese beiden Kennzahlen sind für eine Konfigurationsumgebung nicht kritisch, jedoch können mittels eines externen HTTP-Wrappers auch Drittprogramme eine solche p2p-Verbindung nutzen.
5. *Benutzbarkeit*: Die Benutzbarkeit der eingesetzten p2p-Lösung ergibt sich zum einen daraus, wie stark der Nutzer in die verwendete Technologie eingebunden werden muss, und zum anderen daraus, wie komplex und zeitaufwändig eine Anbindung der Technologie ist. Der Nutzer ist zum größten Teil dazu angehalten, einen Verbindungsaufbau als erwünscht zu bestätigen. Dieser Vorgang sollte so leicht wie möglich gestaltet sein und kann durch eine Gegenüberstellung der einzelnen Schritte, die der Nutzer ausführen muss, bis eine erfolgreiche Verbindung besteht, verglichen werden. Die Komplexität und der Aufwand einer Implementierung kann anhand der Menge an Methoden, die implementiert werden müssen, besonders in einer C-Bibliothek, die eine eigene Schnittstelle zu pharo abbildet, festgestellt werden. Außerdem ist ein Vergleich im Hinblick auf die Komplexität der Logik, die bei der Anbindung gehandhabt werden muss, nötig.

3.1 Einschränkungen

Um den Aufwand dieser Arbeit nicht zu sprengen, gibt es ein paar weitere Rahmenbedingungen, die im Vorfeld definiert werden sollen. Hierbei ist darauf zu achten, wenn bei einer der Technologien ein deutlich größerer Aufwand abzusehen sein sollte, die Implementierung dieser Technologie ausgeklammert werden, da es auch einer Nutzbarkeit widerspricht, eine solche Technologie trotz einem hohen Aufwand in der Implementierung zu nutzen, da dies gleichzeitig auch einen hohen Aufwand in der Wartung und im Support bedeutet.⁵ Weiterhin kann nicht auf Dokumentationen der entsprechenden Standards und Implementierungen der Technologien eingegangen werden, die nicht öffentlich zugänglich sind, da ebenso Änderungen und Erweiterungen an diesen Spezifikationen dann nicht nachvollziehbar sind.

⁵Nachteil Qualitätsmodell: Bezug zu Kosten und Nutzen [Lange Qualitätsmodelle]