



Sammanfattning

Huvudsyftet med projektet är att undersöka hur det går att genomföra decentraliserade växlingar och därefter bygga en prototyp. Detta har uppfyllts med ett system som kan växla mellan Bitcoin, Ethereum och Ethereum Classic. Däremot var syftet inte att bygga ett perfekt system. Målet var att visa att detta koncept är möjligt att använda och andra forskare kommer att ha möjligheten att vidare utforska området och uppnå målet att utveckla ett system som kan användas i produktionsmiljö.

Framtida förbättringar är att felsöka koden och ta bort buggar, förbättra portabiliteten för användargränssnittet samt att förbättra hanteringen av den decentraliserade databasen. Att arbeta med experimentella teknologier är spännande men skapar oundvikliga problem. Vi har gjort allt vi kunde för att genomföra detta projekt och med denna grund är det fullt möjligt att med mer tid bygga någonting ännu bättre.

Metod

Prototypen är uppdelad i två delar: ett användargränssnitt och ett program som sköter kommunikationen med andra decentraliserade system. Användargränssnittet ska vara snabbt, stabilt och användarvänligt men samtidigt ha en stor mängd användbara funktioner. Denna del implementeras som en webbapplikation och körs som en Electronapplikation. Större delen av användargränssnittet är implementerat i det funktionella programmeringsspråket Elm.

Den del som sköter kommunikationen med andra decentraliserade system är i sin tur uppdelad i de tre delar som beskrivs som huvudproblem i inledningen. För att genomföra en växling används så kallade *atomiska växlingar* (engelska: *atomic swaps*). Detta genomförs genom en typ av smarta kontrakt som kallas *hashade tidslåsta kontrakt* (engelska: *hashed timelock contracts*). Dessa kontrakt gör att säkerhet kan garanteras. De kontrakt som skapas i prototypen kommer att bedömas utifrån hur lång tid ett byte tar, hur mycket det kostar samt hur många par av kryptovalutor som går att växla mellan.

För att lagra information som gör att användarna kan hitta varandra och för att därefter upprätta kommunikationskanaler mellan dem används databasen OrbitDB. Det är en decentraliserad databas som bygger på ett antal andra decentraliserade tekniker. Databasen kan spara strängar och JSON-dokument, vilket är vad som behövs. Kommunikationen med själva blockkedjorna sker med Geth och Bitcoin core för Ethereum respektive Bitcoin.

***Intresserad?
Här finns koden
till prototypen:***

