

NubiSave

- Benutzerdokumentation –

Komplexpraktikum Sommersemester 2011

„Entwicklung mobiler und verteilter Systeme“

<http://141.76.40.90/~wiki/index.php/NubiSave>

Johannes Müller

Rico Tzschichholz

Steffen Matthischke

Gerd Bombach

1. Aufgabenstellung

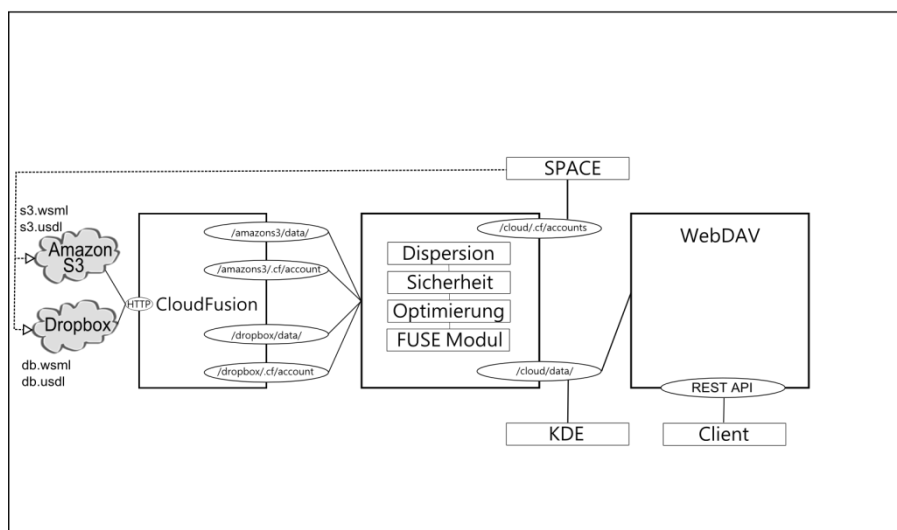
Die Aufgabe des NubiSave-Teams war die Entwicklung eines desktopintegrierten, cloudanbieter-übergreifenden Werkzeugs zur Steuerung der Datensicherung über heterogene Cloud-Storage-Dienste unter Berücksichtigung der Natur der zu sichernden Daten. Dazu musste eine Cloudabstraktions- und Auswahlkomponente und ein grafisches Frontend implementiert werden um anbieterübergreifend den passenden Storage-Dienst auswählen zu können. Durch diese Dienstplattform kann eine benutzerfreundliche Art und Weise geschaffen werden, seine Daten redundant bei verschiedenen Storage-Providern zu speichern, so dass eine erheblich größere Sicherheit bei Serverausfällen entsteht. Durch das Aufsplitten der Dateien und Verteilung bei verschiedenen Firmen ist die Gefahr geringer, dass andere Einsicht in die Daten bekommen.

Für den Benutzer bietet NubiSave den Vorteil seine in der Cloud zu sichernden Daten nicht mehr manuell zu den unterschiedlichen Storage-Providern hochladen zu müssen, und dabei noch den Überblick zu behalten, wo welche Daten gespeichert werden. Er kann seine Dateien einfach in einen virtuellen Ordner legen, welcher dann im Hintergrund die Daten redundant auf die verschiedenen Provider verteilt.

2. Vorstellung des Prototypen

Innerhalb des Sommersemester 2011 entstand der NubiSave Prototyp. Das System ist modularisiert aufgebaut, so dass es leicht mit Erweiterungen ergänzt werden kann und schnell weiter Features entwickelt und angebunden werden können. Zum Beispiel ist ein Steganographie- oder ein Verschlüsselungsmodul denkbar.

Architektur - Exemplarisch



NubiSave besteht aus einem Frontend, was es ermöglicht im System registrierte CloudStorage Anbieter nach bestimmten Quality-of-Service Parametern zu filtern und so die passenden Anbieter für seine Anforderungen zu finden. Dazu zählt zum Beispiel der Preis, den man bereit ist für einen solchen Dienst zu zahlen.

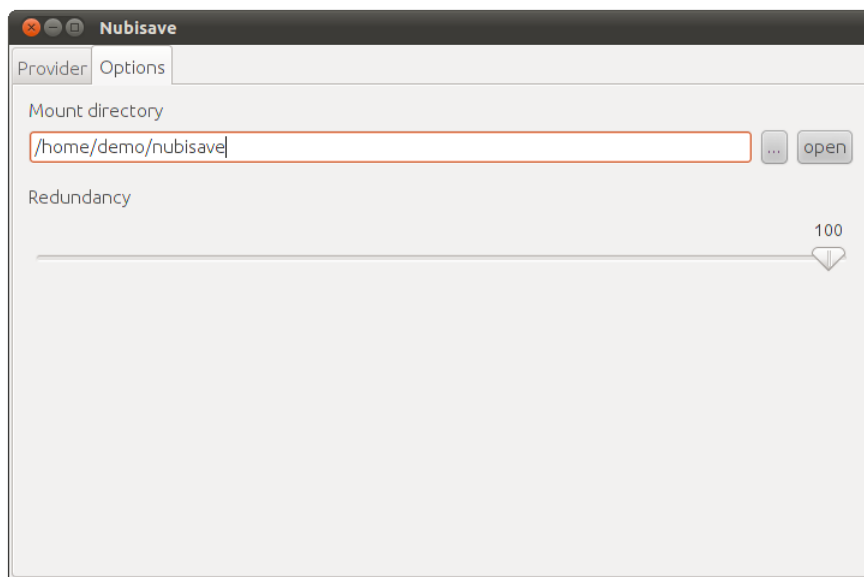
Name	P/D	P/M	A	BW	DT	RT
MS_Sky...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Amazon...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Dropbox	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
STRATO...	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Sugarsy...	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Die zu verwendeten Anbieter wählt man nach der Suche aus. Nun muss man für alle Storage-Anbieter die man verwenden möchte, einen Account besitzen, den man zuvor direkt beim Anbieter erstellt hat. Diese Zugangsdaten müssen in der Konfigurationsoberfläche NubiSave bekannt gemacht werden.

Use	Type	Description	Edit	Remove
<input checked="" type="checkbox"/>	Service	AmazonS3	User/Passwd	remove
<input checked="" type="checkbox"/>	Service	Dropbox	User/Passwd*	remove

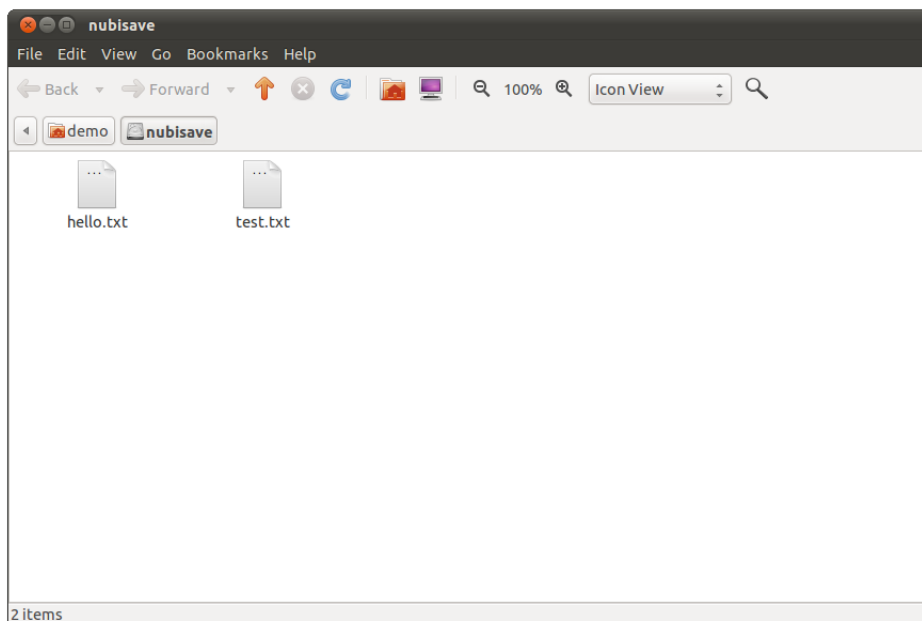
Mit einem Klick auf Apply ist NubiSave vollständig konfiguriert und startet im Hintergrund.

Ein virtueller Ordner, welcher im Options-Tab der GUI konfiguriert werden kann, wurde nun erstellt. In dem Options-Tab kann man außerdem ein Redundanz-Wert in Prozent eingeben. Je größer dieser Wert ist, desto redundanter werden die Daten auf den verschiedenen Cloud-Servern gespeichert. Das bedeutet, dass insgesamt weniger Speicherplatz für den Anwender bereit steht, aber dafür die Sicherheit höher ist, bei einem Serverausfall eines Anbieters trotzdem Zugriff auf alle Daten zu haben.



In den virtuellen Ordnern können nun einfach Dateien hinein kopiert werden, welche automatisch auf die verschiedenen Anbieter verteilt werden.

Im Hintergrund läuft das Core-Modul von NubiSave, welches sich um die Aufteilung und den Upload der Daten kümmert.



3. Installationsanleitung

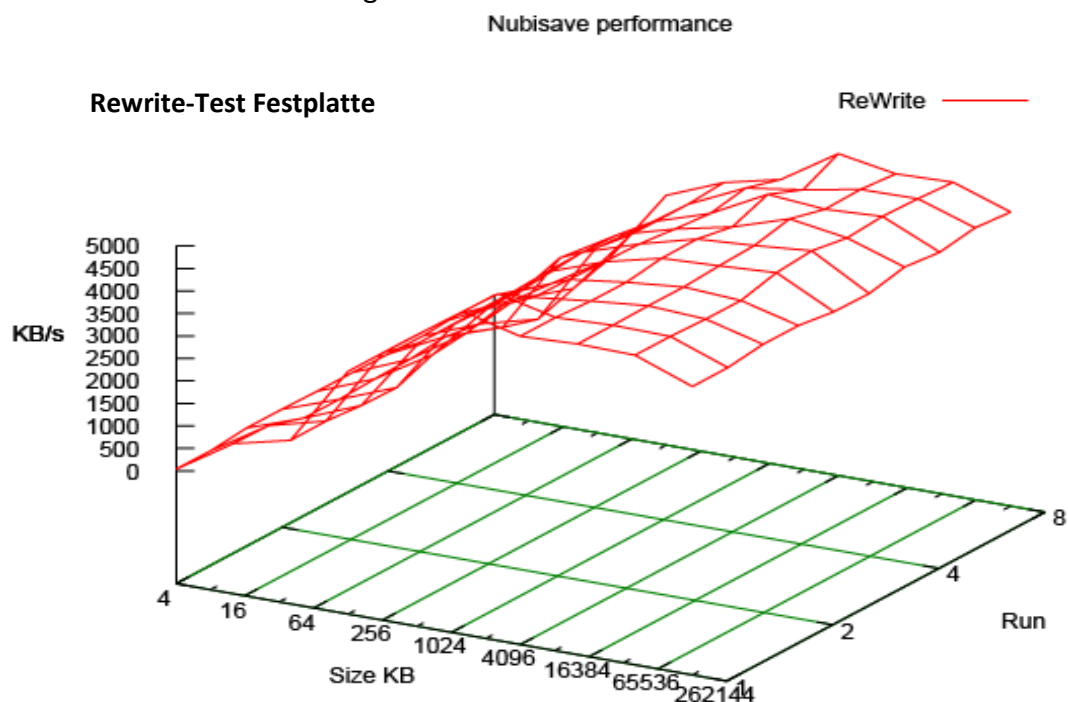
Für die Benutzung von NubiSave wird folgende Software benötigt:

- Unixoides OS
- SPACE – speziell ConQo/ContractWizard
- openJDK
- NubiSave

Alle benötigten Packages mittels des Installationsskript (installation.sh) herunterladen, NubiSave kann dann mittels der start.sh gestartet werden. Es öffnet sich das NubiSave GUI, wo man nun Storage Anbieter auswählen und die Zugangsdaten konfigurieren muss. Danach ist NubiSave einsatzfähig. Der Pfad zum erstellten virtuellen Ordner befindet sich im Options Tab.

4. Performancetests

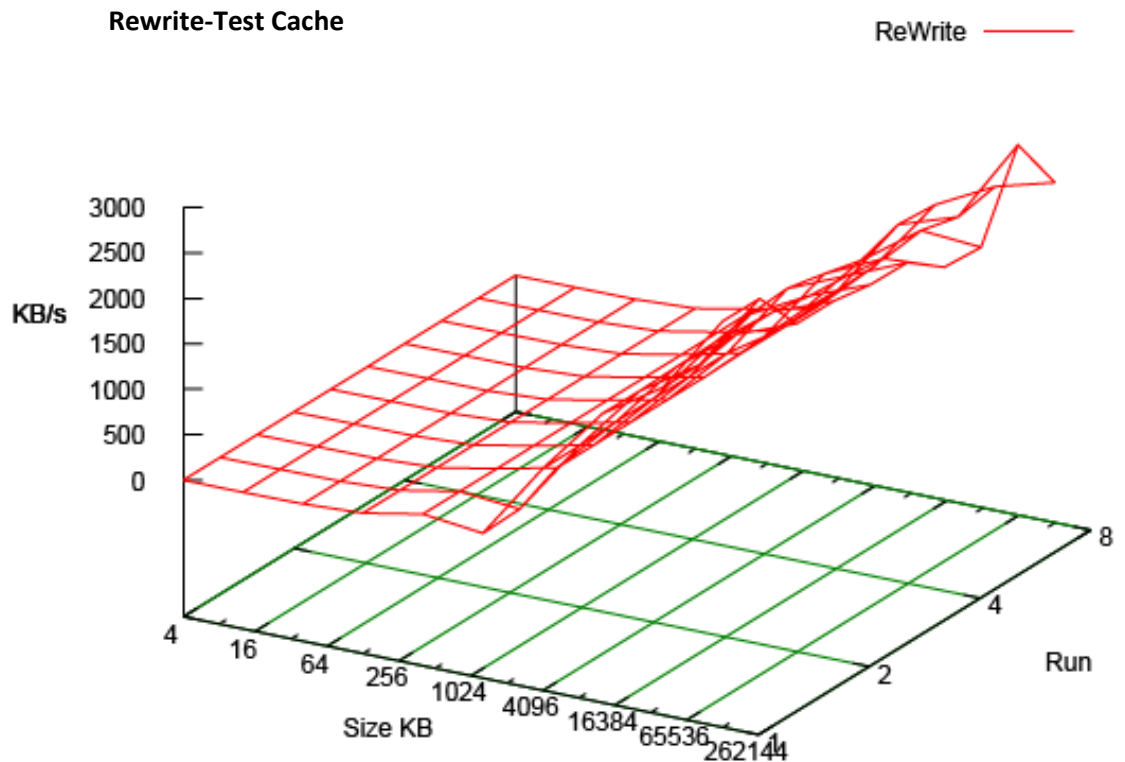
Durch das modulare System von NubiSave haben wir alle drei Bereiche separat Geschwindigkeitstests unterzogen. Als Beispiel möchte ich die Performance am Beispiel der Daten der Rewrite Tests zeigen. Ein Rewrite benötigt einen lesenden und einen schreibenden Zugriff.



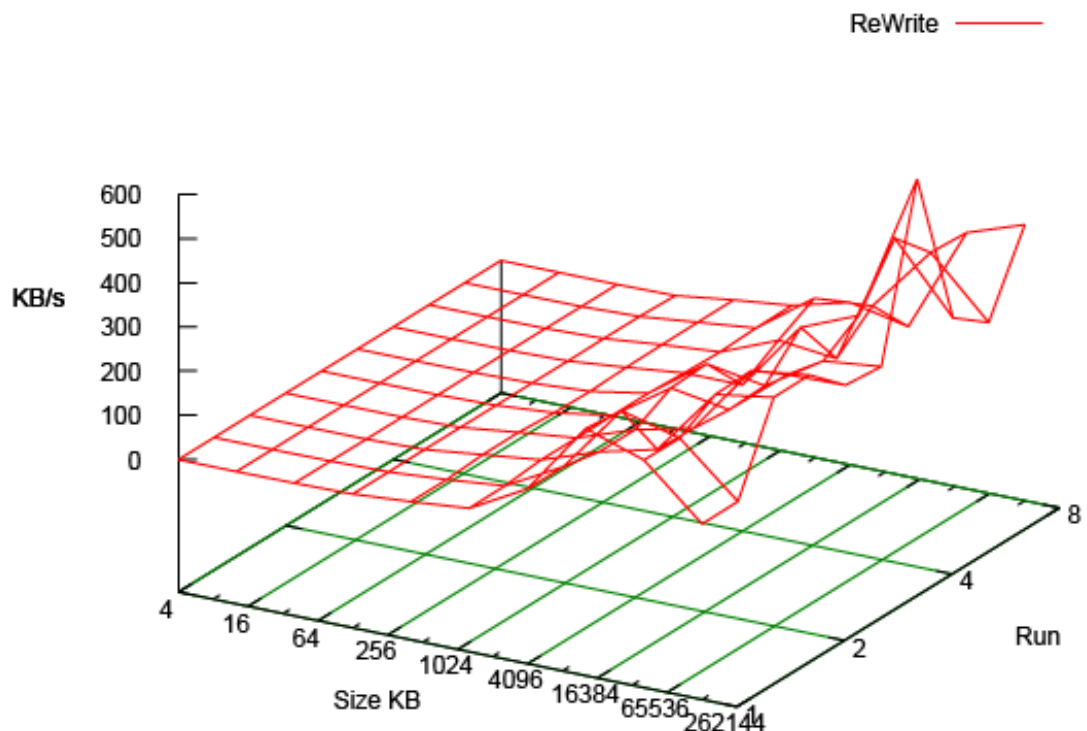
Das Diagramm zeigt die Geschwindigkeit eines wieder schreibenden Zugriffes auf die Festplatte mit unterschiedlichen Datengrößen und Anzahl an Durchläufen des Tests und davon dann jeweils die Durchschnittsgeschwindigkeit. Je größer eine einzelne Datei, desto schneller wird diese bearbeitet. Die Anzahl an Durchläufen

hat nur geringe Auswirkung auf die Geschwindigkeit.

Nubisave performance



Bevor NubiSave mit dem Upload der Daten an die Storage-Anbieter beginnt, werden die Dateien gesplittet und in einem Cache vorgehalten. Die Performance des NubiSave Core-Algorithmus kann man im Rewrite-Test des Caches sehen. Bei kleinen Dateien ist dieser relativ langsam. Erst bei Größen ab 1 MB beginnt die Geschwindigkeit auf bis zu 3000 KB/s zu erhöhen. Für einen Upload für einen Privatanwender ist dies ausreichend, da die Upload-Bandbreite seltener viel höher ist.



Letztendlich noch die Rewrite-Test Messung des Netzwerkes. Dieser zeigt den Flaschenhals einer Cloud-Storage Anwendung. Die Geschwindigkeit variiert sehr zufällig. Insgesamt ist die Geschwindigkeit relativ langsam und wird vom Storage-Anbieter und der Bandbreite des Internetzuganges reglementiert. Hier kann nur mittels Hardware-Ausbau beim Anwender und der Server der Anbieter ein Zuwachs erreicht werden.

5. Ausblick

NubiSave bietet bisher nur eine komfortable Aufteilung seiner Daten auf verschiedene Storage Anbieter an. Durch das modulare System kann dieses leicht mit mehr Funktionalität versehen werden. Ein wichtiger Punkt wäre eine direkte Vertragsaushandlung mit den Anbietern über die GUI von NubiSave. Momentan muss man bei jedem Anbieter separat sich registrieren und die Benutzerinformationen bei NubiSave speichern. Mit dem WS-Agreement Protokoll könnte so eine direkte Vereinbarung zwischen Kunden und Firma getroffen werden, und sofort in NubiSave hinterlegt und aktiviert werden. Solange Firmen eine solche Schnittstelle nicht anbieten, und dieses meist auch nicht im wirtschaftlichen Interesse der großen Storage Anbieter steht, ist eine Umsetzung nur schwer möglich und vermutlich auch nicht mit den AGBs der Firmen vereinbar.

Ein Verschlüsselungs- und Steganografie-Modul wäre Interessant um die Sicherheit seiner Daten weiter zu erhöhen. Momentan werden die Daten im Klartext und unverschlüsselt bei den Anbieter gespeichert – nur fragmentweise,

so dass auch diese Teile einer einzelnen Firma nicht viel brauchbare und vollständige Informationen liefern wird. Eine Verschlüsselung oder Hinterlegung von Informationen in Bildern kann diese Sicherheit aber nochmal um einen großen Faktor erhöhen.

Als letzter Punkt einer Entwicklungsidee könnte ein Multi-PC Support sein. NubiSave funktioniert momentan nur auf einem PC, da die Fragment Tabellen zur Wiederherstellung seiner Daten – also bei welchem Anbieter, welche Fragmente einer Datei liegen – lokal gespeichert werden. Eine Idee wäre, diese Tabelle verschlüsselt bei einigen Anbietern zu hinterlegen, so dass man dann auch auf jedem beliebigen PC seine Dateien wiederherstellen kann.