

## **Diplomarbeit**

# **Media-Server auf Basis von OpenSource Software**

Ausgeführt zum Zweck der Erlangung des akademischen Grades  
**Dipl.-Ing. (FH) für Telekommunikation und Medien**  
an der Fachhochschule für Telekommunikation und Medien, St. Pölten  
Vertiefungsrichtung Medientechnik

**/fh/// Fachhochschule St. Pölten**

Telekommunikation und Medien  
Matthias Corvinus-Straße 15  
3100 St.Pölten

unter der Erstbetreuung von  
**Dipl. Ing. (FH) Fritz GRABO**

Zweitbegutachtung von  
**Markus SEIDL Bakk.**

ausgeführt von

**Alexandra LODER-TAUCHER**  
**Matrikelnummer: 0310038037**

Graz, den 11. September 2007

---

## **Danksagung**

---

Diese Diplomarbeit wurde im Studienjahr 2007 an der Fachhochschule für Telekommunikation und Medien, St. Pölten, in der Vertiefungsrichtung Medientechnik durchgeführt.

Danksagung an alle Personen der Fachhochschule St. Pölten, die an der Durchführung dieser Diplomarbeit beteiligt waren. Im Besonderen an Fritz Grabo für die fachkundige Betreuung während der letzten Monate.

Danksagung an meine Familie, Freundinnen und Freunde, die mich immer begleitet haben. Ganz besonderer Dank gilt meinen Eltern, die mir dieses Studium ermöglicht haben und meinem Freund Michi, der mir in den letzten vier Jahren immer zur Seite stand. Danke für eure Unterstützung zum erfolgreichen Gelingen dieser Arbeit!

---

## **Ehrenwörtliche Erklärung**

---

Ich versichere, dass

- ich diese Diplomarbeit selbständig verfasst, andere als die angegebenen Quellen und Hilfsmittel nicht benutzt und mich auch sonst keiner unerlaubten Hilfe bedient habe.
- ich dieses Diplomarbeitsthema bisher weder im Inland noch im Ausland einem Begutachter/einer Begutachterin zur Beurteilung oder in irgendeiner Form als Prüfungsarbeit vorgelegt habe.

Diese Arbeit stimmt mit der von den Begutachtern beurteilten Arbeit überein.

Graz, den 11. September 2007

Alexandra LODER-TAUCHER

---

## Zusammenfassung

---

Die vorliegende Arbeit setzt sich mit der Thematik der zentralen Datenverwaltung von Audiodaten mittels Medienserver auseinander.

Nach einer Internetrecherche wird ein Überblick über proprietäre Media-Server Systeme für das Heimnetzwerk gegeben. Aus einem Vergleich der Funktionen wird ein Anforderungsprofil für eine Lösung auf OpenSource Basis erstellt.

In Kapitel 3 werden bereits vorhandene OpenSource Komponenten für den Betrieb von Medienservern nach bestimmten Kriterien analysiert und gegenübergestellt. Durch diesen Vergleich sollen jene Tools ausgewählt werden, die für den Betrieb eines Medienservers im internen Netzwerk eingesetzt werden können.

Das zusammengestellte System soll es der Benutzerin/dem Benutzer ermöglichen, große Musiksammlungen (MP3, Audio CDs und ähnliches) zentral auf einem Server abzuspeichern. Innerhalb des Heimnetzwerks soll von allen Computern über verschiedene Anwendungen (Webinterface, netzwerkfähiger Musikplayer, ...) auf die Daten zugegriffen werden können.

---

## **Abstract**

---

This diploma thesis studies data management of digital audio using a mediaserver. Based on internet research a review of proprietary mediaserver systems for home networks is given in the first part of the work. Comparing the common features of proprietary software, a specification for an open-source solution has been defined.

Chapter 3 provides analysis and comparison of existing open-source components for realization of mediaservers. On this basis appropriate tools have been chosen for a concrete mediaserver implementation designed for home network use.

The preconfigured system offers the possibility to save capacious music collections (MP3, Audio CDs, and similiar media) centrally on a server. Within the home network, this media collection is uniformly accessible via different applications (Webinterface, networkenabled music player, ...).

---

## Inhaltsverzeichnis

---

<b>1</b>	<b>Einleitung</b>	<b>1</b>
1.1	Definition des Problems . . . . .	1
1.2	Zielgruppe . . . . .	1
1.3	Forschungsleitende Fragestellung . . . . .	2
1.4	Methodenbenennung . . . . .	2
<b>2</b>	<b>Analyse proprietärer Medienserver</b>	<b>3</b>
2.1	Recherche . . . . .	3
2.1.1	Welche kommerziellen Systeme werden angeboten? . . . . .	3
2.2	Vergleich der Systeme . . . . .	4
2.2.1	Grundsätzliche Funktionen . . . . .	4
2.2.2	Wodurch heben sich die einzelnen Produkte heraus? . . . . .	5
2.3	Zubehör für Medienserver . . . . .	6
2.3.1	YAMAHA - MusicCast . . . . .	6
2.3.2	Slim Devices . . . . .	8
2.4	Fazit . . . . .	9
<b>3</b>	<b>OpenSource Komponenten für Medienserver</b>	<b>10</b>
3.1	Anforderungsprofil für eine vergleichbare Lösung auf OpenSource Basis	10
3.2	Erweiterte Möglichkeiten durch Betrieb am PC . . . . .	11
3.3	Recherche: vorhandene OpenSource Tools . . . . .	11
3.4	Testprotokolle der ausgewählten Programme . . . . .	12
3.4.1	Ampache . . . . .	13
3.4.2	Jinzora . . . . .	16
3.4.3	Subsonic . . . . .	19
3.4.4	VLC . . . . .	21
3.5	Gegenüberstellung der Ergebnisse . . . . .	23
<b>4</b>	<b>Zusammensetzung des Medien-Server Systems</b>	<b>25</b>
4.1	Strukturdiagramm . . . . .	25
4.2	Das Server System . . . . .	27
4.2.1	Linksys NSLU2 . . . . .	27
4.2.2	Live-CD . . . . .	28
4.2.3	Virtuelle Systeme . . . . .	29
4.3	Die Software im Hintergrund . . . . .	30

4.3.1	Das Betriebssystem . . . . .	30
4.3.2	Der Webserver . . . . .	31
4.3.3	Samba Server . . . . .	31
4.3.4	Der Webbrowser . . . . .	32
4.3.5	Audio Player . . . . .	33
4.3.6	Brennen von CDs oder DVDs . . . . .	33
4.4	Netzwerkverbindung . . . . .	34
4.4.1	Verbindung zwischen Server und Client . . . . .	34
4.4.2	Datenraten . . . . .	34
4.4.3	Streaming versus Download . . . . .	35
4.5	Datenverwaltung . . . . .	36
4.5.1	Komprimierung von Audiodaten . . . . .	37
4.5.2	Ordnerstruktur . . . . .	41
4.5.3	Dateinamenvergabe . . . . .	41
4.5.4	Speichermedien . . . . .	43
<b>5</b>	<b>KuBox - ein virtueller Medienserver</b>	<b>47</b>
5.1	Getting Started - Anleitung zur Inbetriebnahme . . . . .	47
5.1.1	Voraussetzungen für den Betrieb eines Medienservers . . . . .	48
5.1.2	Liste der zu installierenden Programme . . . . .	48
5.1.3	Verfügbare Dienste des Servers . . . . .	51
5.2	Beschreibung eines praktischen Beispiels - KuBox . . . . .	52
5.2.1	VirtualBox . . . . .	52
5.2.2	Ubuntu auf der Virtuellen Box - Bereits Vorkonfiguriert... . .	53
5.2.3	Verwendung der KuBox . . . . .	58
5.3	Erweiterungsmöglichkeiten . . . . .	61
5.3.1	Datensicherung . . . . .	61
5.3.2	Anspruch auf hohe Qualität . . . . .	62
5.3.3	Tonstudio . . . . .	62
5.3.4	Ultrastar - Singstar mit OpenSource . . . . .	62
<b>6</b>	<b>Ergebnisse</b>	<b>63</b>
6.1	Diskussion der Ergebnisse . . . . .	63
6.2	Ausblick . . . . .	65
	<b>Literaturverzeichnis</b>	<b>66</b>
	<b>A Passwörterliste</b>	<b>74</b>
	<b>B Schluck Installation</b>	<b>75</b>

## Einleitung

---

### 1.1 Definition des Problems

Die zentrale Datenspeicherung und Verfügbarkeit von Daten für mehrere BenutzerInnen innerhalb eines internen Netzwerks ist ein aktuelles Thema - nicht zuletzt im Audiobereich, wo es zum Beispiel im Tonstudio wichtig ist, Geräusche-Samples, Projektdateien oder Musiktitel auf mehreren Rechnern zur Verfügung zu haben. Aber auch im privaten Bereich steigt die Nachfrage für zentrale Verwaltungssysteme von Daten und Musiksammlungen. Es werden zwar proprietäre<sup>1</sup> Systeme angeboten, dafür entstehen aber zusätzliche Anschaffungskosten für die benötigte Hardware.

Daher liegt die Überlegung nahe, bereits bestehende interne Netzwerke um die entsprechenden Komponenten zu erweitern, um so ein vergleichbares System zu erhalten. Für den Nachbau dieser ist es notwendig, über die Funktionalität bestehender Systeme, sowie die Anforderungen der BenutzerInnen bescheid zu wissen. Daraus ergibt sich ein Testprofil, an Hand dessen angebotene OpenSource Programme analysiert werden können. Im Medienserver sollen unterschiedliche Anwendungen kombiniert werden, um verschiedene Zugriffsvarianten auf die Daten zu ermöglichen.

### 1.2 Zielgruppe

Als Zielgruppe für das Projekt werden semiprofessionelle Computer UserInnen gesehen, die grundlegende Kenntnisse über die Verwaltung von größeren Datenmengen und die gängigen Audioformate haben.

---

<sup>1</sup>„Von proprietär spricht man [...] im IT-Bereich, wenn ein Programm oder System nicht einem allgemeinen Standard entspricht, sondern herstellerspezifische Vorgaben umsetzt. Daten aus einem proprietären System lassen sich nicht oder nur mit sehr hohem Aufwand auf anderen System benutzen.“  
[zit. LEXEXACT, 07.08.2007 (Webseite) [LEX07]]

### **1.3 Forschungsleitende Fragestellung**

Kann durch das Zusammenstellen eines Medienserver Systems aus OpenSource Komponenten die Funktionalität eines vergleichbaren proprietären Systems erreicht beziehungsweise sogar übertroffen werden?

### **1.4 Methodenbenennung**

Zu Beginn wird eine ausführliche Internet-Recherche über bereits existierende Systeme und deren Funktionalität durchgeführt. Nach einer Gegenüberstellung der relevanten Funktionen von kommerziellen Produkten wird ein Anforderungsprofil für eine vergleichbare Lösung auf OpenSource Basis erstellt. Anschließend werden angebotene OpenSource Tools an Hand einer Kriterienliste beurteilt, um geeignete Komponenten für ein System auf OpenSource Basis zu finden.

Im praktischen Teil wird ein System erstellt und getestet, das den Anforderungen aus den vorhergehenden Analysen entspricht. Abschließend werden durch eine ausführliche Beschreibung des Systems dessen Zusammensetzung und Funktionalität sowie die Schritte der Installation und die getätigten Konfigurationen dargestellt.

### Analyse proprietärer Medienserver

---

#### 2.1 Recherche

##### 2.1.1 Welche kommerziellen Systeme werden angeboten?

Bei einer ausführlichen Recherche im Internet zum Thema Medienserver<sup>1</sup> werden einige Produkte gefunden. Viele der angezeigten Lösungen bieten nur Streaming der Audiodateien an oder sind nicht netzwerkfähig, daher nicht als Datenverwaltungssystem einsetzbar. Es werden jedoch durchaus auch professionelle Systeme für den Home Entertainment Bereich angeboten. Folgende zwei Produkte werden für den Vergleich der Funktionen herangezogen:

- \* **Slim Devices (a Logitech Company)** Ein auf die Software *SlimServer* (zur Installation auf Computern) aufbauendes Mediaserver System, für welches zwei verschiedene Empfangstationen angeboten werden. *Squeezebox*: ein kleiner Client für jeden Raum (Abbildung 2.1(a)) und *Transporter*: ein audiophiler Netzwerk-Musikplayer zum Anschluss an die Stereoanlage (Abbildung 2.1(b)), [vgl. SLIM DEVICES, 12.12.2006 (Webseite) [SLI06a]]
- \* **MusicCast (Yamaha)** Ein von YAMAHA entwickeltes System zum Management und zur Verteilung von Musik. Es werden unter anderem folgende Komponenten angeboten: *MCX-2000*: Ein Server mit eingebauter Festplatte und Netzwerkschnittstelle und *MCX-A10*: einer von drei angebotenen Clients (Abbildung 2.2). [vgl. YAMAHA, 12.12.2006 (Webseite) [YAM06a] und [YAM06b]]

---

<sup>1</sup>“Media-Server sind Netzwerkkomponenten für die kommende Netzwerkgeneration (NGN). Sie werden von einem Softswitch oder Application Server gesteuert und liefern Kapazitäten für die Medienübertragung in zeitgetakteten und paketvermittelten Netzen.“ [zit. IT Wissen, 14.12.2006 (Webseite) [ITW06]]



Abbildung 2.1: Slim Devices



Abbildung 2.2: Yamaha MusicCast

## 2.2 Vergleich der Systeme

In diesem Abschnitt werden zunächst die grundlegenden Funktionen der oben angeführten Produkte gegenübergestellt. In weiterer Folge werden die jeweiligen Vorzüge der einzelnen Produkte aufgelistet - auch in Hinblick auf die Anforderungen an den OpenSource Medienserver.

### 2.2.1 Grundsätzliche Funktionen

Als Basiskonfiguration bei *MusicCast* ist zum einen der digitale Audio Server *MCX-2000* und zumindest ein Client (siehe Abschnitt 2.3.1) notwendig. Ein Server kann bis zu 15 Stationen bedienen. Die Musiktitel können vom eingebauten CD-Laufwerk (auch als CD-Brenner einsetzbar), von einem angeschlossenen Computer, über eine der Audioschnittstellen oder bei aufrechter Verbindung aus dem Internet geladen werden.

Bei *Slim Devices* gilt die Computersoftware *SlimServer* als Grundlage, von der aus beliebig viele Stationen über LAN oder WLAN versorgt werden können. Die Software läuft auf allen gängigen Betriebssystemen und kann auch ohne Clients direkt

am Computer verwendet werden. *SlimServer* kann um rund €188 erworben werden, beim Kauf von *Transporter* oder *Squeezebox* wird die Software mitgeliefert. Der *Transporter* ist mit einer Netzwerkschnittstelle und zahlreichen Audioschnittstellen ausgestattet, Musiktitel können über den Computer oder über Internetradio bezogen werden.

Aus einer Analyse der Datenblätter der einzelnen Geräte ergibt sich folgende Übersicht der grundsätzlichen Funktionen [vgl. YAMAHA, SLIM DEVICE, 12.12.2006 (Webseite) [SLI06b, SLI06c, YAM06d, YAM06c]]. In Tabelle 2.1 sind Informationen über Audioformate, wichtige Features, Zubehör und Preise zu finden.

	Yamaha MusicCast	Slim Devices
Basic Setup	1 MCX-2000 + 1 MCX-A10	1 Transporter / 1 Squeezebox + Slimserver (Software)
maximale Anzahl Clients	5 WLAN / 10 LAN	unbegrenzt
Verbindung zum Computer	möglich	notwendig
unterstützte Audioformate	PCM, MP3	alle üblichen Formate
Audio Input analog	Stereo (RCA)	-
Audio Input digital	optisch	optisch, RCA, BNC, XLR
Audio Output analog	Stereo (RCA)	Stereo (XLR, RCA)
Audio Output digital	optisch	optisch, RCA, BNC, XLR
Netzwerkfähig	LAN / WLAN	LAN / WLAN
Streaming	individuell / broadcast	individuell / broadcast
Internet Radio	Ja	Ja
Fernbedienung	Ja	Ja
Preis (Basic Setup; in €)	1799 + 546,50	1510 / 188 (Software incl.)
Preis (weitere Komponenten; in €- Stand 12 2006)	MCX-C15: 300 MCX-CA15: 160	Squeezebox wireless: 225 Infrant ReadyNAS NV+ / Squeezebox Bundle: 1145

**Tabelle 2.1:** allgemeine Gegenüberstellung der Systeme

## 2.2.2 Wodurch heben sich die einzelnen Produkte heraus?

Nach einem detaillierten Vergleich der Systeme zeigen sich einige Unterschiede in deren Funktionsumfang. Diese liegen einerseits in einem unterschiedlichen Serverkonzept und andererseits in der Unterstützung verschiedener Audioformate und Audioschnittstellen.

### **Yamaha MusicCast**

- \* eingebaute Festplatte (160 GB - erweiterbar IDE/ATA)
- \* interner WLAN Router im Hauptgerät
- \* eingebautes CD-Laufwerk inklusive Brenner (allerdings nur für WAV Dateien)
- \* eigene Song Statistik für jedes Gerät
- \* Videoausgang für On Screen Menue
- \* drei verschiedene Clients (zum Teil auch zur Wandmontage)

### **Slim Devices**

- \* *SlimServer* Software auch ohne Endgeräte verwendbar
- \* unterstützt alle gängigen Audioformate
- \* Vielzahl an digitalen und analogen Audio Ein- und Ausgängen, zusätzlich ein Anschluss für die *world clock*
- \* Titel aus dem Verzeichnis können auf Computer heruntergeladen werden (nicht nur Streaming)
- \* unterstützte Serversysteme: Macintosh: Mac OS X 10.3 oder höher, Windows: 733MHz Pentium mit Windows NT/2000/XP, Linux/BSD/Solaris/Other: Perl 5.8.3 oder höher
- \* unbegrenzte Anzahl an Clients möglich (*Transporter, Squeezebox wired, Squeezebox wireless*)

## **2.3 Zubehör für Medienserver**

### **2.3.1 YAMAHA - MusicCast**

Für den *YAMAHA MusicCast* Medienserver werden unterschiedliche Clients angeboten. Zum einen ein Aufstellgerät, das optional auch mit dazupassenden Lautsprechern ausgestattet ist. Zum anderen zwei verschiedene Einbauelemente für die Wand. [vgl. YAMAHA, 12.12.2006 (Webseite) [YAM06a]]

#### **MusicCast MCX-A10 (Abbildung 2.3)**

- \* Wireless LAN Kommunikation mit dem Musik Server
- \* Eingebauter HiFi digital Verstärker für verschiedene Lautsprecherkonfigurationen
- \* On-Screen Video Out für Monitor Display
- \* Line-Out Audio Ausgang zur Anbindung an bestehende Audio Komponenten
- \* Subwoofer Ausgang für aktiven Subwoofer
- \* Audio Eingang zum Anschluss weiterer Eingangsgeräte
- \* LAN (Ethernet) Netzwerk Verbindung

- \* Großes LCD Display



**Abbildung 2.3:** Yamaha MusicCast MCX-A10

### **MusicCast MCX-C15 (Abbildung 2.4)**

- \* LAN Kommunikation mit dem Musik Server
- \* 4 Favoriten-Knöpfe zum direkten Aufruf von Lieblings CD, Playliste oder Radiosender
- \* Erweiterte Infrarot-Funktion
- \* Multi I/O-Box zum Anschluss an andere Komponenten
- \* LCD Display



**Abbildung 2.4:** Yamaha MusicCast MCX-C15

### **MusicCast MCX-CA15 (Abbildung 2.5)**

- \* Einbauverstärker passend zum MCX-C15
- \* 2 x 17W
- \* Anschlussmöglichkeit für weitere Komponenten
- \* Video Ausgang zur Darstellung des On Screen Menüs
- \* Zusätzlicher Line-Eingang (3,5 mm Stereo Klinke)



Abbildung 2.5: Yamaha MusicCast MCX-CA15

### 2.3.2 Slim Devices

Von *Slim Devices* gibt es zusätzlich zum Transporter (Abbildung 2.1(b)) weitere LAN gebundene und kabellose Clients in verschiedenen Farbkombinationen. Des Weiteren wird ein Paket mit einem *Storage Server* und zwei *Squeezeboxen wireless* angeboten. [vgl. SLIM DEVICES, 12.12.2006 (Webseite) [SLI06a]]

#### Squeezebox wireless / Squeezebox wired (Abbildung 2.6)

- \* Digitale (optisch, RCA) und analoge Audioschnittstellen
- \* WLAN und/oder Ethernet Ports (je nach Model)
- \* Spielen von MP3, WAV, FLAC, WMA, AAC, und Ogg Files vom Computer
- \* schnelles Durchsuchen von großen Songlibraries
- \* Zugriff auf Internetradio Stationen, auch ohne laufenden Computer
- \* Streamen von Musik in mehreren Räumen (einzelne Clients) [vgl. SLIM DEVICES, 12.12.2006 (Webseite) [SLI06d]]



Abbildung 2.6: Squeezebox vorne / hinten

### Infrant ReadyNAS NV+ / Squeezebox Bundle (Abbildung 2.7)

Infrant Technologies' ReadyNAS NV+ ist eines der neuesten Produkte der *ReadyNAS* Produkt Gruppe. Die „Network attached storage“ (NAS) Einheit bietet PrivatbenutzerInnen oder Firmen eine einfache Möglichkeit, große Datenmengen verfügbar zu machen.

Im Paket sind enthalten:

- \* 1 x Infrant ReadyNAS NV+, 1.0TB (4x250GB Seagate NL35 Disks), X-RAID, 256MB RAM, Gigabit Ethernet
- \* 2 x Squeezebox Wireless



Abbildung 2.7: 2 Wireless Squeezeboxen und ein terabyte ReadyNAS NV+

## 2.4 Fazit

Zusammenfassend kann über die Ergebnisse in Kapitel 2 gesagt werden, dass sich die angebotenen proprietären Systeme für den Einsatz im Home Entertainment Bereich als sehr praktikabel erweisen. Der Betrieb ist in Zusammensetzung mit den ebenfalls angebotenen Clients vorgesehen. Innerhalb des jeweiligen Systems sind die Komponenten gut aufeinander abgestimmt und die Features BenutzerInnenfreundlich umgesetzt. Als Zielgruppe dieser Produkte gelten PrivatkundInnen im Home Entertainment Bereich, die sich nicht mit technischen Details auseinander setzen wollen, sondern „a system out of the box“ bevorzugen.

Aus den Produktanalysen können wichtige Erkenntnisse und Anregungen für die Spezifikation des Medienservers auf Basis von OpenSource Software gewonnen werden. Diese Ergebnisse fließen in den weiteren Teil der Arbeit ein.

### OpenSource Komponenten für Medienserver

---

Als Bezeichnung des Medienservers auf OpenSource Basis wird in weiterer Folge der Name **Audio Medien Server** auf **Linux-Basis (AMSL)** verwendet.

#### 3.1 Anforderungsprofil für eine vergleichbare Lösung auf OpenSource Basis

Die Kapitel 2 angeführten Featurelisten und Produktvergleiche werden als Basis für die Spezifikation eines OpenSource Medienservers verwendet. Ein derartiges System kann im privaten Home-Entertainment Bereich eingesetzt werden. Es ist jedoch nicht für den Betrieb eines öffentlichen Servers gedacht. Dabei würden zusätzliche sicherheitstechnische und urheberrechtliche Anforderungen an das System gestellt, auf die hier nicht näher eingegangen wird. Zudem haben die meisten Breitband Internetanschlüsse eine zu geringe Uploadgeschwindigkeit, um einen solchen Server zu betreiben. Auch für einen Betrieb in einem Tonstudio ist das System in dieser Form nicht geeignet - Erweiterungsmöglichkeiten dafür werden in Kapitel 5.3.3 diskutiert.

Der Vorteil eines OpenSource Systems im privaten Bereich ist, dass keine zusätzlichen Softwarekosten anfallen. Außerdem kann der Medienserver auf einem zumeist ohnehin vorhandenen Computer installiert werden. In der folgenden Auflistung werden die Anforderungen an ein Medienserver System für den Home-Entertainment Bereich angeführt.

- \* Archivierung von Audio Dateien (Geräusche, Lieder, Hörbücher, ...) auf einer Computerfestplatte oder einem externem Storage Server (in allen gängigen Audioformaten, unkomprimiert oder komprimiert, wie zum Beispiel AIFF, WAV, PCM, MP3, Ogg Vorbis, WMA, ...)
- \* Plattformunabhängigkeit (gilt hier als Auswahlkriterium für Programme die getestet werden)
- \* Effizientes Importieren von Audio Dateien (z.B. MP3 CD) (Speichern am Server und Eintrag in DB)
- \* Einlesen von Audio-CDs über das CD-Laufwerk (Speichern am Server)

- \* Automatisches Eintragen in eine Datenbank
- \* Automatischer CDDDB-Lookup für die importierten CDs<sup>1</sup>
- \* Erstellen und Anzeigen von ID3-Tags bei MP3 Daten
- \* Abspielbar über das Netzwerk in gewöhnlichen Softwareplayern (Amarok, Songbird, Winamp, Windows Media Player, I-Tunes, ...)
- \* Erstellen und Speichern von Playlists, individuell für jede/n BenutzerIn
- \* Webinterface für Suche und Bearbeitung von Einträgen, Upload und Hören von Dateien
- \* Verwaltung der gespeicherten Dateien

Für den Einsatz eines Medienservers im professionellen Studiobetrieb ergeben sich einige zusätzliche Anforderungen an Hard- und Software:

- \* Professionelle Soundkarte im Server Computer mit analogen und digitalen Ein- und Ausgängen, um eine qualitativ hochwertige Digitalisierung und Wiedergabe zu ermöglichen
- \* CD-Brenner im Server Computer zum Brennen von Musiksammlungen oder Archivieren von Projekten
- \* Zur Verfügung stellen der Daten per Samba, FTP, Stream, ... (Download der Files oder lediglich Hören per Stream)
- \* Netzwerkzugriff von mehreren Clients im internen Netzwerk (eventuell NAS) (über LAN oder WLAN)
- \* Backupmöglichkeit und Ausfallsicherheit des Servers

Diese Erweiterungen werden zwar am Ende der Diplomarbeit kurz diskutiert, werden aber im praktischen Teil nicht realisiert.

## 3.2 Erweiterte Möglichkeiten durch Betrieb am PC

Im Gegensatz zu den proprietären Produkten, die an die Sendestation und die Empfangsclients gebunden sind, kann hier das System auf einem herkömmlichen Desktop PC oder Notebook betrieben werden. Im vorhandenen internen Netzwerk kann von jedem Client-Computer aus Musik gestreamt oder auf die eigene Festplatte heruntergeladen werden. Durch den Betrieb auf einem Notebook ist man mit dem System nicht an einen fixen Standort gebunden, sondern kann es auch an anderen Orten betreiben.

## 3.3 Recherche: vorhandene OpenSource Tools

Im Rahmen einer ausführlichen Recherche können zahlreiche Programme gefunden werden, die für die Verwaltung von Musiksammlungen entwickelt wurden. Durch das

---

<sup>1</sup>Compact Disk Database (CDDDB) „... ist eine Online-Datenbank für Audio-CDs in der alle Titel, deren Interpreten, die Namen der Tracks und einige weitere Informationen [...] gespeichert sind. In dieser Datenbank kann ebenso nach Titeln oder Interpreten recherchiert werden.“ [zit. ITWISSEN, 14.08.2007 (Webseite) [ITW07a]]

im Anforderungsprofil (siehe Abschnitt 3.1) enthaltene Entscheidungskriterium *Plattformunabhängigkeit* kann die Auswahl der Programme für den Test auf vier eingeschränkt werden.

- \* AMPACHE „... is a Web-based Audio file manager. [...] It allows you to view, edit, and play your audio files via the web.“ [zit. AMPACHE, 23.04.2007 (Webseite) [AMP07]] - aktuelle Version: 3.3.3.4 (last stable Release: 09.07.2007)
- \* JINZORA „... enables you to stream your digital music and videos to any internet connected computer using a web browser.“ [zit. JINZORA, 23.04.2007 (Webseite) [JIN07]] - aktuelle Version: 2.7.5 (last stable Release: 06.03.2007)
- \* SUBSONIC „... is a free, web-based media streamer, providing access to your entire music collection wherever you are.“ [zit. SUBSONIC, 23.04.2007 (Webseite) [SUB07b]] - aktuelle Version: 3.1 (last stable Release: 30.07.2007)
- \* VLC „... is a free cross-platform media player [...] [that] supports a large number of multimedia formats, without the need for additional codecs“ [zit. VLC, 23.04.2007 (Webseite) [VLC07]] - aktuelle Version: 0.8.6b (last stable Release: 19.04.2007)

## 3.4 Testprotokolle der ausgewählten Programme

Um herauszufinden, welche der gefundenen Programme für den Einsatz im AMSL geeignet sind, werden diese nach folgenden Kriterien getestet und einander gegenübergestellt:

- \* Software Voraussetzungen
- \* Programmiersprache -> eventuelle Erweiterungen
- \* Version (last stable release)
- \* Dokumentation, aktives Forum, Hilfestellung bei Problemen
- \* Installation
- \* GUI (grafische Benutzeroberfläche)
- \* Betrieb
- \* Datenbank
- \* Import von Musikverzeichnissen
- \* unterstützte Audioformate
- \* Netzwerkfähigkeit, Streaming
- \* Playlist Verwaltung

Jedes der ausgewählten Programme wird auf dem selben Betriebssystem (SuSe Linux 10.2) installiert. Die einzelnen Schritte von der Installation über die Konfiguration und den Betrieb werden getestet und ausführlich dokumentiert. In den anschließenden Testprotokollen werden die Punkte der Kriterienliste durch **fette Schreibweise** hervorgehoben.

### 3.4.1 Ampache

#### getestete Version: 3.3.3

*Ampache* ist in PHP umgesetzt und verwendet eine MySQL-Datenbank, daher gelten als **Software Voraussetzungen** PHP (4.3.x oder neuer) und MySQL (4.x oder neuer). Diese müssen auf einem verfügbaren Webserver (siehe Kapitel 4.3.2), wie zum Beispiel *Apache httpd* laufen. Zusätzlich sind für den Betrieb noch ein Webbrowser (siehe Kapitel 4.3.4) und ein Musikplayer (siehe Kapitel 4.3.5) erforderlich.

**Dokumentation** Für das Projekt ist eine eigene Wiki angelegt, auf der Themen zur Installation und Benützung, sowie Problembehandlung von *Ampache* diskutiert werden.<sup>2</sup> Allerdings gibt es für dieses Wiki keine Suchfunktion, was für das Finden von relevanten Informationen sehr hilfreich wäre. Zusätzlich gibt es ein Onlineforum<sup>3</sup>, wo Fragen zum Projekt gepostet werden können, die von den EntwicklerInnen oder der Community beantwortet werden.

Die **Installation** erfolgt direkt über den Webbrowser - nachdem das Paket `ampache-3.3.3.tar.gz` am Webserver entpackt wurde, kann diese durch das Wechseln in das entsprechende Verzeichnis gestartet werden. Über die Eingabe in das vorgegebene Formular werden die benötigten Einstellungen für die **Datenbank** beziehungsweise in der Datei `ampache.cfg.php`, in welcher alle relevanten Daten für die Konfiguration gespeichert werden, vorgenommen. Die MySQL Datenbank wird während der Installation angelegt und kann danach über einen MySQL-Client bearbeitet werden.



Abbildung 3.1: Ampache Login-Fenster

Mit dem bei der Installation angelegten Benutzernamen und dem entsprechenden Passwort kann man sich anschließend auf der Login-Seite anmelden (Abbildung 3.1). Zu erreichen ist diese Seite über die folgende Adresse:

<sup>2</sup>Zu finden unter der Adresse <https://ampache.bountysource.com/wiki>

<sup>3</sup><http://ampache.org/forums/>

localhost/.../ampache-3.3.3/login.php

Bei einem *Admin* Zugang können weitere BenutzerInnen mit unterschiedlichen Berechtigungen angelegt werden. Im Interface für normale *User* sind die Administrator Einstellungen im Menü nicht enthalten.

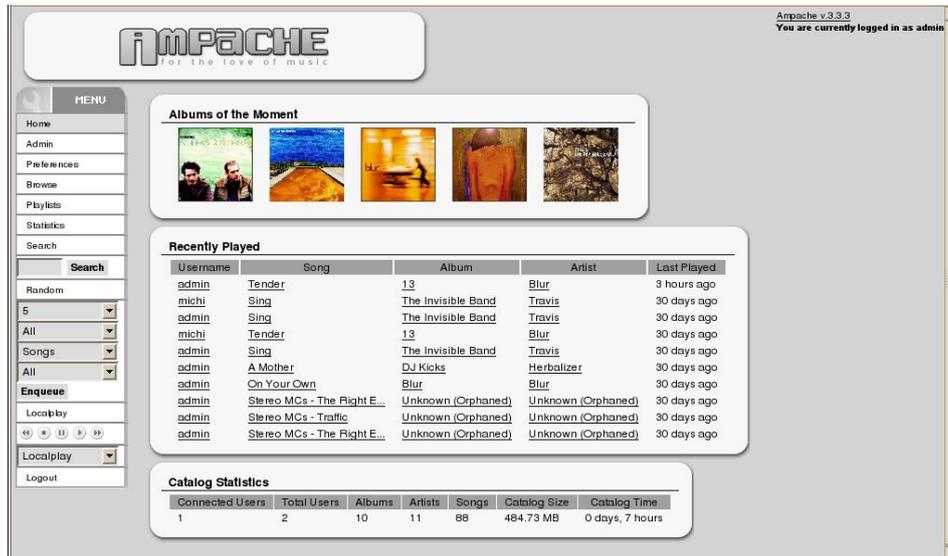


Abbildung 3.2: Ampache Interface

Auf der **grafischen Benutzeroberfläche (GUI)** (siehe Abbildung 3.2) sind die einzelnen Funktionen jeweils in eigenen Fenstern dargestellt, wodurch der verfügbare Platz am Bildschirm nicht optimal genutzt werden kann. Die Darstellung ist für eine größere Bildschirmauflösung, also mindestens 1280x1024 Pixel optimiert. Bei kleineren Auflösungen wirkt die GUI vor allem mit größeren Datenmengen (lange Verzeichnislisten) sehr unübersichtlich und überladen.

Die Menüführung ist nicht einheitlich, einerseits ist das Hauptmenü mit Unterpunkten immer auf der linken Seite des Bildschirms sichtbar. Das Untermenü ist mit Reitern gelöst und teilweise in der Navigation sehr umständlich, da die einzelnen Punkte nicht miteinander verlinkt sind. Daher sind oft sehr viele Klicks nötig, um die gewünschten Einstellungen zu treffen.

**Betrieb** Das Laden von Katalogen beziehungsweise Musikverzeichnissen funktioniert problemlos. Es können auch nach der Installation noch weitere Kataloge hinzugefügt werden, die sich auch auf unterschiedlichen Festplatten befinden können, auch eine URL als Pfad ist möglich. Allerdings können nur Verzeichnisse importiert werden, deren Tracks die entsprechenden Leseberechtigungen aufweisen (-r--r--r--)<sup>4</sup>.

Von *Ampache* werden laut Projektwebseite folgende **Audioformate** unterstützt: *MP3 (Id3v1 & Id3v2), OGG, WMA, RM, M4A/AAC/MP4 (iTunes files), FLAC, MPC,*

<sup>4</sup>Zugriffsrechte, Benutzer und Gruppenzugehörigkeit unter Linux: „Mit jeder Datei bzw. mit jedem Verzeichnis werden folgende Informationen gespeichert: \* der Besitzer (owner) der Datei, \* die Gruppe, der die Datei zuzuordnen ist, \* neun Zugriffsbits (rwxrwxrwx für read / write / execute für den Besitzer, für alle Gruppenmitglieder und für den Rest der Welt) ... .“ [zit. KOFLER 2006, S 625 [KOF06]]

*SPX, RA, APE, SHN, Wavepack*. Leider können Titel, die im Format *WAV* gespeichert sind, nicht abgespielt werden.

**Import von Musikverzeichnissen** Sämtliche Titel aller geladenen Kataloge werden automatisch in die Liste für die Suche aufgenommen, in welcher nach verschiedenen Kriterien gesucht werden kann, unter anderem ist auch eine indizierte Suche möglich. Ausgewählte Titel können verschiedenen bereits bestehenden oder neu angelegten **Playlists** hinzugefügt oder direkt abgespielt werden. Diese können auch abgespeichert und später wieder editiert werden. Allerdings ist die Darstellung der Liste unübersichtlich, weil zu viele, teilweise auch irrelevante, Informationen angezeigt werden.

Zur Wiedergabe der Tracks können grundsätzlich alle Musikplayer verwendet werden, die das Streamen (siehe dazu auch Kapitel 4.4.3) über HTTP unterstützen. Auf der Projektwebseite sind folgende Player für die Verwendung mit *Ampache* aufgelistet: *Winamp, Windows Media Player, XMMS, iTunes, Real Player, XSPF-Player*<sup>5</sup>, *MPD*<sup>6</sup>. Die Auswahl des Players läuft über die Einstellungen im Menü.

Mit der Funktion *localplay* können Tracks direkt im Browser abgespielt werden, zum Beispiel über *MPD*. Bei der Funktion *stream* wird die Playlist an den Standardplayer, wie beispielsweise *Amarok* oder *realplayer* gesendet und dort abgespielt. Einzelne Tracks können direkt aus der Liste, durch das Klicken auf das entsprechende Symbol „*link*“, auf den eigenen Computer heruntergeladen werden.

Für Alben aus der Sammlung werden, soweit in der Bezugsquelle vorhanden, die Bilder der CD-Cover automatisch angezeigt. In den Einstellungen können auch nachträglich Bilder vom eigenen Computer oder anderen Pfaden geladen werden. ID3-Tags<sup>7</sup> von den Titeln im *.mp3* Format können in einem eigenen Formular angezeigt und editiert werden.

Für den **Netzwerkbetrieb** können in einer *Acces-List* IP-Adressen, beziehungsweise Subnetze eingegeben werden, denen der Zugriff erlaubt wird. Mittels Webbrowser kann mit der entsprechenden URL auf den *Amarok* Player zugegriffen werden. Über das Netzwerk können auch mehrere Player gleichzeitig betrieben werden.

---

<sup>5</sup>„XSPF Web Music Player is a flash-based web application that uses xspf playlist format to play mp3 songs. XSPF is the XML Shareable Playlist Format. The software is written in Actionscript 2.“ [zit. XSPF, 07.05.2007 (Webseite) [XSP07]] Um den XSPF-Player verwenden zu können, muss die Datei *xspf\_player.swf* in das *Ampache* Verzeichnis am Webserver kopiert werden. Eventuelle Pop-UP Blocker müssen im Browser deaktiviert sein.

<sup>6</sup>„Music Player Daemon (MPD) allows remote access for playing music (MP3, Ogg Vorbis, FLAC, AAC, Mod, and wave files) and managing playlists. MPD is designed for integrating a computer into a stereo system that provides control for music playback over a local network.“ [zit. MPD, 08.05.2007 (Webseite) [MPD07]] Vor der Verwendung des MPD Players muss der Dienst über die Kommandozeile gestartet werden.

<sup>7</sup>„An ID3 tag is a data container within an MP3 audio file stored in a prescribed format. This data commonly contains the Artist name, Song title, Year and Genre of the current audio file.“ [zit. ID3, 01.05.2007 (Webseite) [ID307a]]

### 3.4.2 Jinzora

#### getestete Version: 2.7.5

Die **Software Voraussetzungen** für den Betrieb von *Jinzora* sind, wie auch bei *Ampache* ein laufender Webserver, PHP (mindestens in der Version 4.2) und MySQL (ab Version 3.23.32 - nicht zwingend erforderlich aber empfohlen), da *Jinzora* auch in PHP umgesetzt ist.

Zur **Installation** des Programmes werden auf der Projektwebseite Verzeichnisse im `.tar.gz`<sup>8</sup> und im `.zip`<sup>9</sup> Format angeboten. Nach dem Herunterladen muss das jeweilige Verzeichnis auf dem Webserver extrahiert werden, von wo aus dann über die Kommandozeile der nachfolgende Befehl ausgeführt werden muss, um die Installation zu starten.

```
# sh configure.sh
```

Der Installationsvorgang ist in der Datei `inst.txt`, die dem Verzeichnis beiliegt, sehr gut dokumentiert. Diese Installationsanleitung ist ebenfalls auf der Projektwebseite zu finden.<sup>10</sup>

Innerhalb der Installationsroutine, die sehr ausführlich und professionell erscheint, werden die Anforderungen des Programmes an das System überprüft und die Ergebnisse aufgelistet. Fehlende Einstellungen oder Voraussetzungen werden angezeigt und können geändert werden. Weitere Konfigurationen wie *Installation-Type* („standalone“ oder „in CMS eingebunden“) und der *Jukebox-Mode* („Streaming only“ oder „Streaming & Jukebox“ = für Server seitige Jukebox) können getätigt werden.

Unter *main settings* wird das Passwort für den Benutzer `admin` angelegt und weitere Einstellungen für die *Login-Modalitäten* und den *Daten Import* getroffen. Durch das *Backend Setup* wird das Passwort für den Datenbank Benutzer angelegt, optional kann die **Datenbank** und alle benötigten Tabellen neu erstellt, oder bereits bestehende übernommen werden. Der **Import vom Medienverzeichnis** läuft problemlos, der Pfad dafür kann aus der Verzeichnisstruktur ausgewählt werden.

Am Ende der Installationsroutine, werden alle Konfigurationen automatisch in die Datei `settings.php` gespeichert. Das Verzeichnis *install* muss danach umbenannt oder gelöscht werden, um unbeabsichtigte Änderungen zu verhindern.

Obwohl auf der Projektwebseite von *Jinzora* Verlinkungen zu einer online **Dokumentation** angeboten werden, ist diese leider nicht verfügbar, da der entsprechende Server<sup>11</sup> nicht antwortet. Informationen rund um das Programm können im Forum gefunden werden<sup>12</sup>, Projektwiki ist keine eigene vorhanden. Allerdings können weitere Informationen nach dem Login im *Jinzora Support Center* gefunden werden, das über den Button „more“ im Programm erreicht werden kann.<sup>13</sup>

Das Login-Fenster (Abbildung 3.3) erreicht man durch das Wechseln in das entsprechende Verzeichnis am Webserver (siehe unten), wo man sich mit dem zuvor angelegten Benutzernamen und Passwort anmelden kann.

---

<sup>8</sup>Dateinamenserweiterung `.tar` = Tape Archive File, Dateinamenserweiterung `.gz` = Gzip Compressed Archive [vgl. FILEExt, 09.05.2007 (Webseite) [FIL07]]

<sup>9</sup>Dateinamenserweiterung `.zip` = Compressed Archive File [vgl. FILEExt, 09.05.2007 (Webseite)]

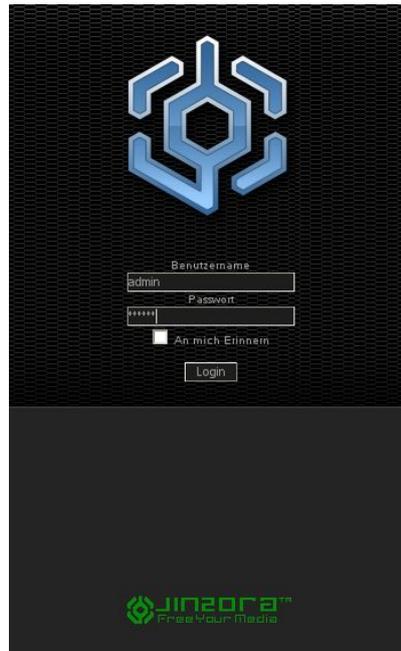


Abbildung 3.3: Jinzora Login-Fenster

localhost/.../jinzora/index.php

Die Gestaltung des **GUI** (Abbildung 3.4) ist sehr unstrukturiert und unübersichtlich. Die Navigation im Menü ist wenig zufriedenstellend, da keine Verlinkung der einzelnen Menüpunkte vorhanden ist. Daher muss nach jeder Einstellung das Menü geschlossen (wodurch die Veränderungen gespeichert werden) und für weitere Änderungen neu geöffnet werden. Benutzereinstellungen werden in der Datei `user_settings`, die sich im Verzeichnis `./jinzora/data/database/` befindet, gespeichert und können von dort aus im Texteditor geändert werden.

Im **Betrieb** läuft *Jinzora* instabil, durch das Ausführen von einigen Befehlen aus dem Menü kann das Programm nicht wieder herstellbar abstürzen. Von den angebotenen Styles sind einige fehlerhaft und unvollständig. Sie enthalten teilweise nicht alle Menüauswahlpunkte, sodass zum Beispiel die Auswahl vom Theme nicht rückgängig gemacht werden kann. Nur durch das Ändern des Themes in den `user_settings` kann das Programm wieder hergestellt werden, allerdings treten danach vermehrt Fehler in der Anwendung auf.

Eine Funktion für das **Hinzufügen** von neuen **Musikverzeichnissen** ist zwar implementiert, funktioniert jedoch nicht entsprechend. Auch die angebotene Funktion

---

[FIL07]]

<sup>10</sup>unter der Adresse <http://www.jinzora.com/forums/viewtopic.php?t=4631>

<sup>11</sup>[www.jinzorahelp.com](http://www.jinzorahelp.com)

<sup>12</sup><http://www.jinzora.com/forums/>

<sup>13</sup>Diese Angaben beziehen sich auf den Zustand zum Zeitpunkt des Tests - mittlerweile wurde die Projektwebseite überarbeitet und stellt die entsprechenden Dienste zur Verfügung. Die getestete Softwareversion 2.7.5 ist immer noch der aktuellste Release.

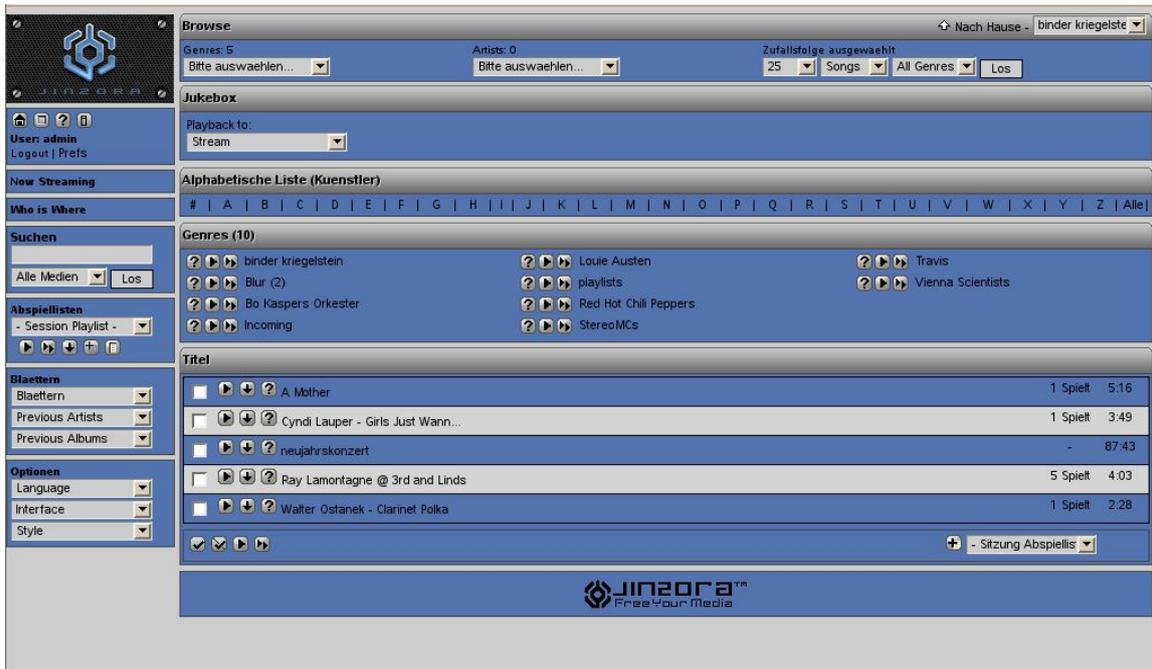


Abbildung 3.4: Jinzora Interface

zum Editieren von ID3-Tags entspricht nicht den Anforderungen. Allerdings gibt es die Möglichkeit, zusätzliche Informationen zu Tracks zu speichern.

Zum Abspielen der Tracks aus unterschiedlichsten **Audioformaten**<sup>14</sup> kann der Standardplayer am PC oder *MPD* (wenn aktiviert) verwendet werden. Nach den im Verzeichnis enthaltenen Musiktiteln kann über einen Index gesucht werden, markierte Titel können neuen oder bereits bestehenden **Playlists** hinzugefügt werden. Diese können direkt abgespielt oder gespeichert werden. Auch einzelne Tracks können vom Musikverzeichnis auf den eigenen PC heruntergeladen werden. Eine Möglichkeit zum Anzeigen von CD-Cover der Alben im Verzeichnis wurde nicht gefunden. Beim Abspielen der Tracks im integrierten *MPD* Player werden für die Anzeige nur die ID3-Tags verwendet, Informationen aus dem Dateinamen des Tracks werden nicht ausgelesen.

Die **Netzwerkfähigkeit** und das **Streamen** konnte auf Grund des instabilen Betriebes nicht getestet werden.

<sup>14</sup> mp3, ogg, wma, wav, midi, mid, flac, aac, mp4, rm, mpc, m4a, wv, shn, ape, ofr

### 3.4.3 Subsonic

#### getestete Version: 3.0

**Software Voraussetzungen** Bevor *Subsonic* in Betrieb genommen werden kann, muss am Computer *Java* und ein JSP Container, wie zum Beispiel *Tomcat*<sup>15</sup> (in der Version 5.5 oder höher) verfügbar sein. Daraus geht auch hervor, dass *Subsonic* in der **Programmiersprache** *Java* umgesetzt ist. Weiters sind für den Betrieb ein Webbrowser und ein Musikplayer erforderlich.

Für die **Installation** von *Subsonic* muss die im heruntergeladenen Ordner `subsonic-3.0.zip` enthaltene Datei `subsonic.war`<sup>16</sup> in das entsprechende Verzeichnis von *Tomcat* (`./webapps`) kopiert und der Dienst mit dem Befehl

```
# sh startup.sh
```

in der Komandozeile gestartet werden. Danach kann *Subsonic* unter folgender Adresse aufgerufen werden (8080 ist der Standardport von *Tomcat*).

```
localhost:8080/subsonic
```

Es erscheint ein Login-Fenster (siehe Abbildung 3.5), bei dem man sich mit dem Benutzernamen `admin` und Passwort `admin` anmelden kann. UserInnen mit Administrator Rechten können Passwörter ändern und neue Benutzer-Accounts anlegen.



Abbildung 3.5: Subsonic Login-Fenster

Eine ausführliche **Dokumentation** für *Subsonic* ist weder auf der Projekt Webseite, noch auf der *Subsonic* Seite von *sourceforge.net* vorhanden - allerdings gibt es ein aktives Online-Forum, in dem diverse Fragen behandelt werden.<sup>17</sup>

Die **grafische Benutzer Oberfläche (GUI)** ist einfach gestaltet, wodurch eine gute Übersichtlichkeit gegeben ist (siehe Abbildung 3.6). Die Darstellung ist in vier Fenster aufgeteilt:

---

<sup>15</sup>„Apache Tomcat is the servlet container that is used in the official Reference Implementation for the Java Servlet and JavaServer Pages technologies. The Java Servlet and JavaServer Pages specifications are developed by Sun under the Java Community Process.“ [zit. APACHE TOMCAT, 24.04.2007 (Webseite) [APA07]]

<sup>16</sup>Dateinamenserweiterung `.war` = Java Web Archive [vgl. FILEExt, 24.04.2007 (Webseite) [FIL07]]

<sup>17</sup>*Subsonic* Forum: <http://www.activeobjects.no/subsonic/forum/index.php>

- \* ein Head-Fenster in dem sich die Menüleiste, eine Suchfunktion und die Auswahl für die indizierte Suche befinden
- \* ein Fenster am linken Rand, in welchem sämtliche Titel alphabetisch sortiert angezeigt werden
- \* ein Fenster am unteren Rand, welches die aktuelle Playlist anzeigt
- \* das Hauptfenster, in welchem Informationen zu den ausgewählten Alben oder Interpreten dargestellt werden

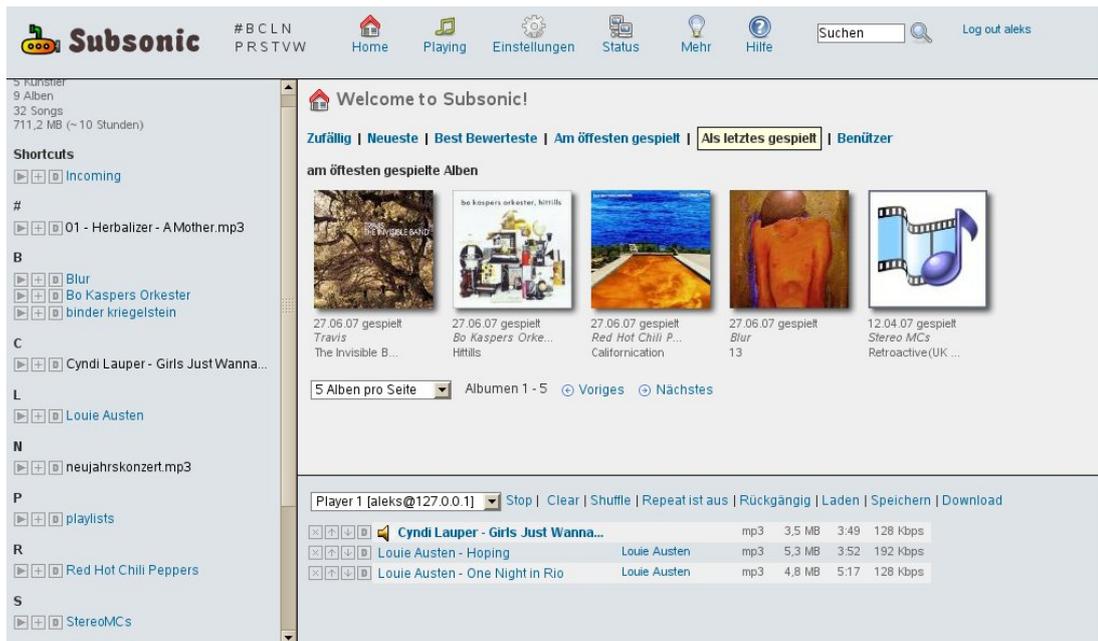


Abbildung 3.6: Subsonic Interface

**Betrieb** Zum Laden der Musikdaten braucht nur der Pfad des entsprechenden Verzeichnisses angegeben werden. Es besteht die Möglichkeit, mehrere **Verzeichnisse hinzuzufügen**, bei Bedarf können diese auch wieder deaktiviert werden. Auch einzelne Tracks können mit einer eigenen *upload* Funktion in beliebige Verzeichnisse geladen werden. Standardmäßig speichert *Subsonic* sie in ein eigenes *Incoming* Verzeichnis. Im Gegensatz zu *Ampache* (vgl. Abschnitt A) werden Titel unabhängig von deren Berechtigungen ins Verzeichnis geladen und gespielt.

Für die Datenverwaltung wird von *Subsonic* eine eigene **Datenbank** angelegt, die allerdings nur als Binärdatei in einem *Subsonic* Unterverzeichnis gespeichert wird. Daher kann diese außerhalb des Programmes nicht direkt geöffnet oder bearbeitet werden.

Alle vorhandenen Titel können problemlos einer bestehenden oder neu angelegten **Playlist** angehängt werden. Die Liste wird als `playlisttitle.m3u`<sup>18</sup> abgespeichert und kann mit einem beliebigen Player am Computer geöffnet werden. Änderungen in der Playlist werden als neues temporäres *.m3u* File gesendet.

<sup>18</sup>Dateinamenserweiterung *.m3u* = MP3 Playlist File [vgl. FILEExt, 25.04.2007 (Webseite) [FIL07]]

*Subsonic* ist Multiuser fähig - es können gleichzeitig mehrere UserInnen angemeldet sein und unabhängig voneinander Playlists abspielen oder verändern. Der Zugriff auf den Streaming-Server verläuft mit dem Webbrowser über die entsprechende URL.

Das Programm ist zwar optimiert für das Abspielen von MP3 Musikdaten, es können jedoch auch alle anderen **Audioformate** gespielt werden, die über HTTP gestreamt werden können. Für das **Streaming** werden alle Audiofiles in das *.mp3* Format transcodiert. Dafür ist im Programm eine Transcoding-Tabelle angelegt, in welcher die jeweiligen Befehle (für LAME<sup>19</sup> oder einen anderen MP3-Encoder, der über die Kommandozeile aufgerufen wird) angegeben sind. Diese „transcoding engine“ erlaubt Streaming von Dateien, die normalerweise nicht streambar sind. In den Standardeinstellungen finden sich die Formate *.mp3*, *.ogg*, *.aac*, *.wav*, *.wma* (weitere Formate können hinzugefügt werden). Es wird laufend in Echtzeit transcodiert, dadurch wird kein zusätzlicher Festplattenspeicher verbraucht. Allerdings ist optimales Streamen nur möglich, wenn der Aufwand für die Umkodierung möglichst gering ist. Grundsätzlich läuft das Streaming jedoch sehr stabil, auch bei größerem Rechenaufwand im Hintergrund.

Zum Abspielen können alle gängigen Musikplayer verwendet werden, die Streaming unterstützen. Laut Projektwebseite sind *Winamp*, *Windows Media Player*, *iTunes*, *MusicMatch*, *XMMS*, *VLC* und *Amarok* erfolgreich getestet. [vgl. SUBSONIC, 01.05.2007 (Webseite) [SUB07a]]

*Subsonic* bietet zusätzlich die Funktionen, CD-Cover von Alben und ID3-Tags von einzelnen Tracks anzuzeigen. Von bekannten Alben werden die Cover über die Amazon.com Webseite automatisch geladen, Bilder unbekannter CDs können manuell hinzugefügt werden. ID3-Tags können in einem eigenen Formular angezeigt und bearbeitet werden.

#### 3.4.4 VLC

##### getestete Version: 0.8.6a

*VLC* ist ein plattformunabhängiges Mediaserver Programm<sup>20</sup>, welches ohne zusätzlich **vorausgesetzter Software** lokal am Rechner oder über einen Webbrowser betrieben werden kann. Zusätzlich ist das in der **Programmiersprache** C umgesetzte Programm als Streaming-Server konfigurierbar.

Ausführliche **Dokumentationen** zu den einzelnen Funktionen sind auf der Projektwebseite zu finden.<sup>21</sup> Außerdem gibt es zu dem Projekt eine eigene Wiki<sup>22</sup> und ein online Forum<sup>23</sup>. Eine weitere Funktion, durch die sich *VLC* von den anderen Programmen hervorhebt, ist die Möglichkeit, auch Videos abzuspielen<sup>24</sup>.

---

<sup>19</sup>„LAME is an MPEG Audio Layer III (MP3) encoder licensed under the LGPL. [...] LAME originally stood for LAME Ain't an Mp3 Encoder.“ [zit. lame.sourceforge.net, 25.04.2007 (Webseite) [LAM07]]

<sup>20</sup>Unterstützte Plattformen: Windows, MacOSX, BeOS, Debian GNU/Linux, Ubuntu Linux, Mandriva Linux, Fedora Core, Familiar Linux, SUSE Linux, Red Hat Linux, Slackware Linux, ALT Linux, YOPY/Linupy, Zaurus, WinCE/PocketPC

<sup>21</sup><http://www.videolan.org/doc>

<sup>22</sup>VLC-Wiki: [http://wiki.videolan.org/Main\\_Page](http://wiki.videolan.org/Main_Page)

<sup>23</sup>VLC-Forum: <http://forum.videolan.org/>

<sup>24</sup>im Zusammenhang mit dieser Diplomarbeit ist diese Funktion aber nicht relevant [Anmerkung der Autorin]

Die **Installation** kann im SuSe-Linux durch das Hinzufügen der entsprechenden Paketquelle<sup>25</sup> relativ einfach durchgeführt werden. Danach kann das Programm aus dem Startmenü gestartet werden.

**Betrieb** Nach ausführlichen Tests zeigt es sich, dass *VLC* eher für die Verwendung als Mediaplayer ausgerichtet, und daher zum Verwalten von Musiksammlungen nicht besonders gut geeignet ist. Um Musikverzeichnisse aufzulisten und darin zu suchen, müssen diese zuerst in die *Playlist* importiert werden. Bei jedem Neustart des Programmes muss die Liste neu geladen, also der Pfad des **Musikverzeichnisses** neu eingegeben werden. Nach den vorhandenen Tracks kann nur durch Eingabe der Suchbegriffe, und nur innerhalb der Abspieliste, gesucht werden, indizierte Suche wird keine angeboten. Für den **Netzwerkbetrieb** sind eigene Funktionen im Programm enthalten.

Audiodaten in zahlreichen **Formaten**<sup>26</sup> können von verschiedenen Medien (CD, Datenträger, URL, ...) geöffnet und auch an anderen Orten gespeichert werden. Allerdings sind sehr viele Einstellungen und Klicks notwendig, bis Titel über den im Programm enthaltenen Player abgespielt werden können.

Die **grafische Oberfläche** ist im Design sehr schlicht gehalten, aber wirkt eher unübersichtlich. Die Menüs enthalten viele Unterpunkte für diverse Einstellungen zu einer Vielzahl an Funktionen. Der Player von *VLC* kann auch über den Webbrowser geladen werden. Da im Programm keine **Datenbank** enthalten ist, können keine Informationen zu den einzelnen Titeln gespeichert werden, auch das Darstellen von CD-Covers wird nicht angeboten. Zum Anzeigen von ID3-Tags ist zwar ein Befehl vorhanden, sie können jedoch nicht dargestellt werden.



Abbildung 3.7: VLC Interface

Für die grafische Benutzeroberfläche werden zwar zusätzliche Skins angeboten, jedoch verschwindet das Bedienfenster beim Wechsel. Obwohl das Abspielen der Playlist weiter läuft, kann die Oberfläche nicht wieder hergestellt werden, *VLC* muss dann über die Systemüberwachung (SuSe Linux) beendet werden.

<sup>25</sup><http://www.videolan.org/vlc/download-suse.html>

<sup>26</sup>MPEG (ES,PS,TS,PVA,MP3), ID3 tags, AVI, ASF, WMV, WMA, MP4, MOV, 3GP, OGG, OGM, Annodex, Matroska (MKV), WAV (including DTS), Raw Audio: DTS, AAC, AC3/A52, Raw DV, FLAC, FLV (Flash)

### 3.5 Gegenüberstellung der Ergebnisse

Nach einer ausführlichen Analyse der oben angeführten Programme, stellt sich heraus, dass die Tools *Ampache* und *Subsonic* für die Integration in das Medienserver-System AMSL geeignet sind. Die anderen beiden Programme entsprechen nicht den Anforderungen an ein solches System. Wenn auch bei *Jinzora* die Installationsroutine sehr professionell gelöst ist, entspricht das Programm im Betrieb nicht den Erwartungen. *VLC* ist in seiner Funktionalität für andere Bedürfnisse ausgerichtet, es bietet in diesem Zusammenhang wenige relevante Funktionen.

*Subsonic* bietet neben guter Übersichtlichkeit auch eine einfache Bedienung, es können problemlos mehrere Musikverzeichnisse von unterschiedlichen Speichermedien und -plätzen eingebunden werden und es kann zu mehreren Clients gleichzeitig gestreamt werden. Ein Nachteil ist jedoch, dass eigenständige Erweiterungen auf Grund der Programmierung schwer möglich sind. Bis auf die Möglichkeit, Informationen zu den einzelnen Tracks in der Datenbank zu speichern, sind alle wichtigen Funktionen enthalten.

Auch die Funktionalität von *Ampache* ist weitgehend überzeugend. Es können ebenfalls mehrere Musikquellen angegeben werden und die Benutzerverwaltung ist gut gelöst. Allerdings ist die Menüführung nicht optimal ausgeführt, auch die Einschränkung auf komprimierte Dateiformate (spielt kein *.wav*) sind als Nachteil zu sehen.

In Tabelle 3.1 werden die wichtigsten Punkte aus dem Vergleich der Programme noch einmal gegenübergestellt.<sup>27</sup>

---

<sup>27</sup>Im Punkt *Installationsaufwand* beziehen sich die Zeitangaben auf die reine Installationszeit des Programmes, ohne die Installation der Softwarevoraussetzungen. Der Punkt *Systemprivilegien* beschreibt die Berechtigung, mit der die Anwendung läuft.

### 3 OpenSource Komponenten für Medienserver

	<b>Ampache</b>	<b>Jinzora</b>	<b>Subsonic</b>	<b>VLC</b>
<b>Version (getestet)</b>	3.3.3	2.7.5	3.0	0.8.6a
<b>Programmiersprache</b>	PHP	PHP	Java	C
<b>Dokumentation</b>	Wiki, Forum	Forum	Forum	Doku, Wiki, Forum
<b>Software Voraussetzungen</b> (notwendig für Installation)	Webserver PHP $\geq$ 4.3.x MySQL $\geq$ 4.x	Webserver PHP $\geq$ 4.2	Java5, JSP z.B Tomcat $\geq$ 5.5	keine
<b>Installation</b>	Webinterface	Webinterface	automatisch	Paketquelle
<b>Installationsaufwand</b>	< 10 min	< 10 min	< 1 min	< 2 min
<b>GUI</b>	überladen	überladen	übersichtlich	übersichtlich
<b>einfache Bedienung</b>	+	-	+	-
<b>Musikverwaltung</b>	+	+	+	-
<b>Verzeichnisse merken</b>	+	?	+	-
<b>Systemprivilegien</b>	user	?	root	?
<b>Betrieb stabil</b>	+	-	+	-
<b>Menüführung</b>	inkonsequent	schlecht umgesetzt	übersichtlich	unübersichtlich
<b>Datenbank</b>	MySQL	MySQL	eigene	keine
<b>Musikimport</b>	Verzeichnisse	?	Verzeichnisse, Tracks	-
<b>mp3 / ogg / wav</b>	+ / + / -	+ / + / +	+ / + / +	+ / + / +
<b>indizierte Suche</b>	+	+	+	-
<b>Netzwerkfähigkeit</b>	+	+	+	+
<b>Playlistverwaltung</b>	+	+	+	+
<b>Playlistformat</b>	m3u	?	m3u	?
<b>Playlist aktualisieren</b>	+	?	+	?

**Tabelle 3.1:** Gegenüberstellung der OpenSource Programme (Funktionen, die mit ? gekennzeichnet sind, konnten auf Grund von Instabilität nicht getestet werden.)

### Zusammensetzung des Medien-Server Systems

---

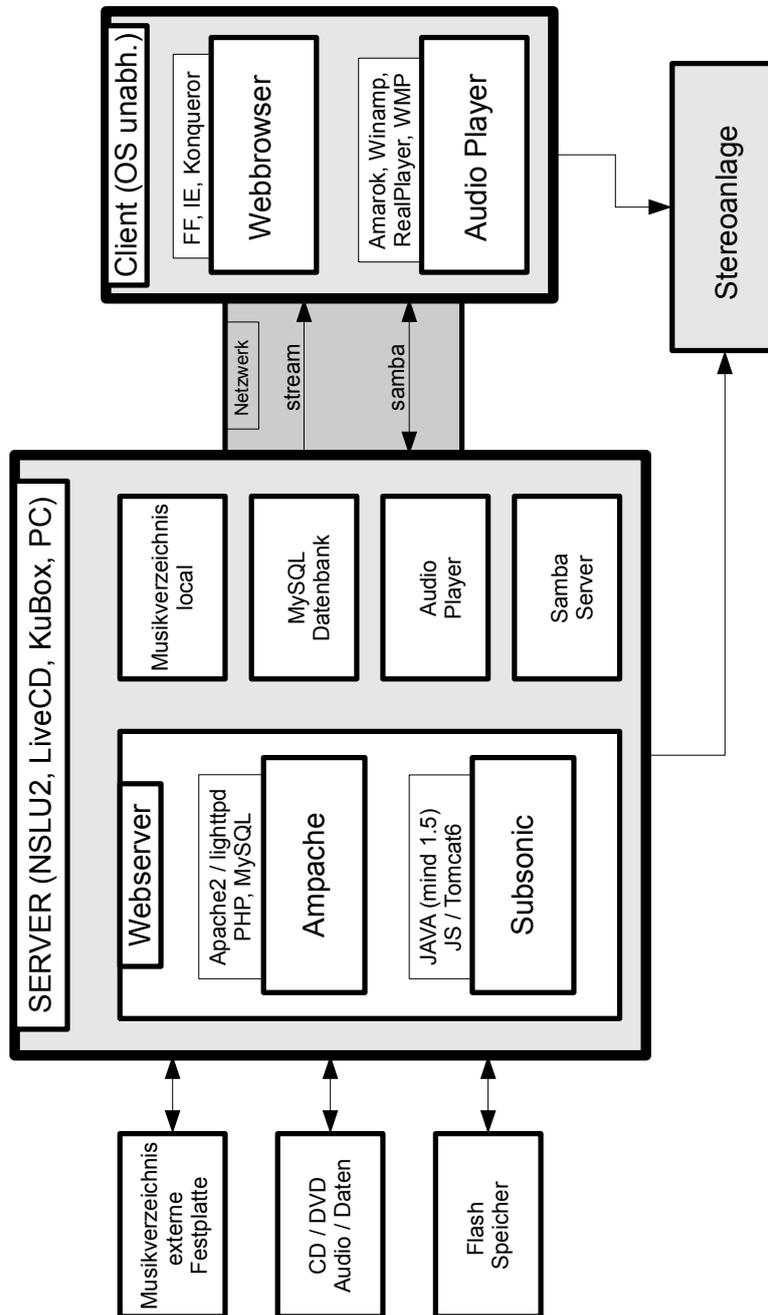
#### 4.1 Strukturdiagramm

Die Zusammensetzung des AMSL wird in Abbildung 4.1 in Form eines Blockschalt-diagramms veranschaulicht. Es wird hier der Zugriff eines Clients auf den Server (einer der angeführten) mittels Webbrowser (Firefox (FF), Internet Explorer (IE), Konqueror oder andere) über das lokale Netzwerk (Ethernet oder WLAN) dargestellt. Auf den AMSL können selbstverständlich mehrere Clients gleichzeitig zugreifen.

Die Webapplikation *Ampache* ist eine Anwendung am Webserver (*Apache httpd* oder *lighttpd* inklusive PHP und MySQL), *Subsonic* am *Java Servlet (JS) Tomcat* (in der Version 6), das unter *Java* (ab Version 1.5) läuft.

Musikdaten werden von lokalen oder externen Speichermedien (Festplatte, CD/DVD, Flashspeicher) bezogen und via Stream oder Sambazugriff zum Client übertragen. Verfügbare Daten können am Server oder Client mit einem Audio Player, wie es zum Beispiel *Amarok*, *Winamp*, *RealPlayer* oder *Windows Media Player (WMP)* sind, wiedergegeben werden. Mit einer entsprechenden Verbindung zur Stereoanlage können die Musiktitel auch dort abgespielt werden.

Wie im Diagramm zu sehen ist, besteht das System aus einer Vielzahl von Komponenten. Dadurch ergeben sich teilweise mehrere Umsetzungsmöglichkeiten für einzelne Dienste. Manche Systeme sind jedoch auf Grund ihrer Hardware auf einige der Anwendungen beschränkt. Diese Möglichkeiten und die Funktionen der einzelnen Komponenten des Blockschaltbildes werden in den folgenden Abschnitten genauer behandelt.



**Abbildung 4.1:** Strukturdiagramm des Medienservers als Blockschaltbild (inklusive eines über das Netzwerk verbundenen Clients)

## 4.2 Das Server System

Um eine einfache Inbetriebnahme des Medienservers zu ermöglichen, bei der keine aufwändigen Installationen mehr nötig sind, wird als Endergebnis ein komplett vor-konfiguriertes System erstellt. Hierfür gibt es drei Technologien, die in Frage kommen würden:

- \* NSLU2, ein Network Storage Link von Linksys<sup>1</sup> (Hostname: *Schluck*<sup>2</sup>)
- \* eine Live-CD
- \* oder ein virtuelles Image (Hostname: *KuBox*<sup>3</sup>)

Deren Funktionen, Voraussetzungen und die wichtigsten Schritte während des Installationsvorgangs werden in diesem Abschnitt vorgestellt. Die beschriebenen Komponenten sind nicht zwingend an das jeweilige System gebunden, es ergeben sich jedoch auf Grund der zur Verfügung stehenden Hardware einige Einschränkungen (vor allem bei *Schluck*).

Natürlich kann das System auch auf einem eigenständigen PC betrieben werden. Hierfür müssen die benötigten Programme, entsprechend der Auflistung der verwendeten Programmpakete im Kapitel 5.1.2, installiert werden.

### 4.2.1 Linksys NSLU2



Abbildung 4.2: Linksys NSLU2

*NSLU2* (Abbildung 4.2), ist ein Network Storage Link von Linksys, mit dem USB Speichermedien ins lokale Netzwerk freigegeben werden können. Die mitgelieferte Software entspricht zwar nicht den Anforderungen, jedoch können verschiedene Firmwareupdates aus unterschiedlichen Linux Distributionen geladen werden. Dadurch können unter anderem beliebige Serverdienste über dieses Gerät betrieben werden. [vgl. Linksys, 28.06.2007 (Webseite) [LIN07a]]

Als Grundlage für *Schluck* wird die Linuxdistribution *Debian Etch*<sup>4</sup> verwendet, die

---

<sup>1</sup><http://www.linksys.com/>

<sup>2</sup>Im Folgenden werden die Hostnamen zur Bezeichnung der jeweiligen Server Systeme verwendet. Der Name *Schluck* entstand als Weiterentwicklung der Typenbezeichnung NSLU2.

<sup>3</sup>Der Name *KuBox* ergibt sich aus dem Betriebssystem *Kubuntu* und der Virtualisierungssoftware *VirtualBox*, auf welchem er betrieben wird.

<sup>4</sup>[www.de.debian.org](http://www.de.debian.org)

nach dem entsprechenden Firmwareupdate auf dem *NSLU2* betrieben werden kann. Eine Beschreibung der Installationsroutine ist im Anhang B zu finden.

*NSLU2* wird in Kombination mit einer USB Festplatte eingesetzt, auf der sich das Betriebssystem und das Musikverzeichnis befindet. Über eine RJ45 Verbindung wird er mit dem WLAN-Router verbunden, der wiederum die Daten im internen Netzwerk für alle Clients zur Verfügung stellt.

Anstatt des *Apache httpd* Webservers wird *lighttpd* eingesetzt, dadurch kann *Ampache* am *Schluck* Ressourcen schonender betrieben werden.

Neue Musikdaten können über den am *Schluck* laufenden *Samba-Server* in das Musikverzeichnis auf der USB-Festplatte hinzugefügt werden.

Der Vorteil von *Schluck* ist, dass kein eigener Computer benötigt wird, um das System zu betreiben. Auf Grund ihrer Größe können die Komponenten sehr platz sparend untergebracht werden, auch der Energiebedarf ist im Vergleich zu einem Computer sehr gering. Allerdings können der verfügbare Arbeitsspeicher von nur 32 MB und die verhältnismäßig geringe Prozessorleistung zu Einschränkungen in der Performance führen (vor allem bei der Verwendung von mehreren Clients). Auf Grund dieser ist es auch nicht möglich, *Tomcat6* (eine Voraussetzung für *Subsonic*) zu betreiben. Alternative *Java* Servlet-Container wurden getestet, haben jedoch bis jetzt noch keine zufriedenstellenden Ergebnisse geliefert.

Daher ist der derzeitige Stand von *Schluck*, dass *Ampache* entsprechend funktioniert, für *Subsonic* jedoch zur Zeit unter *Debian-Etch* mit ARM-Prozessor noch nicht die Softwarevoraussetzungen erfüllt werden können, um es auf dem *NSLU2* betreiben zu können.

### 4.2.2 Live-CD

Vor allem zahlreiche Linux Distributionen werden im Internet als *Live-CD*<sup>5</sup> Images angeboten. Diese sind bereits vorkonfiguriert und können direkt von der CD gebootet werden.

Im Gegensatz dazu gibt es auch die Möglichkeit, aus einer nach den eigenen Anforderungen konfigurierten Linux Distribution eine *Live-CD* zu erzeugen. Hierfür kann zum Beispiel *Linux Live scripts* verwendet werden. Wahlweise ist auch das Generieren eines bootfähigen Images für andere Speichermedien, wie zum Beispiel USB-Stick oder Flash-Card, möglich. [vgl. LINUX-LIVE, 28.06.2007 (Webseite) [LIN07b]]

Eine Anleitung für die Erzeugung einer *Live-CD* ist auf der Webseite von Techchannel zu finden.<sup>6</sup>

Auch wenn die *Live-CD* für den Betrieb der AMSL eine durchaus geeignete Technologie bietet, wird im Rahmen dieser Diplomarbeit darauf nicht näher eingegangen. Der Focus der Arbeit ist auf das System der *KuBox* gerichtet.

---

<sup>5</sup>„A LiveCD is a CD that is able to boot your system into a 'live', working operating system, independent of anything on your hard drive.“ [zit. LFS LiveCD, 10.07.2007 (Webseite) [LFS07]]

<sup>6</sup>[http://www.tecchannel.de/pc\\_mobile/linux/438588/](http://www.tecchannel.de/pc_mobile/linux/438588/)

### 4.2.3 Virtuelle Systeme

Zum Betreiben von virtuellen Betriebssystemen auf einem PC werden zahlreiche Produkte angeboten, die teilweise unter einer OpenSource Lizenz stehen, aber zum Großteil nur als proprietäre Systeme zur Verfügung stehen. (Eine Auflistung der Projekte ist auf der englischen Wikipedia Seite unter dem Stichwort „Comparison of virtual machines“ zu finden.<sup>7</sup>) [vgl. WIKIPEDIA, 09.07.2007 (Webseite) [WIK07]]

Auf der Projektwebseite von *VMware*<sup>8</sup>, einem der bekanntesten virtuellen Systeme, wird die Funktionsweise wie folgt beschrieben: „Virtualization allows multiple virtual machines, with heterogeneous operating systems to run in isolation, side-by-side on the same physical machine. Each virtual machine has its own set of virtual hardware (e.g., RAM, CPU, NIC, etc.) upon which an operating system and applications are loaded. The operating system sees a consistent, normalized set of hardware regardless of the actual physical hardware components.“ [zit. VMWARE, 09.07.2007 (Webseite) [VMW07]]

Auf Grund der Zielsetzung, für den AMSL ausschließlich OpenSource Software zu verwenden, wird für dessen Betrieb in einem virtuellen System *VirtualBox*<sup>9</sup> von *Innotek* als Grundlage ausgewählt. (siehe auch Abbildung 4.3) Es wird sowohl für Windows Rechner, als auch für Mac OS X und die gängigsten Linux Distributionen als *host* Betriebssystem (OS)<sup>10</sup> angeboten.



**Abbildung 4.3:** VirtualBox von Innotek

In der sehr übersichtlich gestalteten GUI der *VirtualBox* können neue virtuelle Maschinen angelegt und konfiguriert werden, wobei unterschiedliche *guest* Betriebssysteme<sup>11</sup> installiert werden können. Eine Liste der unterstützten *guest OS*, sowie eine genaue Anleitung für die Installation und Inbetriebnahme, ist im User Manual der *VirtualBox* zu finden. [vgl. INNTEK, 2007 (S. 12) [INN07]]

Für AMSL wird als *guest OS* die Linux Distribution *Kubuntu* in der Version 7.04 verwendet. Das Musikverzeichnis befindet sich ebenfalls im virtuellen System, allerdings können zusätzlich aus dem *host OS* CD/DVD Laufwerke eingebunden und USB-Controller aktiviert werden. Somit können, neben freigegebenen Ordnern aus dem *host*

---

<sup>7</sup>Hier wird ausnahmsweise auf Wikipedia verwiesen, da sonst leider keine übersichtliche Gegenüberstellung der vorhandenen Programme zu finden ist. [Anmerkung der Autorin]

<sup>8</sup><http://www.vmware.com>

<sup>9</sup><http://www.virtualbox.org/wiki/VirtualBox>

<sup>10</sup>„Host operating system (host OS): the operating system of the physical computer where VirtualBox is running.“ [zit. INNTEK, 2007 (S. 8) [INN07]]

<sup>11</sup>„Guest operating system (guest OS): the operating system that is running inside the virtual machine.“ [zit. INNTEK, 2007 (S. 9) [INN07]]

OS, auch von diesen Speichermedien Musikverzeichnisse verwendet werden.

Ein Vorteil vom virtuellen System ist, dass das guest OS fertig vorinstalliert und konfiguriert werden und dann auf jedem beliebigen Rechner, der *VirtualBox* installiert hat, betrieben werden kann. Als Systemvoraussetzung für das host OS gilt ein ausreichend großer Arbeitsspeicher, der ab dem Zeitpunkt des Startens zur Verfügung stehen muss. Die Größe ist abhängig vom installierten guest OS.

Als einziger Nachteil der *VirtualBox* ist in diesem Zusammenhang zu erwähnen, dass es in diesem System nicht möglich ist, Audio-CDs zu verwenden. Durch einen Workaround, der in Kapitel 5.2.1 genauer beschrieben ist, kann das Problem jedoch umgangen werden.

### 4.3 Die Software im Hintergrund

In diesem Abschnitt werden die zusätzlichen Software Voraussetzungen für den Betrieb des Medienservers angeführt. Hierbei wird zwischen dem Server, auf dem das System läuft und dem Client, der das Service über das interne Netzwerk nutzt, unterschieden.

Wie es auch im Strukturdiagramm (siehe Abbildung 4.1) abgebildet ist, kann der Server auf einem der in Abschnitt 4.2 vorgestellten Systeme betrieben werden. Abhängig davon wird eines der in Abschnitt 4.3.1 vorgestellten Betriebssysteme installiert. Des Weiteren werden von den Medienserverprogrammen *Ampache* und *Subsonic* Webserverdienste und eine MySQL Datenbank für die Verwaltung der Musikdaten benötigt. Ein zusätzlicher Samba Server ermöglicht den Datentransfer zwischen Server und Client Rechner. Die Musikdaten werden entweder in einem lokalen Musikverzeichnis abgespeichert oder können von einem externen Datenspeicher (Festplatte, CD oder Flashspeicher) verwendet werden. Zum direkten Abspielen von Musikstücken über den Server wird zusätzlich ein Media-Player benötigt.

Auf dem Client Rechner kann ein beliebiges Betriebssystem installiert sein. Als Programm Anforderungen gelten hier ein beliebiger Webbrowser und ein streaming-fähiger Musikplayer. Die einzelnen Komponenten werden in den folgenden Abschnitten genauer erläutert.

#### 4.3.1 Das Betriebssystem

Für den Server wird als Betriebssystem eine Linux Distribution installiert, mit welcher der Anforderung an ein OpenSource System entsprochen wird. Im Speziellen wird je nach eingesetzter Hardware *SuSe 10.2* (Betrieb am normalen PC), *Debian Etch* (Betrieb am *NSLU2* von *Linksys*) oder *Kubuntu 7.04* (Betrieb in *VirtualBox*) verwendet<sup>12</sup>. Die benötigte Software kann über die entsprechenden Paketquellen relativ einfach installiert und aktualisiert werden.

Am Client-Rechner kann jedes beliebige Betriebssystem<sup>13</sup> verwendet werden, über dessen Webbrowser die Verbindung zum Medienserver aufgebaut wird.

---

<sup>12</sup>Vergleiche dazu die Projektwebseiten [de.opensuse.org](http://de.opensuse.org), [www.de.debian.org](http://www.de.debian.org) und [www.kubuntu.org](http://www.kubuntu.org)

<sup>13</sup>getestet wurde das System auf SuSe Linux 10.2 und WinXP

In diesem Zusammenhang wird als Betriebssystem für den Clientrechner eine spezielle Linux Distribution für den Multimediabereich empfohlen: *Ubuntu Studio 7.04*.<sup>14</sup> Auf der Projektwebseite wird *Ubuntu Studio* wie folgt beschrieben: „Ubuntu Studio is a multimedia editing/creation flavor of Ubuntu. It’s built for the GNU/Linux audio, video, and graphic enthusiast or professional.“ [zit. UBUNTU, 06.08.2007 (Webseite) [UBU07]]

Eine ausführliche Installationsanleitung ist auf der Webseite von *howtoforge* zu finden.<sup>15</sup>

### 4.3.2 Der Webserver

Darunter versteht man Serveranwendungen, die Daten über das *Hypertext Transfer Protocol (HTTP)*, beziehungsweise über das verschlüsselte *Secure Hypertext Transfer Protocol (HTTPS)* zur Verfügung stellen.<sup>16</sup> Für die Darstellung von dynamischen Inhalten werden Skripte (PHP, .NET, etc.) oft direkt am Webserver ausgeführt, deren Ergebnisse als HTML-Code an den Webbrowser gesendet werden.<sup>17</sup>

Um mit einem Webbrowser über das Netzwerk auf die online Applikationen des Medienservers zugreifen zu können, muss dort ein Webserver gestartet sein. Je nach Art der online Applikation müssen eventuell zusätzliche Komponenten installiert, beziehungsweise aktiviert werden, wie zum Beispiel PHP oder SSL. Die genauen Anforderungen der verwendeten Programme sind bereits in Abschnitt A für *Ampache* und in Abschnitt A für *Subsonic* angeführt.

In diesem Projekt werden zwei verschiedene Arten von Webserver eingesetzt. Einerseits ein klassischer Webserver wie *Apache httpd* oder *lighttpd*, der das für *Ampache* benötigte PHP und MySQL zur Verfügung stellt. Andererseits ein *Java* Servlet-Container, wie es zum Beispiel *Tomcat*<sup>18</sup> ist. Damit kann *Java*-Code auf einem Webserver ausgeführt werden, was als Voraussetzung für die Verwendung von *Subsonic* gilt. Als Alternative zu *Tomcat* wird am *Schluck* der etwas Ressourcen schonendere *Jetty Webserver* eingesetzt<sup>19</sup>.

### 4.3.3 Samba Server

*Samba* ist ein Unix Software Paket, welches mit dem Server Message Block (SMB) Protokoll arbeitet. Der Datei- und Printserver regelt den Netzwerkzugriff von Linux und Windows Client Rechnern auf Linux oder Unix basierten Servern. „Samba uses the TCP/IP protocol that is installed on the host server. When correctly configured, it allows that host to interact with a Microsoft Windows client or server as if it is a

---

<sup>14</sup>Download und Informationen zu Ubuntu Studio: <https://wiki.ubuntu.com/UbuntuStudio>

<sup>15</sup>Installationsanleitung für Ubuntu Studio: [http://www.howtoforge.com/ubuntustudio\\_7.04](http://www.howtoforge.com/ubuntustudio_7.04) und Erweiterungen zum „perfekten Desktop“ [http://howtoforge.org/the\\_perfect\\_desktop\\_ubuntustudio7.04](http://howtoforge.org/the_perfect_desktop_ubuntustudio7.04)

<sup>16</sup>ausführliche Beschreibungen der Protokolle sind auf javvin.com zu finden: <http://www.javvin.com/protocolHTTP.html> für HTTP und <http://www.javvin.com/protocolHTTPS.html> für HTTPS

<sup>17</sup>eine ausführliche Beschreibung des HTML-Codes ist unter der Adresse <http://www.w3.org/TR/html4/intro/intro.html#h-2.2> zu finden

<sup>18</sup>siehe auch <http://tomcat.apache.org/>

<sup>19</sup>siehe auch <http://mortbay.org>

Windows file and print server.“ [zit. SAMBA, 26.07.2007 (Webseite) [SAM07]] Eine ausführliche Beschreibung ist in dem Buch „Using Samba“<sup>20</sup> zu finden.

Im Medienserver wird *Samba* vor allem für das Hinzufügen von Musikdaten vom Client Rechner auf den Host benötigt. Es können selbstverständlich auch Daten vom Host auf den Client Rechner kopiert werden, allerdings gibt es dafür bei *Ampache* und *Subsonic* auch eine Download Funktion für einzelne Songs.

#### 4.3.4 Der Webbrowser

Die Funktion eines Webbrowsers ist die Kommunikation mit einem Webserver und die Interpretation und Visualisierung der übermittelten Daten. Die Kommunikation läuft hauptsächlich über das in Abschnitt 4.3.2 bereits erwähnte HTTP und HTTPS.

Übertragene Daten bestehen meist aus HTML-Code, der mit Technologien wie beispielsweise CSS oder clientseitigen Scriptsprachen (zum Beispiel JavaScript) ergänzt wird. Es werden vielfach, abhängig vom verwendeten Browser, weitere Datenformate (zum Beispiel XML oder DHTML) und Protokolle (zum Beispiel FTP oder SMB) unterstützt. Mittlerweile sind auch multimediale Inhalte wie Audiodaten, Videos oder Flash-Animationen wichtige Bestandteile von Webseiten. In letzter Zeit gewinnen dynamische Webapplikationen, auch bekannt unter dem Schlagwort *Web-2.0* immer mehr an Bedeutung. Grundlage dafür bildet die Kombination von serverseitigen Scriptsprachen (zum Beispiel PHP) und browserseitigen Scriptsprachen. Dadurch wird es möglich, nur relevante Inhalte der Webseiten neu zu laden oder durch andere Inhalte zu ersetzen. Hiermit können innerhalb des Browsers Anwendungen realisiert werden, die gewöhnlichen Desktopapplikationen ähnlich sind.

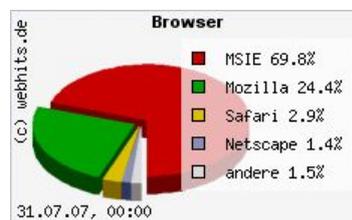


Abbildung 4.4: WebHits Browserstatistik

Die gängigsten Webbrowser sind der *Internet Explorer* von Microsoft (Standardbrowser auf den meisten Windows Rechnern) und der OpenSource Browser *FireFox* von Mozilla. In Abbildung 4.4 wird die Verteilung der Marktanteile der am häufigsten verwendeten Browser gezeigt. Details zur Verbreitung der einzelnen Browser sind im Web-Barometer von WebHits ersichtlich.<sup>21</sup> Hierzu ist noch anzumerken, dass Browserstatistiken immer nur ein Anhaltswert sein können und niemals als Absolutwert gesehen werden dürfen. Es ist nicht möglich, alle Zugriffe im Web vollständig zu erfassen.

<sup>20</sup>Webadresse: [http://us3.samba.org/samba/docs/using\\_samba/toc.html](http://us3.samba.org/samba/docs/using_samba/toc.html) [vgl. TS et. al, 2003 (Kap. 1) [TECB03]]

<sup>21</sup><http://www.webhits.de/deutsch/index.shtml?webstats.html>

Als Webbrowser für den AMSL wird vorzugsweise *Firefox 2.0* verwendet, da dieser ein plattformunabhängiges OpenSource Produkt ist, der sowohl für Windowssysteme, als auch für Linux und Mac OS angeboten wird. Weitere Browser aus dem OpenSource Bereich sind *Konqueror*, der Standardbrowser von KDE-Linux, und der auf dessen Kernel aufbauende *Safari*, Standardbrowser von Mac OS. Abgesehen davon können auch proprietäre Browser wie *Microsoft Internet Explorer* oder *Opera* eingesetzt werden.

### 4.3.5 Audio Player

Der Audio Player dient dazu, Musikdaten von unterschiedlichen Quellen, die in verschiedenen Dateiformaten abgespeichert sein können, wiederzugeben. Einige der häufigsten Audio Player sind *Amarok*, *VLC*, *Winamp*, *RealPlayer*, *Windows Media Player (WMP)*, *iTunes* oder *MusicMatch*. Viele der Programme bieten zahlreiche Zusatz Features wie zum Beispiel:

- \* lokale Musikverwaltung
- \* CD-Ripper
- \* Playlist Erstellung
- \* CD Cover Download
- \* CDs brennen
- \* Importieren von CDs
- \* Internet Radio
- \* Playlists für mobile Player erstellen und übertragen
- \* CDDDB-Lookup
- \* Songtext Anzeige
- \* Streaming

Für den AMSL ist vor allem die Funktion des Streamens von Bedeutung. Hiermit kann Musik, die an einem entfernten Rechner im Netzwerk gespeichert ist, abgespielt werden kann. Der Medienserver schickt hierfür eine Playlist, die im Format *.m3u* abgespeichert ist, an den Audio Player. (Streamen siehe auch Abschnitt 4.4.3)

### 4.3.6 Brennen von CDs oder DVDs

Heutzutage kann davon ausgegangen werden, dass die meisten Computer mit einem CD/DVD Laufwerk ausgestattet sind, das eine Brennfunktion bietet.<sup>22</sup> Je nach Betriebssystem stehen unterschiedliche Programme für das Brennen von CDs oder DVDs zur Verfügung. Neben der häufig eingesetzten proprietären Brennsoftware *Nero*<sup>23</sup> gibt

---

<sup>22</sup>Details zum CD-Brenner: [http://www.itwissen.info/definition/lexikon//\\_\\_cd%20writer\\_cd-brenner.html](http://www.itwissen.info/definition/lexikon//__cd%20writer_cd-brenner.html)

<sup>23</sup>Nero burning: <http://www.nero.com/eng/Downloads.html>

es für Windows Betriebssysteme auch Freeware<sup>24</sup> Brenner, wie zum Beispiel *CDBurnerXP Pro*<sup>25</sup>. Unter Linux ist eines der bekanntesten Produkte die OpenSource Brennsoftware *K3B*<sup>26</sup>.

Im AMSL können über den CD/DVD Brenner Audio-CDs mit den Lieblingstiteln oder MP3-Daten CDs/DVDs erstellt werden. (siehe dazu auch Abschnitt 4.5.4 auf Seite 43f.)

## 4.4 Netzwerkverbindung

### 4.4.1 Verbindung zwischen Server und Client

Im privaten Bereich werden für den Betrieb eines lokalen Netzwerks, im Englischen als *local area network (LAN)* bezeichnet, hauptsächlich zwei Arten von Netzwerkverbindungen eingesetzt - kabelgebundene *Ethernetverbindungen* oder *Wireless-LAN* Verbindungen (WLAN).<sup>27</sup> Auf derartige Netzwerke kann meist aus dem Internet nicht direkt zugegriffen werden. Verbindungen in das Internet werden über Router hergestellt, die häufig auch als WLAN Access-Point Verwendung finden. Oft ist der vorrangige Zweck des LANs, mehrere Rechner gleichzeitig mit dem Internet zu verbinden. Serverdienste werden in privaten Netzwerken nur selten verwendet.

Der AMSL ist so konzipiert, dass er eine zentrale Speichermöglichkeit und Datenverwaltung für große Musiksammlungen bietet, auf die von allen Rechnern im Netzwerk zugegriffen werden kann. *Ampache* und *Subsonic* können von den Clients über den Webbrowser erreicht werden. Von dort aus kann Musik über einen Webstream empfangen oder über die Download Funktion lokal auf dem Rechner gespeichert werden. Der zusätzlich installierte Samba-Server ermöglicht das Kopieren von größeren Datenmengen.

### 4.4.2 Datenraten

„Der Begriff Datenrate wird synonym für die Übertragungsgeschwindigkeit und die Übertragungsrate verwendet. Es handelt sich dabei um die Anzahl der Informationseinheiten, die in einer bestimmten Zeit über das Übertragungsmedium, eine Schnittstelle oder den Übertragungskanal übertragen werden. Die Datenrate wird in Bit pro Sekunden resp. in kbit/s, Mbit/s oder Gbit/s angegeben.“ [zit. ITWISSEN, 02.08.2007 (Webseite) [ITW07b]]

Je nach verwendeter Netzwerktechnologie steht für Verbindungen zwischen Netzwerkteilnehmern eine maximale Datenrate zur Verfügung. Sind die Rechner über WLAN verbunden, teilen sich alle TeilnehmerInnen die maximale Übertragungsrate

---

<sup>24</sup>Freeware like „free as in beer“ - im Gegensatz dazu: „*Free software* is a matter of liberty, not price. To understand the concept, you should think of *free* as in *free speech*, not as in *free beer*.“ [zit. GNU, 06.08.2007 (Webseite) [GNU07]]

<sup>25</sup>Projektwebseite von CDBurnerXP Pro: <http://www.cdburnerxp.se/features.php>

<sup>26</sup>Infos zu K3B auf der Projektwebseite: <http://k3b.plainblack.com/>

<sup>27</sup>Detaillierte Informationen zu diesen Technologien sind auf der Webseite <http://www.itwissen.info/> unter den Schlagworten „Ethernet“ und „WLAN“ zu finden.

von 54 Mbit/s.<sup>28</sup> Bei einer Verbindung über Ethernet, mittels Switch als Netzwerkknoten, können einzelne TeilnehmerInnen des Netzes kommunizieren, ohne das restliche Netz zu belasten. Ziel ist es, die maximal vorhandene Datenübertragungsrate möglichst effizient zu nutzen. Heute übliche Ethernetverbindungen nutzen eine Übertragungsrate 100 Mbit/s oder 1 Gbit/s, Verbindungen über ein 10 Mbit/s Netzwerk verlieren immer mehr an Bedeutung.

Die theoretischen Maximalwerte werden in der Praxis jedoch selten erreicht. Durch mitgesendete Protokollheader und Prüfsummen verringert sich die tatsächlich verfügbare Datenrate, die Werte sind abhängig von der verwendeten Technologie.

Bei WLAN Verbindungen ist die Differenz noch größer als bei Ethernet, die Bruttodatenrate beträgt hier 54 MBit/s. Unter optimalen Bedingungen können damit maximal 25 MBit/s netto übertragen werden. Bei schlechter Verbindungsqualität verringern viele Access Points die Geschwindigkeit auf bis zu 1 MBit/s brutto. [vgl. NETWORK LAB, 03.08.2007 (Webseite) [NET07]]

Der Betrieb des Medienservers stellt gewisse Anforderungen an die Übertragungskapazität des Netzwerks, um eine störungsfreie Wiedergabe der Audiodaten (streaming oder directplay) zu ermöglichen.

Getestet wurde AMSL mit sechs gleichzeitig laufenden Clients in einem 54 Mbit/s WLAN. Dabei konnte auf dem host Rechner eine durchschnittliche nach Außen gehende Datenrate von 1,2 Mbit/s gemessen werden. Hier waren keine Beeinträchtigungen in der Übertragungsqualität festzustellen.

Aus den gemessenen Daten geht hervor, dass innerhalb eines WLANs grob geschätzt bis zu 20 Clients gleichzeitig betrieben werden könnten (bei einer durchschnittlichen Datenrate von 1 Mbit/s für jeden Client). Diese Anzahl an Clients wird innerhalb eines Heimnetzwerks vermutlich selten erreicht werden. Demnach stellt die Datenrate im Betrieb auf einem 100 Mbit/s Ethernet LAN im Heimnetzwerk keine Einschränkung dar.

### 4.4.3 Streaming versus Download

Grundsätzlich werden zwei Arten von Datenempfang über das Netzwerk unterschieden - der *Download* und der *Stream*. Beim *Download* wird die empfangene Datei auf den Client-Rechner kopiert und dort gespeichert. Während beim konventionellen Download, die Datei erst verwendet werden kann, wenn sie vollständig am Client-Rechner gespeichert ist, kann beim progressiven Download bereits während des Herunterladens auf die Datei zugegriffen werden. „It is important to note that streaming is not a progressive download, where the display starts as soon as enough of the data is buffered locally while the remainder of the file is being retrieved and stored on the local disk [...].“ [zit. DASHTI et. al, 2003 (S. 4) [DKSZ03]]

---

<sup>28</sup>Die Bitrate von 54 Mbit/s gilt für den 2003 definierten WLAN-Standard IEEE 802.11g. Dieser häufig verwendete Standard arbeitet im 2,4 GHz Band. Weitere Informationen zu WLAN-Standards sind unter folgender Adresse zu finden: [http://www.itwissen.info/definition/lexikon/\\_\\_\\_ieee%20802.11\\_802.11.html](http://www.itwissen.info/definition/lexikon/___ieee%20802.11_802.11.html). Sämtliche Tests des AMSL im lokalen Netzwerk wurden über diesen Standard durchgeführt.

Bei *Streaming*<sup>29</sup> hingegen werden die empfangenen Daten am Client direkt wiedergegeben und daher nicht auf der Festplatte gespeichert. „Da die Wiedergabe der Daten gleichzeitig mit dem Empfang stattfinden soll, muss eine ausreichende Datenübertragungsrate [...] zur Verfügung stehen.“ [zit. WEBHOSTING, 31.07.2007 (Webseite) [WEB07]]

Im Buch „*Streaming Media Server Design*“ werden einige Vor- und Nachteile der Technologie aufgelistet. [vgl. DASHTI et. al, 2003 (S. 4) [DKSZ03]]

- + kein Warten auf den Download
- + es werden lokal keine Kopien der Datei gespeichert, wodurch einerseits keine Copyright Bestimmungen verletzt werden können und andererseits keine Speicherplatz Anforderungen an den Client gestellt werden
- + dadurch werden Live-Übertragungen ermöglicht
- Server und Netzwerk müssen real-time fähig sein
- Verlorene oder defekte Pakete im Datenstrom oder Zeitüberschreitungen können zu Störungen in der Wiedergabe führen.

### Stream on Demand oder Live

„A media stream can be on demand or live. On-demand streams are stored on a server for a long period of time, and are available to be transmitted at a user’s request. Live streams are only available at one particular time, as in a video stream of a live sporting event.“ [zit. JAVVIN, 01.08.2007 (Webseite) [JAV07a]]

Die beim AMSL verwendete Streaming Methode ist das *Stream on Demand* - hier wird auf Anfrage des Clients eine Playlist an dessen Musik Player gesendet. Die entsprechenden Musikdaten werden darauf hin vom Server mittels *Unicast*<sup>30</sup> Verfahren über das Netzwerk zur Verfügung gestellt. Das heißt die Daten werden nur zu diesem Client übertragen, sollten andere Clients die selben Daten anfordern, werden diese mehrmals separat verschickt. [vgl. JAVVIN, 01.08.2007 (Webseite) [JAV07b]]

Aus der *KuBox* können im Heimnetzwerk Streams von MP3-Daten (mit einer Bitrate von bis zu 320 kbps) in Echtzeit empfangen werden (getestet mit bis zu sechs Clients in 54 Mbit/s WLAN).

## 4.5 Datenverwaltung

Die Datenmengen einer immer weiter wachsenden digitalen Musiksammlung erfordern eine systematische und möglichst effiziente Speicherung. Hieraus ergibt sich einerseits die Überlegung einer sinnvollen, einheitlichen Ordnerstruktur (Abschnitt 4.5.2) und Dateinamenvergabe im Musikverzeichnis (siehe Abschnitt 4.5.3).

---

<sup>29</sup>weitere Informationen zu dieser Technologie sind unter folgender Adresse zu finden: [http://www.itwissen.info/definition/lexikon//\\_streaming%20media\\_streaming-media.html](http://www.itwissen.info/definition/lexikon//_streaming%20media_streaming-media.html)

<sup>30</sup>im Gegensatz dazu werden bei *Multicast* die Daten vom Server an mehrere Rechner gleichzeitig gesendet - dieses Verfahren wird zum Beispiel bei Webstreams von Radios verwendet

Andererseits gilt es geeignete Komprimierungsverfahren auszuwählen, die trotz möglichst geringem Speicherplatzverbrauch eine zufriedenstellende Wiedergabequalität der Audiodateien liefern. (vergleiche Abschnitt 4.5.1) Abschließend werden in Abschnitt 4.5.4 derzeit gängige Speichermedien für Audiodateien behandelt.

### 4.5.1 Komprimierung von Audiodaten

Nicht nur für die Speicherung der Musikdaten, sondern vor allem auch für das Streamen der Musiktitel ist es notwendig, die Datenmengen möglichst gering zu halten. Allerdings soll die subjektive Qualität des wiedergegebenen Signals davon nicht zu sehr beeinträchtigt werden. Das Ziel ist also, „... eine rasche, effiziente Übertragung bei möglichst hoher Qualität des Audiosignals ...“. [zit. RAFFASEDER, 2002 (S. 128) [RAF02]]

Bei den Verfahren zur Audiodatenkompression wird zwischen verlustlosen und verlustbehafteten unterschieden. Für die verlustlose Komprimierung werden Algorithmen eingesetzt, wie sie in ähnlicher Form auch bei anderen digitalen Daten Anwendung finden. Die Daten werden hier für die Übertragung so codiert, dass sie anschließend vollständig wiederhergestellt werden können. Die erzielte Reduktion der Datenmengen ist jedoch kaum ausreichend. [vgl. RAFFASEDER, 2002 (S. 129f) [RAF02]]

Verlustbehaftete Audiokompression versucht hingegen, irrelevante Daten, also für das menschliche Gehör nicht wahrnehmbare Signalanteile, herauszufiltern. Bei der sogenannten *predictiven Kodierung* wird die Vorhersagbarkeit des Signalverlaufs zur Datenreduktion eingesetzt. „*Perceptual Coding* nutzt die Erkenntnisse der Psychoakustik, um die Übertragungsrate zu reduzieren.“ [zit. RAFFASEDER, 2002 (S. 130) [RAF02]]

Zum Erstellen von verlustbehafteten Audiodaten werden eine Vielzahl von Codecs angeboten (*MPEG*, *MP3*, *AC-3*, *ATRAC*, *TwinVQ*, *Ogg Vorbis*, *aacPlus*, *mp3pro* und die *AAC-Kompression*), wobei die Ergebnisse der einzelnen Codecs oft sehr in ihrer Qualität variieren. Im Folgenden werden zwei davon näher behandelt:

- \* *Ogg Vorbis* ist ein patentfreier Encoder, der die Anforderungen des OpenSource Standards erfüllt - hiermit entspricht er der Thematik der vorliegenden Arbeit
- \* *MP3* ein patentgeschützter Codec, der mit einer Vielzahl von Lizenzen versehen ist. Da dieses Format jedoch am weitesten verbreitet ist, kann es in diesem Zusammenhang nicht ausgelassen werden, auch wenn es nicht dem OpenSource Schwerpunkt der vorliegenden Arbeit entspricht.

#### Ogg-Vorbis

Einer der Entwickler von *Ogg Vorbis* schreibt in einem Artikel der Zeitschrift *LINUX JOURNAL*: „*Ogg Vorbis* is patent free and it was designed that way from the beginning. There are no licensing fees or costs associated with using the format for any purpose whether it is commercial or noncommercial. It's also open source under terms of the LGPL, so even the source code is free for companies and fellow hackers.“ [zit. MOFFITT, 2001 [MOF01]]

Der Name *Ogg* stammt von einem allgemeinen Container-Format, das verschiedene Multimedia Dateien enthalten kann. *Vorbis* steht hingegen für den eingesetzten Co-

dec, dessen Dateien im Format *.ogg* abgespeichert werden. Laut Projektwebseite wurde er entwickelt, um alle proprietären, patentierten Audioformate vollständig ersetzen zu können. Des Weiteren wird der Codec wie folgt beschrieben: „Ogg Vorbis is [...] roughly comparable to other formats used to store and play digital music, such as MP3, VQF, AAC, and other digital audio formats. It is different from these other formats because it is completely free, open, and unpatented.“ [zit. VORBIS, 28.07.2007 (Webseite) [XIP07]]

Es bietet ein komprimiertes Audio Format in mittlerer bis hoher Qualität (8kHz-48kHz), mit einer variablen Bitrate im Bereich von 16 bis 128 kbps/Kanal. Darüber hinaus kann eine *.ogg* Datei bis zu 255 unabhängige Audio-Kanäle enthalten. Für das Speichern von Informationen zum Musiktrack<sup>31</sup> stellt Ogg-Vorbis „... einen flexiblen Tagging-Standard zur Verfügung, der die vollständige Anpassung der Tags an eine bestimmte Datei erlaubt - einschließlich benutzerdefinierter Tags ...“. [zit. WAGNER, 28.07.2007 (Webseite) [WAG07]]

Als weiterer Vorteil von *Ogg Vorbis* wird vielfach auch der Qualitätsunterschied zu MP3 codierten Musikstücken angeführt. Auf der Projektwebseite von XIPH werden unter dem Schlagwort „*Dare to compare!*“ Hörbeispiele zum Vergleich angeboten.<sup>32</sup> Grundsätzlich kann davon ausgegangen werden, dass *Ogg Vorbis* bei vergleichbaren Datenmengen im MP3-Format bessere Qualität liefert.

Die Qualitätseinstellung wird über sogenannte *quality levels* in einem Bereich von *-1 bis 10* getätigt. Für ein Stereosignal ergeben sich hier mögliche Bitraten von *50 kbps im Level -1* bis zu *rund 460 kbps im Level 10*. Innerhalb dieses Bereiches kann stufenlos eingestellt werden.

### MP3

Eines der bekanntesten datenreduzierten Audioformate ist das MP3-Format, welches vom deutschen *Fraunhofer-Institut für integrierte Schaltungen* entwickelt wurde und im *MPEG1 Layer 3* codiert wird. Auf der Projektwebseite wird es beschrieben als „... wahrnehmungsangepasstes Audiocodierverfahren [...] bei [dem] die decodierten Daten keine exakte Kopie der Originaldaten sind. Stattdessen werden die digitalen Audiodaten so bearbeitet, dass sie in der Wiedergabe trotz hoher Kompressionsfaktoren exakt - oder zumindest so genau wie möglich - klingen wie das Original. Dies erreicht man, indem die Codierung an die Wahrnehmungseigenschaften des menschlichen Ohres angepasst wird: Die Teile des Audiosignals, die man als Mensch sehr gut wahrnehmen kann, werden mit hoher Genauigkeit codiert; weniger gut wahrnehmbare Signale werden weniger genau codiert. Signalbestandteile, die wir überhaupt nicht wahrnehmen können, werden entfernt oder durch Quantisierungsrauschen ersetzt.“ [zit. FRAUNHOFER INSTITUT, 19.07.2007 (Webseite) [FRA07]]

Die technisch mögliche Bitrate der Kodierung liegt im Bereich von 8 kBit/s bis 320 kBit/s, allerdings wird von den EntwicklerInnen für Mono Signale eine Datenrate ab 80 kBit/s und für Stereosignale eine Rate ab 160 kBit/s empfohlen.

In folgendem Rechenbeispiel wird das Komprimierungsverhältnis zwischen einer

---

<sup>31</sup> genaue Informationen zu den Ogg-Tags sind unter <http://reactor-core.org/ogg-tagging.html> zu finden

<sup>32</sup> *Dare to compare* - der Link zur Webseite: <http://www.xiph.org/vorbis/listen.html>

Datei im Audiorohformat *.wav* und einer Datei im *.mp3* Format ermittelt. Als Grundlage für die Berechnung der Datenrate (*DR*) dient eine Standard *.wav* Datei im Stereoformat (Anzahl der Tonspuren  $A = 2$ ) mit einer 16 Bit Quantisierung ( $n$ ) und einer Samplingfrequenz von 44,1 kHz ( $f_s$ ). Um zur Einheit der MP3-Datenraten konform zu sein, wird das Ergebnis in kbps umgerechnet (Division durch 1024). Dies ergibt unter der Verwendung von Gleichung 4.1 einen Wert von 1378, 123 kbps. [vgl. RAFFASER, 2002 (S. 125) [RAF02]]

Mit einer Datenrate der *.mp3* Datei von 160 kbps ergibt sich ein Kompressionsfaktor von 1 : 8,6.

$$DR_{kbps} = \frac{A \cdot n \cdot f_s}{1024} \quad (4.1)$$

In der praktischen Anwendung gilt es aus der verfügbaren Bandbreite des Netzwerks und den klanglichen Qualitätsansprüchen das richtige Maß für die Kodierung zu finden.

**ID3** Eine weitere Besonderheit des MP3-Formats ist das sogenannte *ID3-Tag*, welches Informationen über den Musiktitel enthält. In der ursprünglichen Version eins (*ID3v1*) wird das 128 Byte lange Tag am Ende der MP3 Datei gespeichert. In Abbildung 4.5 und Tabelle 4.1 wird der Aufbau von einem ID3-Tag der Version 1.1 dargestellt. [vgl. ID3, 19.07.2007 (Webseite) [ID307b]]

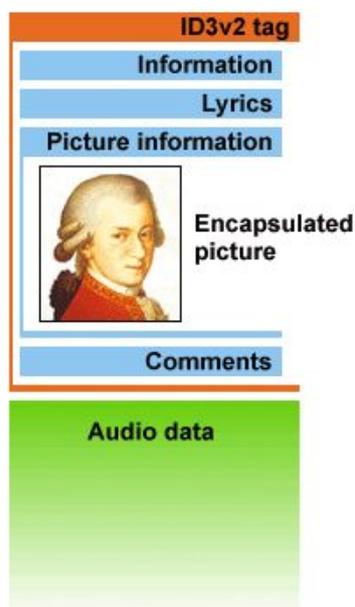


Abbildung 4.5: Blockdiagramm ID3-Tag Version 1

Byte	Beschreibung
3	TAG-Identification
30	Song Titel
30	Artist
30	Album
4	Year
28	Comment
2	Track Number
1	Genre

**Tabelle 4.1:** Aufbau eines MP3-Tags der Version 1.1

Da die technischen Gegebenheiten von *ID3v1* mittlerweile nicht mehr ausreichen, wird mit *ID3v2* eine vollständig überarbeitete Version veröffentlicht. Das Tag befindet sich, im Gegensatz zur *ID3v1*, am Beginn der jeweiligen MP3-Datei und es können darin bis zu 256 MB gespeichert werden. Weitere Erneuerungen sind die Möglichkeit, den Liedtext des Songs und Bilder (also zum Beispiel CD-Cover) oder andere Dateien zum Track zu speichern. Abbildung 4.6 zeigt die Zusammensetzung eines *ID3v2*-Tags. Eine sehr genaue Beschreibung von *ID3v2* inklusive einer Auflistung der Identifizierungsnummern für 80 verschiedene Genres ist auf der Projektwebseite von *ID3* zu finden. [vgl. ID3, 14.08.2007 (Webseite) [ID307c]]



**Abbildung 4.6:** Blockdiagramm ID3-Tag Version 2

Die im *AMSL* verwendeten Programme unterstützen beide Versionen des ID3-Tags.

### 4.5.2 Ordnerstruktur

Die Übersichtlichkeit der Musiksammlung hängt sehr von der Wahl der Ordnerstruktur ab. Dafür gibt es zwar keine definierten Standards, jedoch hat sich folgende Einteilung (siehe Abbildung 4.7), die auch bei vielen Musikplayern verwendet wird, als bewährt erwiesen:

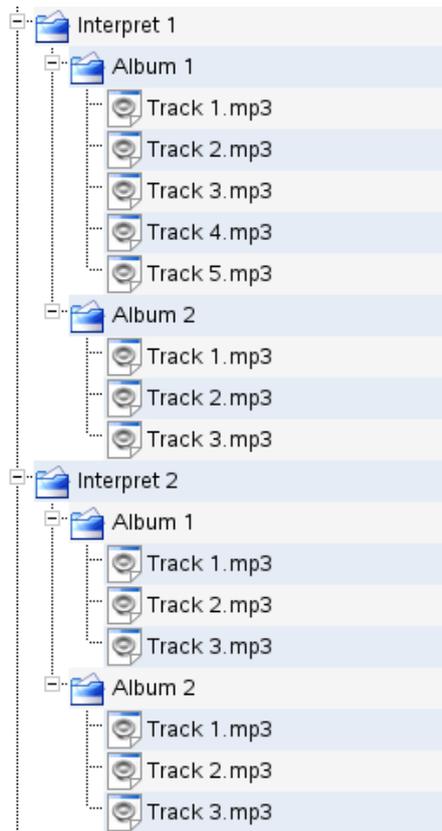


Abbildung 4.7: Ordnerstruktur in einem Musikverzeichnis

Diese Verzeichnisstruktur wird auch in den Programmen *Ampache* und *Subsonic* für die Anzeige der vorhandenen Titel übernommen. Durch eine konsequente Sortierung der Musiktitel können zudem doppelte Einträge vermieden und somit Speicherplatz gespart werden. Außerdem kann damit die Suche nach bestimmten Titeln optimiert werden.

### 4.5.3 Dateinamenvergabe

Für die Vergabe von Dateinamen scheint es ebenfalls keine Konventionen zu geben. Jedoch kann auch hier, durch eine sinnvolle Wahl der Tracktitel, Struktur in die Musikverzeichnisse gebracht werden. Grundsätzlich sollte bei der Bezeichnung darauf geachtet werden, den Dateinamen möglichst kurz zu halten. Dennoch sollten die wichtigsten Informationen enthalten sein. Leerzeichen sollten als Trennzeichen möglichst

vermieden werden, um Interpretationsfehler zu umgehen. Statt dessen ist es sinnvoller „-“ oder „\_“ zu verwenden.

Bei Programmen zum Rippen von Audiodateien (siehe dazu auch Absatz *Audio-CD* auf Seite 43) werden die Titel automatisch von vordefinierten Templates generiert, die nach den eigenen Ansprüchen angepasst werden können. Der Name setzt sich hier häufig aus Interpret, Album, Track-Nummer und Titel des Musikstücks zusammen, wobei die Reihenfolge dabei meist verändert werden kann. Erfahrungsgemäß hat es sich bewährt, bei in Verzeichnisstrukturen organisierten Sammlungen, wie es zum Beispiel im Abschnitt 4.5.2 aufgezeigt ist, die Tracknummer und den Titel des Musikstücks als erstes anzugeben und danach den Interpreten. Dadurch ergibt sich eine bessere Lesbarkeit, vor allem wenn nur Teile des Dateinamen angezeigt werden.

```
00-tracktitle-artist.ogg
```

Bei unsortierten Sammlungen kann es hingegen von Vorteil sein, im Dateinamen zuerst den Interpreten anzuführen, anschließend Tracknummer und Tracktitel. So bleiben die Dateien bei alphabetischer Auflistung übersichtlich nach dem Interpreten sortiert.

```
artist-00-tracktitle.ogg
```

Informationen über eine CD (wie zum Beispiel Interpret, Album- und Tracktitel) können beim Rippen, sofern diese dort erfasst ist, aus einer online Musikdatenbank (CDDDB) ausgelesen werden. Dieses Service wird zum Beispiel von FreeDB.org angeboten, welches vor allem bei OpenSource Programmen häufig zum Einsatz kommt und daher oft als Standardeinstellung gesetzt ist. Andernfalls kann FreeDB mit der Adresse *freedb.freedb.org* als Quelle eingetragen werden. Die erhaltenen Informationen werden im ID3-Tag zu jedem Titel gespeichert. Aus MP3s, deren ID3-Tags bereits eingetragen sind, können die Dateinamen mit den entsprechenden Informationen auch automatisch generiert werden. Dazu kann zum Beispiel das Programm *ID3-TagIT* verwendet werden.<sup>33</sup>

### Sonderzeichen

Bei Dateinamen können durch „... Umlaute und Sonderzeichen (+ ~ % \$ ! \ & , : # [ ] { } ( ) < > ) sowie Zeichen aus nordischen und anderen Sprachen ...“ Probleme in der Kodierung auftreten. [zit. UNI KÖLN, 03.08.2007 (Webseite) [UNI07]]

Je nach verwendetem Zeichensatz ergeben sich, nicht zuletzt auf Grund der verwendeten Anzahl der Bytes pro Sonderzeichen, unterschiedliche Kodierungen. Für die ISO 8859-xx Kodierung<sup>34</sup> ist beispielsweise nur ein Byte (entspricht 8 Bit) pro Zeichen vorgesehen, wobei xx eine Bezeichnung für den jeweiligen Ländercode ist. UTF-8 hingegen ist ein internationaler Standard, bei welchem „... die lateinischen Standardbuchstaben und Ziffern mit einem Byte codiert [werden], Sonderzeichen und Umlaute mit zwei oder drei Byte.“ [zit. ITWISSEN, 03.08.2007 (Webseite) [ITW07c]]

---

<sup>33</sup><http://www.id3-tagit.de/german/index.htm>

<sup>34</sup>siehe dazu auch [http://www.itwissen.info/definition/lexikon/\\_\\_\\_iso%208859.html](http://www.itwissen.info/definition/lexikon/___iso%208859.html)

Die sicherste Methode um Probleme in der Kodierung zu umgehen ist wohl, die entsprechenden Zeichen nicht zu verwenden, vor allem wenn ein Ändern des Zeichensatzes nicht möglich sein sollte.

Im AMSL treten bei *Subsonic* vor allem bei skandinavischen Schriftzeichen Kodierungsfehler auf. Das Problem liegt offensichtlich an der Kodierung der *ID3v1*-Tags, wenn die Tracks mit *ID3v2*-Tags bezeichnet werden, können sie in *Subsonic* fehlerfrei dargestellt werden. Bei der Verwendung eines *Lame-Encoders* zum Erzeugen der MP3 Daten muss für die Verwendung der *ID3v2*-Tags die Option `-id3v2-only` hinzugefügt werden. (vergleiche dazu auch den Absatz *ID3* auf Seite 39)

### 4.5.4 Speichermedien

Als Speichermedium für digitale Musikdaten stehen einige sehr unterschiedliche Systeme zur Verfügung. Diese reichen von der herkömmlichen Audio-CD über Daten CDs oder DVDs bis zu Festplatten und Flashspeicher. Die einzelnen Systeme werden im Folgenden näher beschrieben.

#### Audio-CD

Das von der Musikindustrie wohl am häufigsten verwendete Speichermedium ist die *Compact Disc* (CD). Ursprünglich war diese als Nachfolger der Schallplatte gedacht, mittlerweile finden sich jedoch auch außerhalb der Musikbranche zahlreiche Anwendungsgebiete.

„Für eine Schallaufzeichnung in CD-Qualität [...] werden pro Minute rund 10 Megabyte an Speicherplatz benötigt.“ [zit. RAFFASEDER, 2002 (S. 125) [RAF02]]  
Daraus ergibt sich bei der für die CD standardisierte Quantisierungsrate von 16 Bit und der Samplingfrequenz von 44,1 kHz eine Kapazität von bis zu 74 Minuten Audiomaterial im Stereoformat. Eine sehr ausführliche Beschreibung über die Funktionsweise der CD ist in dem „*Handbuch der Tonstudioteknik*“ von DICKREITER zu finden. [vgl. DICKREITER, 1997 (S. 334ff) [DIC97]]

Die Daten der Audio-CD können nicht direkt auf den Computer kopiert werden, wie es bei Daten CDs der Fall ist. Dazu müssen die Audiodaten erst mit einem entsprechenden Programm *gerippt* werden. „Unter Rippen versteht man das digitale Extrahieren von Audiodaten, also ein Prozess, bei dem ein digitaler Audio-Track [...] von einer bereits existierenden Audio-CD in eine MP3-Datei [oder ein anderes Audio Format] auf der Festplatte [des] Rechners umgewandelt wird.“ [zit. BETZ, 2004 (S. 73) [BET04]] Einige Programme rippen die Dateien im Rohformat *.wav*, von dem aus dann in verschiedene verlustbehaftete Audioformate (beispielsweise *.ogg* oder *.mp3*) transcodiert werden kann. Beispielprogramme für Audioripper sind *kaudiocreator*<sup>35</sup> - ein OpenSource Programm für Linux, und *Audiograbber*<sup>36</sup> - ist zwar kein OpenSource Programm, jedoch geeignete *Freeware* für Windows. Um eine konstante Wiedergabelautstärke beim Abspielen der Playlist zu ermöglichen, empfiehlt es sich, die MP3s beim Erzeugen zu normalisieren.

---

<sup>35</sup>*kaudiocreator*: zu finden unter <http://www.icefox.net/programs/?program=KAudioCreator>

<sup>36</sup>*Audiograbber* zu finden unter <http://www.audiograbber.de/>

### Daten CD/DVD

Neben der Verwendung in der Musikbranche ist die CD auch in der Computerwelt weit verbreitet. Durch die Möglichkeit mit einem CD-Brenner Daten vom eigenen Computer auf eine CD zu sichern, hat sich die Technologie zu einem beliebten Speichermedium entwickelt. Auch das Kopieren von Original-CDs ist dadurch möglich geworden, was zu großen Urheberrechtsdiskussionen in der Musikindustrie geführt hat.

Für den AMSL ist in diesem Zusammenhang das Speichern von komprimierten Musikdaten auf einer CD von Bedeutung. So sind vor allem MP3-CDs weit verbreitet, die sogar auf manchen CD-Playern abgespielt werden können (CD-Player mit MP3 Funktion). Im Vergleich zur Audio CD, wo auf den verfügbaren 700 MB ein Album mit ungefähr 15 - 20 Titeln gespeichert ist, können, abhängig von der Komprimierungsrate der Daten, ungefähr zehn Mal so viele Titel gespeichert werden. Für die Berechnung wird von einer Audio-CD mit der Gesamtlänge von 74 Minuten, bei einer Anzahl von 20 Tracks à drei Minuten, ausgegangen. Daraus ergibt sich pro Musiktitel ein Speicherplatz von rund 30 MB.

Bezugnehmend auf die Berechnungen im Abschnitt 4.5.1 (Gleichung 4.1) kann die mögliche Anzahl an MP3s auf einer Daten-CD abgeschätzt werden. Durch die Komprimierung (160 kbps im MP3 Format) kann der benötigte Speicherplatz pro Track um den Faktor 8,6 verringert werden. Hiermit kann für ein Album von einem Speicherplatzverbrauch von rund 80 MB ausgegangen werden. Somit können rund 170 Titel von je drei Minuten Länge auf einer Daten-CD abgespeichert werden.

Als Weiterentwicklung der CD hat sich die *Digital Versatile Disc (DVD)*<sup>37</sup> etabliert, die in der technischen Umsetzung sehr der CD ähnelt. Im Laufe der Zeit haben sich mehrere Typen und Formate der DVD entwickelt, von denen sich aber nur einige wenige durchgesetzt haben. Das am Häufigsten verwendete Format zur Datenspeicherung ist die DVD-R, die eine Speicherkapazität von 4,7 GB bietet. In Anlehnung an die Berechnungen für die Daten-CD können auf einer DVD rund 1150 Titel im MP3 Format gespeichert werden

### Festplatte

Festplatten sind derzeit wohl das wichtigste Massenspeichermedium im Computerbereich. Demnach befindet sich diese Technologie in ständiger Weiterentwicklung.<sup>38</sup> Abbildung 4.8 zeigt die Veränderung der Kapazität und des Preis-Leistungsverhältnisses (Angaben in US-Dollar) von Festplatten in zehn Jahres Schritten, beginnend mit dem Jahr 1956.

Mittlerweile werden Festplatten mit einer Kapazität von bis zu 1 Tera Byte (1 TB entspricht 1000 GB) angeboten, bei denen sich die Kosten pro GB auf ungefähr 0,40 US-Dollar<sup>39</sup> belaufen. Das beste Preis-Leistungsverhältnis erhält man in Österreich

---

<sup>37</sup>genauere Informationen über die DVD: [http://www.itwissen.info/definition/lexikon//\\_dvddvd\\_dvddigital%20versatile%20discdvd\\_dvd.html](http://www.itwissen.info/definition/lexikon//_dvddvd_dvddigital%20versatile%20discdvd_dvd.html)

<sup>38</sup>Details zur Funktionsweise der Festplatte siehe: [http://www.itwissen.info/definition/lexikon//\\_hdhd\\_hdhard%20diskhd\\_hdfestplatte.html](http://www.itwissen.info/definition/lexikon//_hdhd_hdhard%20diskhd_hdfestplatte.html)

<sup>39</sup>1 US Dollar (USD) = 0.73165 Euro (EUR)

Jahr	Festplatten-größe	Speicher-kapazität	Kosten pro GB
1956	24 Zoll	5 MB	1,4 Mio. Dollar
1976	14 Zoll	NA	NA
1996	5,25/3,5 Zoll	1,44 GB	179 Dollar
2006	3,5/2,5/1,8/ 1,0/0,85 Zoll	750 GB	ca. 0,50 Dollar

Abbildung 4.8: Entwicklung der Speicherkapazität von Festplatten

derzeit mit Festplatten in der Größe von 300 bis 400 GB - hier kommen die Kosten pro GB auf rund 0,25 USD. (Stand August 2007)

Festplatten sind entweder direkt in den Computer eingebaut, oder in einem externen Gehäuse, welches über ein USB-Kabel oder FireWire-Kabel mit dem Computer verbunden wird. In Speicherstationen, die meistens über das Netzwerk angesprochen werden (Network Attached Storage - NAS<sup>40</sup>), können auch mehrere Festplatten in einem RAID<sup>41</sup> System zu einem großen logischen Datenträger zusammengefasst sein.

Auch im AMSL ist die Festplatte das primäre Speichermedium, die Standardmusikverzeichnisse werden hier abgelegt.

## Flash-Speicher



Abbildung 4.9: Flash Speichermedien

Flash-Speicher (siehe Abbildung 4.9) sind handliche, mobile Speichermedien, die keine mechanischen Elemente für den Speichervorgang benötigen. Sie „...“ behalten

<sup>40</sup>Erklärung NAS: [http://www.itwissen.info/definition/lexikon//\\_nasnas\\_nasnetwork%20attached%20storage%20nas\\_nas.html](http://www.itwissen.info/definition/lexikon//_nasnas_nasnetwork%20attached%20storage%20nas_nas.html)

<sup>41</sup>Erklärung des RAID-Systems: [http://www.itwissen.info/definition/lexikon//\\_raidraid\\_raidredundant%20array%20of%20inexpensive%20disksraid\\_raidraid-system.html](http://www.itwissen.info/definition/lexikon//_raidraid_raidredundant%20array%20of%20inexpensive%20disksraid_raidraid-system.html)

nach dem Abschalten der Stromversorgung ihren Speicherzustand (= nichtflüchtig). Ihre Arbeitsweise ist mit den Arbeitsspeichern (RAM) eines PCs vergleichbar.“ [zit. ASBYON, 02.08.2007 (Webseite) [ASB07]]

Hauptsächlich werden sie bei portablen Geräten wie MP3-Playern, Digitalkameras, Camcordern oder PDAs eingesetzt, auch in mobilen Aufnahmegeräten finden sie neuerdings Verwendung. Außerdem können sie mit den entsprechenden Lesegeräten sehr einfach mit dem PC oder Laptop verbunden werden, um die gespeicherten Daten zu sichern oder zu kopieren.

Beispiele für Flash-Speicher sind USB-Stick, Compact-Flash-Karte (CF), Smart-Media-Karte, Memory-Stick-Karte, Multimedia-Karte (MMC) und die SD-Karte. Mittlerweile sind Speicherkapazitäten von mehreren hundert MB bis hin zu einigen GB erhältlich.<sup>42</sup>

Nicht zuletzt auf Grund der Anwendung als MP3-Player sind Flash-Speicher ein relevantes Medium für den Medienserver. Musikverzeichnisse von derartigen Speichern können mit dem entsprechenden Pfad direkt in die Programme *Ampache* und *Subsonic* eingebunden werden.

---

<sup>42</sup>Ausführliche Erklärungen zu den einzelnen Speichermedien sind auf der Webseite <http://www.itwissen.info/> unter den Schlagworten USB-Stick, CF Karte, Speicherkarte, SD Karte zu finden

### KuBox - ein virtueller Medienserver

---

Dieses Kapitel beinhaltet eine Beschreibung des Medienservers von der Installation der Komponenten bis zum Betrieb der Serverdienste. Als Grundlage für den Medienserver wird die *KuBox*<sup>1</sup> verwendet, die sich nach ausführlichen Tests und Vergleichen mit den verfügbaren Technologien als das optimale vorkonfigurierte System herausgestellt hat. Der Medienserver kann je nach verfügbarer Hardware auch auf anderen Systemen (siehe Kapitel 4.2) betrieben werden. Hier wird zur Demonstration der praktischen Umsetzung die Installation auf einer virtuellen Maschine genauer betrachtet.



Abbildung 5.1: Loginfenster der KuBox

### 5.1 Getting Started - Anleitung zur Inbetriebnahme

Der folgende Abschnitt befasst sich mit der genauen Zusammensetzung der *KuBox* - zu Beginn sind einige Anforderungen an das System, auf dem die *KuBox* installiert

---

<sup>1</sup>vergleiche dazu auch Kapitel 4.2.3

werden soll, angeführt. Danach sind die verwendeten Software Pakete in einer Tabelle aufgelistet. Abschließend werden die in der *KuBox* verfügbaren Dienste vorgestellt.

### 5.1.1 Voraussetzungen für den Betrieb eines Medienservers

Für die Inbetriebnahme der *KuBox* müssen einige Hard- und Softwarevoraussetzungen erfüllt sein, diese werden im Folgenden kurz erläutert.

**Netzwerkverbindung** Um einen Medienserver im Heimnetzwerk betreiben zu können, muss ausreichend Bandbreite zur Verfügung stehen - also ein 100 MBit Ethernet oder WLAN (54 MBit). (vergleiche dazu auch Kapitel 4.4.1 und 4.4.2)

**Audiowiedergabe** Für die Audiowiedergabe über den Computer wird eine Soundkarte<sup>2</sup> benötigt, welche auch die Anschlüsse für eine eventuelle Verbindung zur Stereoanlage zur Verfügung stellt. Heutzutage sind in den meisten Computern bereits standardmäßig Soundkarten enthalten.

**Betriebssystem** Je nach verwendetem Serversystem werden unterschiedliche Ansprüche an das Betriebssystem gestellt. Für eine Installation direkt am PC wird eine Linux Distribution vorausgesetzt, da hier die benötigten Serverdienste zur Verfügung stehen. Außerdem entsprechen diese auch den OpenSource Anforderungen des Projekts. Für den Client kann grundsätzlich jedes beliebige Betriebssystem verwendet werden. (siehe dazu auch Kapitel 4.3.1)

**Geeignete Hardware** Die verwendeten Rechner müssen für das Betreiben des AMSL ausreichend Arbeitsspeicher (RAM) zur Verfügung haben, sinnvoll sind aus Performance Gründen mindestens 512 MB RAM. Bei Verwendung der *VirtualBox* wird ein Arbeitsspeicher von mindestens 1 GB RAM empfohlen - bei Computern mit weniger RAM muss mit Performance Einbußen bei anderen Prozessen am PC und mit erhöhten Ladezeiten im Webinterface gerechnet werden.

Eine weitere wichtige Komponente des Systems ist das CD/DVD Laufwerk, es kann jedoch davon ausgegangen werden, dass es standardmäßig im Computer eingebaut ist.

**Speicherplatz** Für digitalisierte Musiksammlungen sollte von vorne herein ausreichend genug Speicherplatz berechnet werden, damit die Daten nicht immer wieder umkopiert werden müssen. Natürlich können im Nachhinein weitere externe Festplatten hinzugefügt werden.

### 5.1.2 Liste der zu installierenden Programme

In Tabelle 5.1 sind alle Programmpakete aufgelistet, die am Medienserver installiert sind. Die Programme können alle entweder direkt von der Installations-DVD oder der

---

<sup>2</sup>siehe dazu auch: [http://www.itwissen.info/definition/lexikon/\\_\\_\\_sound%20card\\_soundkarte.html](http://www.itwissen.info/definition/lexikon/___sound%20card_soundkarte.html) - je nach Qualitätsanspruch sollte eine entsprechende Soundkarte verwendet werden.

entsprechenden online Paketquelle bezogen werden. In den neueren Versionen von *Ubuntu* (ab Feisty Fawn 7.04) sind die benötigten Paketquellen nach der Installation bereits aktiviert.<sup>3</sup>

Spalte eins der Tabelle listet die Paketnamen der jeweiligen Programme auf, wie sie in den Paketquellen zu finden sind. Die Funktionen der einzelnen Programme sind in Spalte zwei angeführt, Spalte drei zeigt, für welche Programme diese als Voraussetzung gelten. Spalte vier (mit der Bezeichnung S/M) gibt an, ob das Paket standardmäßig installiert wird (S) oder manuell nachinstalliert werden muss (M).

---

<sup>3</sup>im deutschen Ubuntu Forum werden die Grundlagen zur Installation in Ubuntu beschrieben <http://wiki.ubuntu-forum.de/index.php/Paketverwaltung>

<b>Paketname</b>	<b>Funktion</b>	<b>Verwendet in</b>	<b>S/M</b>
amarok	Audio Player		S
amarok-xine	xine engine	Amarok	S
ampache	Webapplikation		M
apache2	Webserver	Ampache	M
firefox	Webbrowser	Ampache, Subsonic	S
k3b	CD-Brenner		S
kaudiocreator	CD-Ripper		M
lame	MP3 encoder	kaudiocreator	M
libfcgi-dev	Plugin	lighttpd	M
libmad0	Plugin	Amarok, mp3	M
libxine-extracodecs	Plugin	Amarok, mp3	M
lighttpd	Webserver	Ampache	M
MPD	Audio Player	„local play“ Funktion	M
mysql-server	MySQL Datenbank Server	Ampache	M
mysql-client	MySQL Datenbank Client	Ampache	M
php5	Web Script Sprache	Ampache	M
php5-cgi	Plugin	PHP5, lighttpd	M
php5-mysql	MySQL Modul für PHP	Ampache	M
phpmyadmin	MySQL Administration	MySQL	M
samba	Fileserver	Stand PC	M
subsonic	Webapplikation		M
sun-java5-bin	Java Runtime Environment	Subsonic, Tomcat	M
sun-java5-jre	Java Runtime Environment	Subsonic, Tomcat	M
sun-java5-plugin	Java Plugin	Subsonic, Tomcat	M
tomcat6	Java Servlet	Subsonic	M
XSPF-Player	Web Audio Player	Ampache	M

**Tabelle 5.1:** Programmpakete im Medienserver

### 5.1.3 Verfügbare Dienste des Servers

**Archivierung** von Audio Dateien (Geräusche, Lieder, Hörbücher, ...) in allen gängigen Audioformaten. Je nach Speicherplatz und Anwendung können diese unkomprimiert (beispielsweise AIFF, WAV, ...) oder komprimiert (wie zum Beispiel MP3, Ogg Vorbis, WMA, ...) auf einer Computerfestplatte oder einem externen Storage Server gespeichert werden. Diese Daten stehen dann zentral im Netzwerk allen Clients zur Verfügung.

**Importieren** Effizientes Hinzufügen von Audio Dateien aus beliebigen Speichermedien in die Musiksammlung. Diese können entweder in das Musikverzeichnis am Server kopiert werden oder von einem externen Speichermedium in das Webinterface eingebunden werden. Hierfür braucht nur der entsprechende Pfad eingetragen werden. Nach einem Reload des Musikverzeichnisses im Webinterface werden die aktuellen Daten angezeigt.

**Audio-CDs einlesen** Über das CD-Laufwerk werden mittels eines CD-Rippers die Audiodaten eingelesen und in ein kompatibles Audio-Format transcodiert (vergleiche dazu auch Kapitel 4.5.4). Die Daten können dann automatisch im Musikverzeichnis je nach Anforderung im entsprechenden Dateiformat abgespeichert werden. Dabei werden die einheitliche Ordnerstruktur und die Dateinamenvergabe aus den Einstellungen des CD-Ripping Programms übernommen - gegebenenfalls müssen die standard Einstellungen auf die gewünschte Form geändert werden.<sup>4</sup>

**CDDB-Lookup** Automatisches Laden von Informationen wie Interpreten, Albumtitel oder Tracktitel zu den importierten CDs aus der *Compact Disk Database* (CDDB). Dieses Feature wird vom CD-Ripper *kaudiocreator* angeboten.

**Webinterface** *Ampache* und *Subsonic* können gleichwertig über den Webbrowser verwendet werden. Auf Grund der vergleichbaren Funktionalität sind beide Programme auf der *KuBox* installiert. Die Wahl des persönlichen Favoriten bleibt dem/der BenutzerIn überlassen. Das Webinterface dient vor allem zur Suche und Bearbeitung von Einträgen, dem Erstellen von Playlists und dem Hören und Download von Dateien.

**Informationen zur Datei speichern** ist einerseits möglich über ID3-Tags bei MP3 Daten (siehe dazu auch Kapitel 4.5.1), andererseits können auch über das Webinterface von *Ampache* oder *Subsonic* Kommentare zu Musiktiteln abgespeichert werden, welche dann allerdings nur im jeweiligen Programm verfügbar sind.

**Netzwerkplayer** Sämtliche zentral gespeicherten Musiktitel sind für die angeschlossenen Clients über das Netzwerk verfügbar. Die als Stream empfangene Musik kann am Clientrechner mit gewöhnlichen Softwareplayern (*Amarok*, *Songbird*, *Winamp*, *Windows Media Player*, *iTunes*, ...) abgespielt werden.

---

<sup>4</sup>Da in der *KuBox* keine Audio-CDs erkannt werden, muss dies über den host Rechner vorgenommen werden, es stellt jedoch keine erhebliche Einschränkung dar.

**Playlist** Über die Musikverwaltungssoftware (*Ampache* und *Subsonic*) können die BenutzerInnen individuelle Playlists anlegen, welche dann zum Abspielen an den lokalen Musikplayer im Format *.m3u* gesendet werden. Im Webinterface können auch Playlists anderer UserInnen zum Abspielen ausgewählt werden. Mit den entsprechenden Einstellungen kann die Playlist auch direkt über das Webinterface abgespielt werden. (vorausgesetzt es ist ein lokaler Player aktiviert)

**Datenbank** läuft im Hintergrund der jeweiligen Programme, die Titel werden beim Laden des Verzeichnisses automatisch in die Datenbank eingetragen. Während die Daten von *Subsonic* nur schwer zugänglich sind, weil die Datenstruktur direkt in Java ausprogrammiert ist, können Daten von *Ampache* eventuell auch in anderen Anwendungen verwendet werden, da es sich um eine MySQL-Datenbank handelt.

**Samba Server** dient zum Datenaustausch zwischen Server und Client Rechner über das Netzwerk. Über Samba freigegebene Verzeichnisse erscheinen bei Windows-Clients in der Netzwerkkumgebung und können auch bei Linux-Clients als Laufwerk eingebunden werden. (siehe dazu auch Kapitel 5.1.3)

**Daten exportieren** Um Musikdaten zu exportieren, können entweder einzelne Tracks über das Webinterface von *Ampache* oder *Subsonic* auf den Client Rechner heruntergeladen oder mehrere Dateien gleichzeitig über den Samba-Server kopiert werden. Aus diesen Daten kann am Client Rechner dann beispielsweise eine Audio-CD erstellt werden (siehe dazu auch Kapitel 4.3.6), oder sie können auf einen portablen MP3 Player, wie zum Beispiel iPod, übertragen werden. Das Kopieren von Musik auf den portablen MP3 Player kann auch direkt am Server Rechner durchgeführt werden, sofern diese mittels USB Schnittstelle miteinander verbunden werden können. Der am Server standardmäßig installierte Musik Player *Amarok* bietet hierfür ein eigenes Plugin zum Verwalten von Playlists am iPod. Eine Anleitung dafür ist auf der Webseite von *howtoforge* zu finden.<sup>5</sup>

## 5.2 Beschreibung eines praktischen Beispiels - KuBox

### 5.2.1 VirtualBox

Grundlage für die *KuBox* ist, wie schon in Abschnitt 4.2.3 erwähnt, *VirtualBox* - eine OpenSource Virtualisierungs-Software. Hierfür wird ein virtuelles Image der Linux Distribution *Kubuntu* in der Version 7.04 erstellt (*KuBox* - das Image ist auf der beigelegten DVD enthalten). Dieses Image kann an jedem Host Rechner, der *VirtualBox* installiert hat, verwendet werden.

Hierfür muss es in der grafischen Oberfläche der *VirtualBox* über Datei > Manager für virtuelle Laufwerke geladen werden und erscheint dann in der Liste der verfügbaren virtuellen Maschinen. Natürlich können auch neue virtuelle Maschinen angelegt werden, auf denen jedes beliebige Betriebssystem vom entsprechenden Installationsmedium installiert werden kann.

<sup>5</sup>iPod Playlists in Amarok: [http://howtoforge.org/linux\\_amarok\\_ipod](http://howtoforge.org/linux_amarok_ipod)

Das ausgewählte Image kann dann über das Icon `Starten` gestartet werden. Daraufhin bootet die *KuBox* als virtuelles Betriebssystem und ist von dort an im Betrieb vergleichbar mit einem System auf einem herkömmlichen Computer.

In der *VirtualBox* laufen standardmäßig ein virtueller Router und DHCP-Server, welche die Verbindung in das Internet ermöglichen. Die Netzwerkkonfigurationen werden vom System bei der Installation automatisch auf das Subnetz *10.0.2.15* gesetzt.

Durch die zusätzliche Installation der *Linux Guest Additions* kann die Performance des Systems verbessert werden und einige weitere Features, wie zum Beispiel *Mouse pointer integration*, *Shared Folders* oder *Shared Clipboard* werden aktiviert. Nähere Informationen darüber können im Users Manual der *VirtualBox* nachgelesen werden. [vgl. INNOTEK, 2007 (S. 44f) [INN07]]

Physikalische Laufwerke des host OS, wie zum Beispiel das CD/DVD Laufwerk oder USB Schnittstellen können virtualisiert und im guest OS eingebunden werden. Außerdem können Ordner am host OS freigegeben werden, auf welche vom guest OS zugegriffen werden kann. Die Audioschnittstelle der *VirtualBox* ist standardmäßig deaktiviert, für den Betrieb der *KuBox* muss diese Einstellung geändert werden.

Die einzige Einschränkung durch den Betrieb des AMSL als virtuelles System ist, dass über das CD-Laufwerk keine Audio-CDs direkt in die *KuBox* eingebunden werden können. Das virtualisierte Laufwerk akzeptiert nur Daten CDs/DVDs. Als Workaround dafür muss die CD am host OS gerippt und die Daten dann in ein freigegebenes Verzeichnis kopiert werden (vergleiche *sharedMusic* im Abschnitt 5.2.2) - dadurch ist die Musik auch am host OS verfügbar.

Eine Anleitung für die Installation von *VirtualBox* auf Ubuntu ist auf der Webseite von *howtoforge*<sup>6</sup> zu finden. Außerdem stellt Innotek ein sehr ausführliches Benutzerhandbuch zur Verfügung, das sehr hilfreiche Informationen zur Installation und zum Betrieb von *VirtualBox* bietet.<sup>7</sup>

### 5.2.2 Ubuntu auf der Virtuellen Box - Bereits Vorkonfiguriert...

Dieser Abschnitt 5.2.2 gibt einen Überblick über die Konfiguration des AMSL. Die beschriebenen Einstellungen sind am System bereits durchgeführt und müssen von dem/der AnwenderIn nicht verändert werden. In Abschnitt 5.2.3 werden dann darauf folgend jene Arbeitsschritte genauer angeführt, die von dem/der BenutzerIn noch durchzuführen sind, um die *KuBox* verwenden zu können.<sup>8</sup>

#### Grundkonfiguration

Nach dem ersten Start von *Kubuntu* müssen einige Einstellungen getätigt werden, um daraus die *KuBox* zu erhalten.

Bei der Installation wurde ein Benutzerkonto unter dem Namen `AMSLadmin` mit

---

<sup>6</sup>[http://howtoforge.org/virtualbox\\_ubuntu](http://howtoforge.org/virtualbox_ubuntu)

<sup>7</sup>Benutzerhandbuch zur Innotek VirtualBox: <http://www.virtualbox.org/download/UserManual.pdf>

<sup>8</sup>Eine Auflistung sämtlicher Passwörter ist im Anhang A zu finden.

dem Passwort `Ams1PW` angelegt. Als *root* Passwort wurde `DAMedia` angegeben, welches auch bei allen anderen Anwendungen in der *KuBox* als Passwort verwendet wird.

Über die Paketverwaltung *Adept* wird für die *KuBox* zusätzlich benötigte Software nachinstalliert. (siehe dazu auch Abschnitt 5.1.2) Programme aus der Standardinstallation, die im AMSL keine Verwendung finden, werden aus Kapazitätsgründen über die Paketverwaltung entfernt.

### Verzeichnisse Einbinden

Für das Einbinden von Musikverzeichnissen aus dem Host Rechner wird von *VirtualBox* eine eigene Funktion angeboten. Dieser Vorgang wird durch ein Skript auf der *KuBox* automatisiert. Ein beliebiger Ordner vom host OS kann in der *VirtualBox* unter Geräte > Gemeinsame Ordner als *neuer gemeinsamer Ordner* mit dem Namen `sharedMusic` von dem entsprechenden Pfad hinzugefügt werden. Der Name *sharedMusic* als Bezeichnung des gemeinsamen Ordners in der *VirtualBox* ist Voraussetzung für die automatische Einbindung.

Anschließend kann durch klicken auf das Icon `sharedMusic` einbinden (siehe Abbildung 5.2) am Desktop der *KuBox* das Verzeichnis automatisch eingebunden werden. Zum Ausführen dieses Skripts ist unter Umständen die Eingabe des `root` Passworts erforderlich. Diese Funktion erlaubt das nachträgliche Einbinden des Verzeichnisses `sharedMusic`. Die Mountoptionen für die *KuBox* sind in der Datei `/etc/fstab` festgelegt. Damit erscheint das Verzeichnis `sharedMusic` sowohl im guest OS als auch im host OS - es kann von beiden Orten darauf zugegriffen werden.



**Abbildung 5.2:** Icon vom Desktop der *KuBox* zum Einbinden des Verzeichnisses `sharedMusic`

### Ports Weiterleiten

Um eine Netzwerkverbindung vom host OS auf das guest OS zu ermöglichen, die für den Zugriff des Clients auf das Webinterface vom AMSL benötigt wird, müssen für die jeweiligen Services die entsprechenden Ports freigeschaltet werden. Dies wird über ein *Port-Forwarding*, wie es auch bei Firewalls eingesetzt wird, gelöst. Tabelle 5.2 zeigt die Zuordnung der weitergeleiteten TCP-Ports am host OS und am guest OS. Um Konflikte mit Diensten am host OS zu vermeiden, wurden hohe Portnummern gewählt, die höchstwahrscheinlich noch nicht verwendet werden. *VirtualBox* stellt für die Konfiguration der Weiterleitung den Befehl `VBoxManage` mit der Option

setextradata zur Verfügung. Um dies am host OS zu auszuführen ist das Script `forwardingPorts.bat` (für Windows) und `forwardingPorts.sh` (für Linux) auf der Installations-DVD enthalten. Eine genauere Erklärung dazu ist im Abschnitt *Ports einrichten* auf Seite 58 zu finden.

Service	Port am host	Port am guest
Webserver lighttpd	10080	10080
Webserver Tomcat	18080	8080
SSH	10022	22

**Tabelle 5.2:** Weitergeleitete TCP-Ports vom host OS auf das guest OS

### Webserver - Lighttpd

Er stellt die nötigen Webservices für *Ampache* zur Verfügung. Im Vergleich zum Webserver *Apache http*, der eher für größere Serveranwendungen entwickelt wurde, ist *lighttpd* ressourcenschonender und eher für kleine Systeme gedacht. Die angebotenen Dienste reichen für die Anforderungen des AMSL (internes Netzwerk und geringer Traffic) vollkommen aus.

PHP5 läuft in *lighttpd* nicht standardmäßig und muss daher manuell aktiviert werden. Dazu muss im Verzeichnis `/etc/lighttpd/conf-enabled` ein Link mit der Datei `/etc/lighttpd/conf-available/10-fastcgi.conf` erstellt werden. Anschließend muss in dieser Datei der Eintrag `/usr/bin/php4-cgi` auf `/usr/bin/php5-cgi` ausgebessert werden. Dafür muss das Paket *php5-cgi* installiert sein.

Auf Grund der im Absatz *Ports Weiterleiten* auf Seite 54 beschriebenen Änderungen der Portnummern muss der Standardport des *lighttpd* Webservers von 80 auf 10080 gesetzt werden. Dies kann in der Datei `/etc/lighttpd/lighttpd.conf` bei dem Eintrag `server.port = 80` vorgenommen werden. Dadurch kann *Ampache* auch lokal in der KuBox im vollen Funktionsumfang verwendet werden.

Der Serverdienst muss vor dem Laden der Startseite von *Ampache* selbstverständlich gestartet sein, dies wird in der *KuBox* über einen Link im Verzeichnis des jeweiligen Runleves<sup>9</sup> automatisch beim Start ausgeführt. (`/etc/rc<x>/` wobei `<x>` für die Runlevelnummer steht)

### Ampache

Die Webseite von *help.ubuntu*<sup>10</sup> bietet eine ausführliche Installationsanleitung für *Ampache* auf einem *Ubuntu-Linux* System inklusive der benötigten Zusatzprogramme. Da

---

<sup>9</sup>Eine Erklärung der Runlevels in Linux ist auf folgender Webseite zu finden: <http://www.skullbox.net/init.php>

<sup>10</sup>Installationsanleitung: *Ampache* auf Ubuntu: <https://help.ubuntu.com/community/ampache>

in der *KuBox lighttpd* als Webserver verwendet wird, müssen die Einstellungen während der Installation entsprechend angepasst werden. Die Punkte 8 - 10 müssen nicht ausgeführt werden.

Nach der Installation können Konfigurationen (wie Passwordeingaben und Erstellen des Admin-Benutzers) über das Webinterface von *Ampache* durchgeführt werden. Diese Angaben werden abschließend in der Datei `ampache.cfg.php` gespeichert.

Der Webbrowser *Firefox* hat in der *KuBox* die *Ampache*-Login Seite als eine der Startseiten eingetragen und erscheint somit automatisch beim Laden. Zusätzlich wurde die Seite zu den Bookmarks hinzugefügt.

Als Passwort für den User `admin` wurde `DAMedia` angegeben, dieses kann im Webinterface von *Ampache* jederzeit nachträglich geändert werden.

Nach dem ersten Start von *Ampache* müssen vorerst die Pfade der Musikverzeichnisse angegeben werden, die in *Ampache* eingebunden werden sollen. Als Standard Verzeichnisse sind das lokale Musikverzeichnis (`/KuBoxMusic`) und das freigegebene Verzeichnis aus dem host OS `/sharedMusic` eingetragen. Weitere Pfade können im Nachhinein hinzugefügt werden.

Des Weiteren muss für den lokalen Betrieb in der *KuBox* im Menü unter `Admin > Config > System` für den *Non-Standard Http Port* `10080` eingetragen werden, um mit den Einstellungen des Port-Forwardings konform zu sein. (siehe dazu auch Abschnitt *Ports Weiterleiten* auf Seite 54)

### Java Servlet - Tomcat

Dieser oder ein ähnlicher Java-Servlet Container ist, gemeinsam mit *Java 5*, Voraussetzung für den Betrieb von *Subsonic*. Auf der Projektwebseite wird eine Version ab 5.5 empfohlen, hier in der *KuBox* wird die aktuellste Version - *Tomcat6* - verwendet. Eine Anleitung zur Installation ist im Forum von Ubuntu zu finden.<sup>11</sup>

Der Ordner von *Tomcat6* ist auf der *KuBox* unter dem Pfad `/usr/share/apache-tomcat-6.0.10/` zu finden.

Auch dieser Serverdienst muss vor Inbetriebnahme von *Subsonic* gestartet sein. Der Startup Befehl wird ebenfalls beim Start der *KuBox* automatisch ausgeführt.

### Subsonic

Nach dem *Tomcat* installiert ist und alle Konfigurationen bezüglich der *Java*-Engine richtig gesetzt sind, braucht nur die Datei `subsonic.war` in das Verzeichnis `<TOMCAT_HOME>/webapps` kopiert und der *Tomcat*-Server neu gestartet werden. Danach kann das *Subsonic* Login-Fenster in der *KuBox* über den entsprechenden Pfad erreicht werden.

Auch die *Subsonic*-Login Seite ist im Webbrowser *Firefox* als eine der Startseiten eingetragen, somit erscheint auch sie beim Laden automatisch. Ein Bookmark für diese Seite ist ebenfalls eingetragen.

---

<sup>11</sup>Tomcat6 installation guide for Feisty: <http://ubuntuforums.org/showthread.php?t=420034&highlight=tomcat6>

Als standard Benutzername und -passwort ist `admin/admin` vorgegeben, dies kann jedoch im Menü `Settings > Users` jederzeit nachträglich geändert werden.

Die Pfade der Musikverzeichnisse können über `Settings > Music folders` angegeben werden. Zu den standardmäßig erscheinenden Eingabezeilen können weitere hinzugefügt werden. Auch hier ist das lokale Musikverzeichnis `/KuBoxMusic` und das Verzeichnis `/sharedMusic` bereits eingetragen. Des Weiteren ist ein Ordner für die Playlists unter dem Pfad `/playlists` angelegt, auf welchen alle BenutzerInnen zugreifen können.

Wenn eine neue Version von *Subsonic* verfügbar ist, wird automatisch am rechten oberen Bildschirmrand eine Meldung angezeigt (siehe Abbildung 5.3). Über den Link zur Webseite kann dann die neue Version heruntergeladen werden. Auf der Download Seite ist unter *Upgrading Subsonic* eine Anleitung für die nötigen Schritte zum Upgrade zu finden.

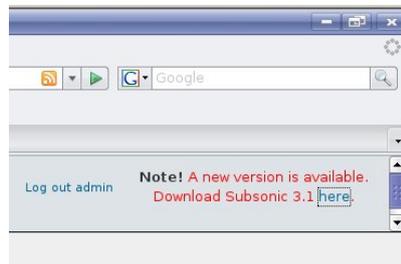


Abbildung 5.3: Versions Upgrademeldung in Subsonic

### Amarok Musikplayer

Als Standard Musikplayer auf der *KuBox* ist *Amarok*<sup>12</sup> ausgewählt - daher werden alle Playlists, die innerhalb der *KuBox* aus den Programmen gesendet werden, automatisch hier abgespielt.

Um über *Amarok* MP3 Daten abspielen zu können, müssen zusätzlich die Pakete `libmad0` und `libxine-extracodecs` installiert werden.

Des Weiteren kann *Amarok* auch als lokaler Musikmanager innerhalb der *KuBox* verwendet werden. Hierfür können Musikverzeichnisse der *KuBox* direkt in die Musiksammlung von *Amarok* geladen und von dort abgespielt werden. Die Standardordner `/KuBoxMusic` und `/sharedMusic` sind bereits der Sammlung hinzugefügt.

### Ordner im Konqueror

Die standard Musikverzeichnisse `/KuBoxMusic` und `/sharedMusic` sind, wie schon oben erwähnt, im `root` Verzeichnis der *KuBox* angelegt. Außerdem wurden sie als Lesezeichen hinzugefügt.

---

<sup>12</sup>Amarok Musikplayer: <http://amarok.kde.org/>

## 5.2.3 Verwendung der KuBox

### Kubuntu Update

Verfügbare Softwareupdates (Batches für die aktuelle Version) der verwendeten Programme können auf Wunsch installiert werden. Sobald neue Updates verfügbar sind, erscheint in der rechten unteren Bildschirmcke ein Symbol (siehe Abbildung 5.4). Von dort aus kann auch die Installation gestartet werden. Durch klicken auf das Symbol erscheint ein Fenster, in welchem nur den Anleitungen gefolgt werden muss.



Abbildung 5.4: Symbol für verfügbare Updates für Kubuntu

Software Upgrades (neuere Versionen der jeweiligen Programme) müssen manuell über die Paketverwaltung *Adept* gestartet werden.

### Ports einrichten

Damit der Zugriff via Netzwerk auf die *KuBox* möglich ist, muss das in Abschnitt *Ports Weiterleiten* auf Seite 54 beschriebene Port-Forwarding am host OS aktiviert werden. Um die entsprechenden Ports (10022, 10080 und 18080) möglichst einfach freischalten zu können, wurde je ein Script für Windows (*forwardingPorts.bat*) und Linux (*forwardingPorts.sh*) auf der Installations-DVD beigelegt. Das Script verwendet den Namen „*KuBox*“ als Bezeichnung der Maschine, in der *VirtualBox* (die richtige Bezeichnung ist notwendig für die korrekte Funktion des Scripts).

Außerdem ist zu beachten, dass in der Firewall am host OS die Ports 10022, 10080, 18080 aktiviert sein müssen, damit auch andere Clients aus dem Netzwerk auf die *KuBox* zugreifen können.

### Ampache

Die **Ampache Startseite** kann auf der *KuBox* über folgenden Pfad erreicht werden:

```
localhost:10080/ampache/login.php
```

Zum Erreichen von *Ampache* als Client über das Netzwerk muss anstatt *localhost* die entsprechende IP-Adresse des host Rechners angegeben werden.

Eine Beschreibung der **Konfigurationsmöglichkeiten** bei einem *admin* Zugang, die über das Menü *Admin > Config* getätigt werden können, ist auf deren Projektwebseite zu finden.<sup>13</sup> Getroffene Änderungen müssen auf jeder Seite extra gespeichert werden.

---

<sup>13</sup>Configuring Ampache: <https://ampache.bountysource.com/wiki/Configuration>

**Add User** Weitere BenutzerInnen können über `Admin > Users` angelegt werden. Dabei kann aus den drei verschiedenen Gruppen gewählt werden:

- \* **admin** - besitzt alle Berechtigungen
- \* **user** - hat teilweise eingeschränkte Rechte, der Menüpunkt *Admin* ist nicht vorhanden, kann keine neuen Accounts anlegen, keine Musikkataloge hinzufügen und die ID3-Tags nicht editieren
- \* **guest** - hat sehr eingeschränkte Rechte, Einstellungen können nicht geändert werden

Nach dem **Hinzufügen von neuen Musikdaten** in eines der Verzeichnisse müssen die Kataloge neu geladen werden. Dies kann bei einem *admin* Zugang über das Menü `Admin > Catalog > Update All` ausgeführt werden. Dabei ist jedoch im Speziellen für *Ampache* zu beachten, dass hier nur Dateien mit den entsprechenden **Berechtigungen** (`-r--r--r--`) hinzugefügt werden können. Die jeweiligen Rechte können in einem Linux System über die Kommandozeile mit folgendem Befehl gesetzt werden (für das Verzeichnis `sharedMusic` müssen die Berechtigungen am host OS geändert werden):

```
# chmod a+r <Verzeichnisname> -R
```

Um den Titeln Informationen hinzufügen zu können, müssen auch die Schreibrechte auf die entsprechende Datei vergeben sein. Dies kann mit dem folgenden Befehl durchgeführt werden:

```
# chmod a+w <Verzeichnisname> -R
```

**Player** Die Auswahl, ob die Playlist über den Modus *Localplay* oder *Stream* abgespielt werden soll, kann entweder über das Menü auf der linken Bildschirmseite oder über `Admin > Config > Streaming` bei einem *admin* Zugang, oder über `Preferences > Streaming` bei einem *user* Zugang getätigt werden. Wobei der *Localplay* Modus offensichtlich nur bei einem *user* Account funktioniert. Hier wird die Playlist über den MPD-Player direkt am Rechner abgespielt (MPD muss dazu gestartet sein). Wird ein Titel von einem Client aus dem Netzwerk über den *Localplay* Modus gestartet, wird dieser trotzdem auf dem host Rechner der *KuBox* abgespielt. Wenn auf dem host Rechner gleichzeitig gearbeitet wird, können hickups auftreten (nur beim *Localplay* Modus). Der *Stream* Modus ermöglicht das Abspielen der Playlist über einen Musikplayer auf einem entfernten Rechner im Netzwerk. Es können mehrere Clients gleichzeitig und unabhängig voneinander per Stream Musik von *Ampache* empfangen.

**Playlists** können unter dem selbigen Menüpunkt aus einer Datei importiert oder neu angelegt werden. Die erstellten Listen können entweder als *private* oder *public* markiert werden. Dem entsprechend sind die Playlists bei anderen Accounts sichtbar oder nicht. *Admin* Accounts können alle Listen sehen und gegebenenfalls leere Listen entfernen. Titel aus sämtlichen Verzeichnissen können jeder beliebigen, angezeigten Playlist hinzugefügt werden. Während des Abspielens einer Playlist muss der/die jeweilige BenutzerIn angemeldet bleiben, sobald sich der User ausloggt, wird die Wiedergabe im Musikplayer vorzeitig gestoppt.

**Informationen zu Tracks speichern** ist nur über einen *admin* Zugang möglich. Über das Symbol *Edit* in der Spalte *Action* der Trackliste wird ein Formular zum editieren der ID3-Tags geöffnet. Dort kann ein Kommentar zum Titel hinzugefügt werden. Durch klicken auf das Symbol **Download** kann der jeweilige Track auf den Clientrechner heruntergeladen werden.

## Subsonic

In der *KuBox* kann die **Subsonic-Login Seite** über folgenden Pfad erreicht werden:

```
localhost:8080/subsonic
```

Wenn das Programm über das Netzwerk gestartet werden soll, muss anstatt *localhost* der entsprechende *hostname* angegeben werden. Des Weiteren ist zu beachten, dass als Port die Nummer 18080 verwendet werden muss. (siehe dazu auch Abschnitt *Ports Weiterleiten* auf Seite 54) (Beispielpfad für das Erreichen von *Subsonic* über einen Client im Netzwerk `http://192.168.2.7:18080/subsonic`)

Nach dem **Laden neuer Verzeichnisse** werden die Titel in der Anzeige automatisch in die Liste aufgenommen (nach Reload der Page). Um die Tracks auch über die Suchfunktion finden zu können, muss der Suchindex über das Menü `Settings > Search > Update search index now` aktualisiert werden. Dieser Vorgang dauert einige Minuten, währenddessen kann das Programm jedoch weiter verwendet werden. Es besteht auch die Möglichkeit, das Update des Suchindex zu einer bestimmten Zeit automatisch durchführen zu lassen. Die Pfade weiterer Musikquellen können über `Settings > Music folders` eingetragen werden.

Unter dem Menüpunkt `Settings > Appearance` können **Einstellungen** wie Sprache, Themes und Informationen, die zu einem Track angezeigt werden sollen, geändert werden. Der Punkt `Settings > Transcoding` zeigt eine vorkonfigurierte Transcoding Tabelle, über welche Tracks, die in anderen Dateiformaten gespeichert sind, in *.mp3* umkodiert werden können. Bei Internet TV/radio können über das Web verfügbare Streams geladen werden.

**Add User** Das Hinzufügen von weiteren und das Editieren von bereits bestehenden BenutzerInnen kann über das Menü `Settings > Users` vorgenommen werden. Jedem Account kann eine Liste von Funktionen freigegeben werden:

- \* User is administrator
- \* User is allowed to download files
- \* User is allowed to upload files
- \* User is allowed to create and delete playlists
- \* User is allowed to change cover art and tags
- \* User is allowed to create and edit comments and ratings

Bei Usern, die keine Administratoren sind, erscheinen im Menüpunkt `Settings` nur die Einträge *Appearance*, *Password* und *Players*.

Einschränkungen in den **Dateiberechtigungen** stellen bei *Subsonic*, im Gegensatz zu *Ampache*, keine Probleme dar, da es offensichtlich mit root Privilegien läuft. Hier

ist sowohl das Abspielen, als auch das Editieren jener Dateien möglich, die nur für den Benutzer *root* lesbar sind (`-r-----`). Diese erweiterten Privilegien können ein potentiell Sicherheitsrisiko darstellen.

Einzelne Tracks oder ganze Alben können durch klicken auf das Add Symbol [+] einer **Playlist** hinzugefügt werden. Die zusammengestellte Liste wird im Bottom-frame angezeigt. Dort können einige Einstellungen direkt getroffen werden, weitere (zum Beispiel *Load*, *Save* oder *Download*) können über die Auswahlliste *More actions...* gewählt werden. Es können hier auch Listen von anderen Usern geladen werden.

**Player** Alle Tracks oder Playlisten, die abgespielt werden sollen, schickt *Subsonic* an den lokalen standard Musikplayer als *.m3u* Datei. Wenn eine *.wav* Datei abgespielt werden soll, muss von einem Administrator beim jeweiligen Player das Transcodieren aktiviert und die gewünschte maximale Bitrate gesetzt werden (`Settings > Players > Select Player | Activate transcodings | Max bitrate`).

**Download / Upload** Die einzelnen Musikdateien oder auch ganze Alben (diese werden als *.zip* Datei gespeichert) können auch direkt auf die Festplatte heruntergeladen werden. Außerdem können einzelne Tracks über den Menüpunkt *More > Upload file* direkt mit *Subsonic* in das lokale Musikverzeichnis */KuBoxMusic/Incoming* kopiert werden. Verzeichnisse können als *.zip* importiert und auf Wunsch automatisch entpackt werden.

**Informationen zu Tracks speichern** In der *Album* Ansicht können dessen Tags editiert und zusätzliche Kommentare hinzugefügt werden. Des Weiteren können unter *Album info* Informationen, die im Internet auf den Seiten *Amazon.com*, *allmusic.com* und *Google Music* dazu verfügbar sind, nachgelesen werden.

## 5.3 Erweiterungsmöglichkeiten

### 5.3.1 Datensicherung

Wie bei herkömmlichen Computerdaten ist die Sicherung auch bei Audiodaten ein wichtiger Aspekt. Die verwendeten Technologien, im Speziellen die Festplatten, sind zwar sehr sichere Speichermedien, es kann jedoch vor allem im Dauerbetrieb immer wieder zu unerwarteten Datenverlusten kommen. Deshalb empfiehlt es sich, von wichtigen Daten immer eine Sicherungskopie anzulegen, wenn auch Musikdaten, die aus eigenen CD-Sammlungen generiert wurden, relativ einfach wieder herstellbar sind.

Eine weitere Möglichkeit stellt eine redundante Datenspeicherung, wie sie RAID bietet, dar. Hier „... wird die Arbeit mehrerer Festplatten-Laufwerke durch einen RAID-Controller koordiniert. Dieser verteilt die Datenlast auf die einzelnen Festplatten und reduziert das Fehler- und Verlustrisiko.“ [zit. IT WISSEN, 07.08.2007 (Webseite) [ITW07d]]

Diese Art von Datenspeicherung wird vor allem in Servercomputern und Storage-Devices, aber immer häufiger auch im PC verwendet.

### 5.3.2 Anspruch auf hohe Qualität

Bei einer Verwendung des AMSL in Kombination mit einer guten Stereoanlage sind die Qualitätsmängel von hoch komprimierten MP3s sehr deutlich hörbar. Daher sollte beim Anlegen der Sammlung darauf geachtet werden, die Titel in einer angemessenen Qualität (ab 192 kbps) abzuspeichern.

### 5.3.3 Tonstudio

Für professionelle Anwendungen, wie zum Beispiel für den Einsatz im Tonstudio, ist das System in dieser Form nicht geeignet. Hier besteht eher der Bedarf an zentralen Archivierungsmöglichkeiten und uneingeschränkter Verfügbarkeit von Audiodateien im Rohformat auf allen Arbeitsplätzen. Sowohl die erweiterte Suche nach bestimmten Kriterien, als auch der direkte Zugriff auf alle Daten, sind wichtige Anforderungen an ein System für das Tonstudio. Auch die Datensicherheit und Zuverlässigkeit des Systems spielen in diesem Zusammenhang eine wesentliche Rolle.

### 5.3.4 Ultrastar - Singstar mit OpenSource

Dieses OpenSource Projekt bietet ein mit Singstar (ein Karaoke-Spiel für die Sony PlayStation2) vergleichbares Karaoke-Programm. Zu finden ist es auf der Sourceforge Seite<sup>14</sup>.

Ultrastar Songs sind in einzelnen Verzeichnissen abgelegt und bestehen aus der jeweiligen *.mp3* Datei, einer *.txt* Datei (hier werden der Songtext, die Midi-Daten und der Timecode gespeichert), optional kann das Cover oder Musikvideo zum Song als Hintergrundbild mitgespeichert werden.

Eigene Songs können mit dem Ultrastar-Editor selber erstellt oder angepasst werden. Dazu wird die *.mp3* Datei und der dazugehörige Songtext benötigt. Über den enthaltenen Midi-Converter können dann die entsprechenden Dateien generiert werden. Eine Anleitung dazu ist auf der deutschen Projektwebseite zu finden.<sup>15</sup>

Die ursprüngliche Version wurde für Windows entwickelt, mittlerweile ist auch eine Version für Linux in Entwicklung.

---

<sup>14</sup><http://ultrastardx.sourceforge.net>

<sup>15</sup>Anleitung zum Editieren von Songs für Ultrastar: [http://ultrastar.byto.de/guide\\_internaeditor.html](http://ultrastar.byto.de/guide_internaeditor.html)

### Ergebnisse

---

Ausgehend von der forschungsleitenden Fragestellung (siehe Kapitel 1.3) wird nach einer Recherche zu proprietären Medienserver Systemen eine Anforderungsliste erstellt. An Hand dieser werden OpenSource Produkte zur Medienverwaltung getestet und verglichen.

Für den Test werden vier Produkte herangezogen, die dem Auswahlkriterium der Plattformunabhängigkeit entsprechen. Nach der Gegenüberstellung erweisen sich *Ampache* und *Subsonic* als geeignete Komponenten für einen **Audio Medien Server** auf Linux Basis (AMSL).

Als Grundlage für das System des AMSL stehen, neben dem Betrieb auf einem normalen PC, drei weitere Technologien zur Verfügung, die in Kapitel 4.2 vorgestellt werden. Für die Umsetzung des AMSL wird ein virtuelles System ausgewählt, auf welchem alle benötigten Komponenten vorinstalliert sind.

Das Ergebnis ist eine DVD, auf der das fertig konfigurierte virtuelle Image (*KuBox*), die Installationsdateien für die *VirtualBox* (Linux- und Windowsversion), eventuell benötigte Konfigurationsskripten für die Inbetriebnahme sowie eine Readme-Datei enthalten sind.

#### 6.1 Diskussion der Ergebnisse

Die zu Beginn definierte Fragestellung kann abschließend durchaus zufriedenstellend beantwortet werden. Durch die Vielzahl der verfügbaren Komponenten auf OpenSource Basis, sind sehr vielfältige Kombinationen möglich. Somit kann das System sowohl in der Wahl der Hardware, als auch mit der Software nach den eigenen Ansprüchen angepasst werden. Ganz im Gegensatz zu den proprietären Systemen, die teilweise an deren Sendestationen und Empfangsclients gebunden sind.

Für die Umsetzung des AMSL sind grundsätzlich alle der vorgestellten Technologien geeignet. Da für jedes System spezielle Anpassungen nötig sind, um einen den Anforderungen entsprechenden Medienserver zu erhalten, wird im Rahmen der Diplomarbeit nur eines vollständig umgesetzt. Die Wahl fällt nach ausführlichen Tests

auf die *VirtualBox* mit dem Betriebssystem *Kubuntu 7.04* - daraus ergibt sich der Hostname *KuBox*.

Die Zusammensetzung des AMSL ist in einem Strukturdiagramm (siehe Kapitel 4.1) veranschaulicht. Anschließend daran werden die einzelnen Komponenten und Technologien behandelt. Dadurch wird ein guter Überblick über die Funktionalität des AMSL gegeben. Im Speziellen sind folgende Dienste über den AMSL verfügbar:

- \* Archivierung von Audiodaten
- \* Importieren von Daten aus beliebigen Speichermedien
- \* Einlesen von Audio-CDs
- \* zwei gleichwertig einsetzbare Webinterfaces (*Ampache* und *Subsonic*), die plattformunabhängig von Browsern der Clientrechner betrieben werden können
- \* CDDDB-Lookup
- \* Abspielen der Musik über Clients im lokalen Netzwerk
- \* Erstellen und Editieren von Playlists
- \* Verwaltung der Daten über eine Datenbank
- \* Datenaustausch zwischen Server und Client mittels Samba Server
- \* Exportieren von Daten

Mit der *KuBox* steht ein vielseitiges System für den AMSL zur Verfügung, das einfach speziellen Anforderungen angepasst und jederzeit erweitert werden kann. Dennoch kann über das virtuelle Image ein vollständig konfiguriertes Produkt zur Verfügung gestellt werden, das für die Inbetriebnahme nur noch wenige Anpassungen auf dem host OS bedarf. Die Bedienung der *KuBox* ist so einfach wie möglich gehalten. Um jedoch Veränderungen am System vornehmen zu können, bedarf es gewisser Kompetenzen der BenutzerInnen im Umgang mit Linux-Systemen, da das Endprodukt auf Grund seiner Vielseitigkeit recht komplex ist.

Das Endergebnis der vorliegenden Diplomarbeit - die *KuBox* - entspricht bis auf einige wenige Punkte der zu Beginn definierten Anforderungsliste:

- \* Audio-CDs können nicht direkt in der *KuBox* verwendet werden, deren Daten müssen über das host OS eingelesen und dann in ein gemeinsames Verzeichnis kopiert werden
- \* Einträge in die Datenbank sind nur im jeweiligen Programm verfügbar - es gibt keine Möglichkeit, die Daten gesammelt außerhalb der Medienverwaltungssoftware zu betrachten oder editieren
- \* Erweiterungen der OpenSource Tools waren nicht nötig, da die Programme *Ampache* und *Subsonic* weitgehend den Anforderungen entsprechen

Abschließend kann die *KuBox* als sehr zufriedenstellendes Endergebnis präsentiert werden, welches im praktischen Einsatz innerhalb eines lokalen Netzwerks einwandfrei funktioniert.

## 6.2 Ausblick

Eine interessante Umsetzung des Konzepts ist die Verwendung als zentraler Home-Medienserver, der im Dauerbetrieb sämtliche Daten allen beteiligten Clients zur Verfügung stellt. Bei den angeführten Technologien ist deren Weiterentwicklung auf alle Fälle verfolgenswert. Vor allem das System des *Schluck* kann mit zukünftigen Versionen der Software, entsprechend den Funktionen der *KuBox*, erweitert werden.

Die verwendete Software befindet sich seitens der Projektteams in ständiger Weiterentwicklung - verfügbare Updates werden in der *KuBox* automatisch angezeigt und können dort einfach nachinstalliert werden. Nach dem Release von neuen Softwareversionen kann eine *KuBox2* mit den entsprechenden Komponenten zusammengestellt werden.

Im Rahmen dieser Diplomarbeit ist das Hauptaugenmerk für die Zusammensetzung des Systems auf den Audiobereich gerichtet. Als sinnvolle Erweiterung des Projekts würde sich durchaus eine Aufweitung auf den gesamten Multimedia-Bereich anbieten. Vor allem das Einbinden einer Verwaltungsmöglichkeit für digitale Fotos und Videofilme könnte hier interessant sein.

Digitales Home-Entertainment gewinnt immer mehr an Bedeutung, da diese Technologien mittlerweile auch für Konsumer preislich interessant sind. Für die dadurch anfallenden Datenmengen werden vermehrt Verwaltungssysteme benötigt. Auf Grund der hohen Lizenzkosten von proprietären Systemen werden OpenSource Produkte an Popularität gewinnen. Somit stellt das Konzept des AMSL eine zukunftsweisende Technologie dar.

---

## Literaturverzeichnis

---

- [AMP07] AMPACHE [webpage]. About ampache, 2007 [cited 17. April 2007]. Link: <http://www.ampache.org/>.
- [APA07] APACHE TOMCAT [webpage]. The apache software foundation, 2007 [cited 24. April 2007]. Link: <http://tomcat.apache.org/>.
- [ASB07] ASBYON [webpage]. Flash-Speicher, 2007 [cited 2. August 2007]. Link: [http://www.asbyon.com/screen/wissen/lexikon/gl\\_055\\_0.asp](http://www.asbyon.com/screen/wissen/lexikon/gl_055_0.asp).
- [BET04] Joe BETZ. *MP3 Musik finden, laden, hören, brennen*. Markt+Technik, München, 2004.
- [DIC97] Michael DICKREITER. *Handbuch der Tonstudioteknik, Band 2*. K.G. Saur, München, 1997.
- [DKSZ03] Ali E. DASHTI, Seon Ho KIM, Cyrus SHAHABI, and Roger ZIMMERMANN. *Streaming Media Server Design*. IMSC Press Multimedia Series, New Jersey, 2003.
- [FIL07] FILExt [webpage]. Filext - the file extension source, 2007 [cited 24. April 2007]. Link: <http://filext.com/>.
- [FRA07] FRAUENHOFER INSTITUT [webpage]. MPEG Audio Layer-3, 2007 [cited 19. Juli 2007]. Link: <http://www.iis.fraunhofer.de/bf/amm/projects/mp3/index.jsp>.
- [GNU07] GNU [webpage]. The free software definition, 2007 [cited 6. August 2007]. Link: <http://www.gnu.org/philosophy/free-sw.html>.
- [ID307a] ID3 [webpage]. The audience is informed, 2007 [cited 1. Mai 2007]. Link: <http://www.id3.org/>.
- [ID307b] ID3 [webpage]. What is id3 (v1)?, 2007 [cited 1. Mai 2007]. Link: <http://www.id3.org/ID3v1?highlight=%28id3v1%29>.
- [ID307c] ID3 [webpage]. Id3 tag version 2, 2007 [cited 14. August 2007]. Link: <http://www.id3.org/id3v2.3.0>.
- [INN07] INNOTEK. innotek VirtualBox®User Manual. 1.4, 2007. <http://www.virtualbox.org/download/UserManual.pdf>.
- [ITW06] ITWissen [webpage]. Media-Server, 2006 [cited 14. Dezember 2006]. Link: [http://www.itwissen.info/definition/lexikon/\\_\\_\\_media%20server\\_media-server.html](http://www.itwissen.info/definition/lexikon/___media%20server_media-server.html).

- [ITW07a] ITWISSEN [webpage]. CDDb (compact disc database), 2007 [cited 14. August 2007]. Link: [http://www.itwissen.info/definition/lexikon//\\_cddb\\_cddb\\_compact%20disc%20database\\_cddb\\_cddb.html](http://www.itwissen.info/definition/lexikon//_cddb_cddb_compact%20disc%20database_cddb_cddb.html).
- [ITW07b] ITWISSEN [webpage]. Datenrate, 2007 [cited 2. August 2007]. Link: <http://www.itwissen.info/?id=31\&ano=01-004148>.
- [ITW07c] ITWISSEN [webpage]. UTF-8 (unicode transformation format 8), 2007 [cited 3. August 2007]. Link: [http://www.itwissen.info/definition/lexikon//\\_utf-8utf-8\\_utf-8unicode%20transformation%20format%208utf-8\\_utf-8.html](http://www.itwissen.info/definition/lexikon//_utf-8utf-8_utf-8unicode%20transformation%20format%208utf-8_utf-8.html).
- [ITW07d] ITWISSEN [webpage]. RAID (redundant array of inexpensive disks), 2007 [cited 8. August 2007]. Link: [http://www.itwissen.info/definition/lexikon//\\_raidraid\\_raidredundant%20array%20of%20inexpensive%20disksraid\\_raidraid-system.html](http://www.itwissen.info/definition/lexikon//_raidraid_raidredundant%20array%20of%20inexpensive%20disksraid_raidraid-system.html).
- [JAV07a] JAVVIN [webpage]. Media stream, 2007 [cited 1. August 2007]. Link: <http://www.javvin.com/networkingterms/MediaStream.html>.
- [JAV07b] JAVVIN [webpage]. Unicast, 2007 [cited 1. August 2007]. Link: <http://www.javvin.com/networkingterms/Unicast.html>.
- [JIN07] JINZORA [webpage]. The ultimate web based media streaming and management system, 2007 [cited 17. April 2007]. Link: <http://www.jinzora.com>.
- [KOF06] Michael KOFLER. *Linux - Installation, Konfiguration, Anwendung*. Addison-Wesely, München, 2006.
- [LAM07] LAME MP3 Encoder [webpage]. The lame project, 2007 [cited 25. April 2007]. Link: <http://lame.sourceforge.net/index.php>.
- [LEX07] LEXEXAKT [webpage]. Suchwort proprietär, 2007 [cited 7. August 2007]. Link: <http://www.lexexakt.de/glossar/proprietaer.php>.
- [LFS07] LFS LiveCD [webpage]. Linux from scratch, 2007 [cited 10. Juli 2007]. Link: <http://www.linuxfromscratch.org/livecd/index.html>.
- [LIN07a] LINKSYS [webpage]. A division of cisco system, inc., 2007 [cited 28. Juni 2007]. Link: <http://www.linksys.com/>.
- [LIN07b] LINUX LIVE SCRIPTS [webpage]. The ultimate way to bring your linux to life, 2007 [cited 28. Juni 2007]. Link: <http://www.linux-live.org/>.
- [MOF01] Jack MOFFITT. Ogg vorbis–open, free audio–set your media free. *Linux Journal*, 2001.
- [MPD07] MPD [webpage]. Music player daemon, 2007 [cited 15. Mai 2007]. Link: <http://www.musicpd.org/>.
- [NET07] NETWORK LAB [webpage]. Übertragungsraten im Netzwerk, 2007 [cited 3. August 2007]. Link: <http://www.nwlab.net/art/geschwindigkeit/uebertragungsraten.html>.
- [RAF02] Hannes RAFFASEDER. *Audiodesign*. Fachbuchverlag Leipzig im Carl Hanser Verlag, München Wien, 2002.

- [SAM07] SAMBA [webpage]. Opening windows to a wider world, 2007 [cited 26. Jul 2007]. Link: [http://us1.samba.org/samba/what\\_is\\_samba.html](http://us1.samba.org/samba/what_is_samba.html).
- [SLI06a] SLIM DEVICES [webpage]. Transporter, Squeezebox, 2006 [cited 12. Dezember 2006]. Link: <http://www.slimdevices.com/index.html>.
- [SLI06b] SLIM DEVICES [webpage]. Squeezebox Owners Guide, 2006 [cited 14. Dezember 2006]. Link: <http://www.slimdevices.com/documentation/Squeezebox-Owners-Guide.pdf>.
- [SLI06c] SLIM DEVICES [webpage]. Transporter Owners Guide, 2006 [cited 14. Dezember 2006]. Link: <http://www.slimdevices.com/documentation/Transporter-Owners-Guide.pdf>.
- [SLI06d] SLIM DEVICES [webpage]. Squeezebox Datasheet, 2006 [cited 15. Dezember 2006]. Link: <http://www.slimdevices.com/marketing/Squeezebox-v3-Datasheet.pdf>.
- [SUB07a] SUBSONIC [webpage]. Frequently Asked Questions, 2007 [cited 01. Mai 2007]. Link: <http://subsonic.sourceforge.net/faq.html>.
- [SUB07b] SUBSONIC [webpage]. Easy listening, 2007 [cited 17. April 2007]. Link: <http://subsonic.sourceforge.net/>.
- [TECB03] Jay TS, Robert ECKSTEIN, and David COLLIER-BROWN. *Using Samba, 2nd Edition*. O'Reilly & Associates, USA, 2003.
- [UBU07] UBUNTU [webpage]. Ubuntu Studio, 2007 [cited 6. August 2007]. Link: <https://wiki.ubuntu.com/UbuntuStudio>.
- [UNI07] UNI KÖLN [webpage]. Ungültige Dateinamen im WWW-Verzeichnis, 2007 [cited 3. August 2007]. Link: [http://www.uni-koeln.de/rrzk/www/eingabe/ungueltige\\_dateinamen.html](http://www.uni-koeln.de/rrzk/www/eingabe/ungueltige_dateinamen.html).
- [VLC07] VLC [webpage]. VideoLAN - VLC media player, 2007 [cited 17. April 2007]. Link: <http://www.videolan.org/>.
- [VMW07] VMWARE [webpage]. Desktop Virtualization, 2007 [cited 09. Juli 2007]. Link: <http://www.vmware.com/virtualization/>.
- [WAG07] WAGNER [webpage]. Eine Einführung in komprimiertes Audio mit Ogg Vorbis, 2007 [cited 28. Jul 2007]. Link: [http://simon.wwwagner.org/open-sources/for-artists/ogg-intro.html#part\\_13](http://simon.wwwagner.org/open-sources/for-artists/ogg-intro.html#part_13).
- [WEB07] WEBHOSTING [webpage]. Ratgeber für Webhosting, 2007 [cited 31. Juli 2007]. Link: <http://www.webhosting.de/fragen-antworten/lexikon/>.
- [WIK07] WIKIPEDIA [webpage]. Comparison of virtual machines, 2007 [cited 09. Juli 2007]. Link: [http://en.wikipedia.org/wiki/Comparison\\_of\\_virtual\\_machines](http://en.wikipedia.org/wiki/Comparison_of_virtual_machines).
- [XIP07] XIPH [webpage]. the xiph open source community, 2007 [cited 28. Jul 2007]. Link: <http://www.vorbis.com/faq/#what>.

- [XSP07] XSPF Web Music Player [webpage]. Plays MP3 on your website, 2007 [cited 07. Mai 2007]. Link: <http://musicplayer.sourceforge.net/#about>.
- [YAM06a] YAMAHA [webpage]. Musiccast, 2006 [cited 12. Dezember 2006]. Link: <http://www.musiccast.de/index.htm>.
- [YAM06b] YAMAHA [webpage]. Musiccast, 2006 [cited 12. Dezember 2006]. Link: <http://www.yamaha-online.de/index.php>.
- [YAM06c] YAMAHA [webpage]. Musiccast client owners manual, 2006 [cited 14. Dezember 2006]. Link: [http://www.yamaha-service.de/service-download/owners\\_manual/musiccast/MCX-A10\\_e\\_kai1.pdf](http://www.yamaha-service.de/service-download/owners_manual/musiccast/MCX-A10_e_kai1.pdf).
- [YAM06d] YAMAHA [webpage]. MusicCast Server Bedienungsanleitung, 2006 [cited 14. Dezember 2006]. Link: [http://www.yamaha-service.de/service-download/owners\\_manual/musiccast/MCX-2000\\_de.pdf](http://www.yamaha-service.de/service-download/owners_manual/musiccast/MCX-2000_de.pdf).

---

## Tabellenverzeichnis

---

2.1	allgemeine Gegenüberstellung der Systeme . . . . .	5
3.1	Gegenüberstellung der OpenSource Programme . . . . .	24
4.1	Aufbau eines MP3-Tags der Version 1.1 . . . . .	40
5.1	Programmpakete im Medienserver . . . . .	50
5.2	Weitergeleitete TCP-Ports vom host OS auf das guest OS . . . . .	55

---

## Abbildungsverzeichnis

---

2.1	Slim Devices, <a href="http://www.slimdevices.com/images/">http://www.slimdevices.com/images/</a> . . . . .	4
2.2	Yamaha MusicCast, <a href="http://www.yamaha-online.de/imgs/mcx-2000_A10_titan_one.jpg">http://www.yamaha-online.de/imgs/mcx-2000_A10_titan_one.jpg</a> . . . . .	4
2.3	Yamaha MusicCast MCX-A10, <a href="http://www.musiccast.de/bilddaten/mcx-a10_mcx-sp10_frontview_tn.jpg">http://www.musiccast.de/bilddaten/mcx-a10_mcx-sp10_frontview_tn.jpg</a> . . . . .	7
2.4	Yamaha MusicCast MCX-C15, <a href="http://www.musiccast.de/bilddaten/mcx-c15_frontview_tn.jpg">http://www.musiccast.de/bilddaten/mcx-c15_frontview_tn.jpg</a> . . . . .	7
2.5	Yamaha MusicCast MCX-CA15, <a href="http://www.musiccast.de/bilddaten/mcx-ca15_frontview_tn.jpg">http://www.musiccast.de/bilddaten/mcx-ca15_frontview_tn.jpg</a> . . . . .	8
2.6	Slim Devices, <a href="http://www.slimdevices.com/images/">http://www.slimdevices.com/images/</a> . . . . .	8
2.7	2 Wireless Squeezeboxen und ein terabyte ReadyNAS NV+, <a href="https://secure.slimdevices.com/order/images/icons/RNV-2XSB3-PROMO.jpg">https://secure.slimdevices.com/order/images/icons/RNV-2XSB3-PROMO.jpg</a> . . . . .	9
3.1	Ampache Login-Fenster, Screenshot vom Programm . . . . .	13
3.2	Ampache Interface, Screenshot vom Programm . . . . .	14
3.3	Jinzora Login-Fenster, Screenshot vom Programm . . . . .	17
3.4	Jinzora Interface, Screenshot vom Programm . . . . .	18
3.5	Subsonic Login-Fenster, Screenshot vom Programm . . . . .	19
3.6	Subsonic Interface, Screenshot vom Programm . . . . .	20
3.7	VLC Interface, Screenshot vom Programm . . . . .	22
4.1	Strukturdiagramm des Medienservers, Blockschaltbild . . . . .	26
4.2	Linksys NSLU2, <a href="http://www.linksys.com">http://www.linksys.com</a> . . . . .	27
4.3	VirtualBox von Innotek, <a href="http://www.virtualbox.org/graphics/vbox_logo2_gradient.png">http://www.virtualbox.org/graphics/vbox_logo2_gradient.png</a> . . . . .	29
4.4	WebHits Browserstatistik, <a href="http://www.webhits.de/deutsch/index.shtml?webstats.html">http://www.webhits.de/deutsch/index.shtml?webstats.html</a> . . . . .	32
4.5	Blockdiagramm ID3-Tag Version 1, <a href="http://www.id3.org/ID3v1?action=AttachFile&amp;do=get&amp;target=id3v1.1_blocks.gif">http://www.id3.org/ID3v1?action=AttachFile&amp;do=get&amp;target=id3v1.1_blocks.gif</a> . . . . .	39
4.6	Blockdiagramm ID3-Tag Version 2, <a href="http://www.id3.org/ID3v2Easy?action=AttachFile&amp;do=get&amp;target=id3v2_blocks.gif">http://www.id3.org/ID3v2Easy?action=AttachFile&amp;do=get&amp;target=id3v2_blocks.gif</a> . . . . .	40
4.7	Ordnerstruktur in einem Musikverzeichnis . . . . .	41
4.8	Entwicklung der Speicherkapazität von Festplatten, <a href="http://www.itwissen.info/media/lex_pics/id50t12.png">http://www.itwissen.info/media/lex_pics/id50t12.png</a> . . . . .	45

## Abbildungsverzeichnis

---

4.9	Flash Speichermedien, <a href="http://www.shutterbug.com/images/archivesart/1105kingston1.jpg">http://www.shutterbug.com/images/archivesart/1105kingston1.jpg</a> . . . . .	45
5.1	Loginfenster der KuBox, Screenshot . . . . .	47
5.2	Icon aus KuBox, Screenshot aus Programm . . . . .	54
5.3	Versions Upgrademeldung in Subsonic, Screenshot aus Programm . .	57
5.4	Symbol für verfügbare Updates für Kubuntu, Screenshot aus Programm	58

# Anhang

## Passwörterliste

---

### **KuBox Benutzerkonto**

- \* Benutzername: AMSLadmin
- \* Passwort: DAmelia

### **Ampache**

- \* Benutzername Admin: admin
- \* Passwort Admin: DAmelia
- \* Benutzername Gast: gustl
- \* Passwort Gast: gustl

### **Subsonic**

- \* Benutzername Admin: admin
- \* Passwort Admin: DAmelia
- \* Benutzername Gast: gustl
- \* Passwort Gast: gustl

### **MySQL-Datenbank**

- \* Datenbankname: ampache
- \* Benutzername: ampache
- \* Passwort: DAmelia

## ANHANG B

---

### **Schluck Installation**

---

Auf den folgenden Seiten ist eine Installationsanleitung für *Schluck* zu finden.

# Installation von Debian/Slug

---

Zur Installation von Debian/Slug auf dem NLSU2 muss man folgende Schritte ausführen

## upslug2 installieren

---

Die Source-Codes für upslug2 können mit dem Befehl

```
svn co http://svn.nslu2-linux.org/svnroot/upslug2/tags/LASTVERSION upslug2
```

oder von [hier](#) heruntergeladen werden.

Danach folgende Befehle als *root* ausführen:

```
cd upslug2
autoreconf -i
./configure
make
```

Eine detaillierte Anleitung zu *upslug2* gibt es auf <http://www.nslu2-linux.org/>

## Flashen des NSLU2

---



Wenn eine statische IP-Nummer verwendet werden soll (empfohlen) müssen *alle* Netzwerkeinstellungen eingetragen werden! Also auch Gateway, Netzwerksmaske und DNS-Server.

- Ein vorkompiliertes Archiv des Debian-Installers kann von <http://www.slug-firmware.net/> heruntergeladen werden.
- Den NSLU2 in den *Upgrade Modus* bringen.
  - Den NSLU2 ausschalten.
  - Alle USB-Datenträger entfernen
  - Den Resetbutton auf der Rückseite drücken und gedrückt halten
  - Den NSLU2 einschalten und den Resetbutton ca. weitere 10 Sek. gedrückt halten
  - Die Status-LED leuchtet zuerst orange und wechselt dann in ein dunkleres orange. Dann den Resetbutton sofort (!) loslassen.
  - Die Status-LED sollte jetzt abwechselnd grün/orange blinken. Falls nicht, den NSLU von der Stromversorgung trennen und mit dem flashen neu beginnen.
- Den NLSU2 mit *upslug2* flashen `upslug2 -i di-nslu2.bin`.

## Debian Installation

---

Nach dem erfolgreichen flashen bootet der NSLU2 neu. Sobald der Neustart abgeschlossen ist, leuchtet das Statuslicht grün und es kann mit der Installation über *ssh* weitergemacht werden.



Beim ersten booten dauert es einige Minuten bis die SSL-Zertifikate erstellt wurden. Nachdem der NSLU2 3mal geblippt hat, kann über *ssh* darauf zugegriffen werden.



Für eine Installation zu einem späteren Zeitpunkt kann der NSLU2 vom Netz getrennt und später neu gestartet werden.

Über die *ssh*-Verbindung kann man sich mit dem Benutzer `installer` und dem Passwort `install` anmelden. Danach gelangt man in ein Installationsmenü in dem die Installation des Debian-Systems ausgeführt werden kann.

Da von Debian-Etch nicht alle notwendigen Kernel-Module geladen werden, müssen zunächst beim Installer bei der Auswahl *Download installer components* einige der benötigten Module nachgeladen werden:

```
scsi-core-modules-2.6.18-4-ixp4xx-di: Core SCSI subsystem
usb-storage-modules-2.6.18-4-ixp4xx-di: USB storage support
partman-ext3: Add to partman support for ext3
ext3-modules-2.6.18-4-ixp4xx-di: EXT3 filesystem support
partman-auto: Automatically partition storage devices (partman)
```

Unmittelbar nach dem Auswählen der Module muss die USB-Festplatte angeschlossen und eingeschaltet werden, damit das System auf dem Datenträger installiert werden kann. (Vor der Partitionierung!!)

Durch das Modul *partman-auto* erfolgt die Partitionierung automatisch, falls gewünscht können Einstellungen manuell getroffen werden.

Danach den Installationsanweisungen folgen bis die Installation abgeschlossen ist. Zum Abschluss kann noch eine Aktualisierung des Systems und die Installation der *NSLU2-Uutils* sowie *joe* durchgeführt werden.

```
apt-get update
apt-get dist-upgrade
apt-get install nslu2-utils
apt-get install joe
```



Zur Installation von zusätzlichen Paketen dient der Befehl `apt-get install PACKETNAME`. Zum Suchen von installierbaren Paketen kann der Befehl `apt-cache search PACKET` verwendet werden.

## Installation von Medienservern

---

### Ampache

Vor der Installation von Ampache müssen folgende Pakete installiert werden:

- php5
- php5-gd
- php5-cgi
- php5-mysql
- mysql-server
- libapache2-mod-php5
- apache2
- wget

Das kann mit dem folgenden Befehl erledigt werden:

```
aptitude install php5 php5-gd php5-cgi php5-mysql mysql-server libapache2-mod-php5 apache2 wget
```

Falls gewünscht kann auch noch *phpMyAdmin* installiert werden.



Das Zugangspasswort für den Benutzer *root* soll in *phpMyAdmin* auf jeden Fall geändert werden, da sonst jeder Zugriff auf die mySQL-Datenbank bekommen kann.

Herunterladen der aktuellen Version von Ampache und entpacken. Danach das Verzeichnis umbenennen und in den Webserverpfad verschieben. Aus Sicherheitsgründen müssen auch die Berechtigungen richtig gesetzt werden. Das alles geschieht mit den folgenden Befehlen:

```
cd /tmp
wget http://apache.org/downloads/apache-3.3.3.3.tar.gz
tar -xzf apache-3.3.3.3.tar.gz
mv apache-3.3.3.3 /var/www/ampache
cd /var/www/
chown -R www-data:www-data ampache
find . -type d -exec chmod 755 {} + ;
find . -type f -exec chmod 644 {} + ;
```

Danach muss noch der Apache Webserver neu gestartet werden:

```
/etc/init.d/apache2 restart
```

Die Restliche Installation kann nun über das Webinterface durchgeführt werden.

<http://unshluck-ip/ampache/install.php>

Als *Special Feature* kann der *MPD-Player* installiert werden. Dazu müssen die Pakete *mpd* und *mpc* installiert werden:

```
aptitude install mpd mpc
```

Dann in der Konfigurationsdatei */etc/mpd.conf* den Pfad für das gewünschte Musikverzeichnis (wurde zuvor bereits bei der Ampache Installation angegeben) angeben. Abschließend in dieser Datei das Kommentarzeichen vor dem gewünschten **audio-output** entfernen (meistens passt **alsa**).

Natürlich muss der MPD-Daemon auch gestartet werden:

```
/etc/init.d/mpd start
```

## Samba konfiguration

---

Der *Samba-Server* kann mit dem Webadministrationstool WEBMIN konfiguriert werden.

Dazu muss folgender Befehl ausgeführt werden um benötigte Pakete vorab zu installieren:

```
aptitude install libnet-ssleay-perl openssl libauthen-pam-perl libio-pty-perl libmd5-perl
```

Danach das Installationspaket von <http://www.webmin.com> herunterladen und mit dem Befehl `dpkg -i webmin-XXX.deb` installieren.

## SubSonic

funktioniert leider auf Grund der fehlenden JAVA-Version 1.5 nicht. Diese steht für die vorliegende Hardware noch nicht zur Verfügung.

## Quellen

---

- Deutsche Installationsanleitung: <https://systemausfall.org/wikis/howto/DebianSlug/DebianInstaller>
- *upslug2* Installationsanleitung: <http://www.nslu2-linux.org/wiki/Main/UpSlug2>
- *upslug2*-Quellcode herunterladen: <http://trac.nslu2-linux.org/upslug2/browser/>
- Installationsanleitung für Ampache unter Debian: [https://ampache.bountysource.com/wiki/Ampache\\_on\\_Debian](https://ampache.bountysource.com/wiki/Ampache_on_Debian)
- Wiki für MPC: <http://mpd.wikia.com/wiki/Configuration>