

Sailfish OS

PC-Treff-BB Aidlingen

Günter Waller

Agenda

- Motivation
- Was ist Sailfish OS?
- Historie von Jolla und Sailfish OS
- Jolla heute
- Installation
- Die Welt der Repositories
 - Am Beispiel der Screencast App
- Erste Eindrücke:
 - Oberes Menü
 - SSH/SCP
- Demo
- Erkenntnisse, was geht nicht
- Links

Motivation

- Hatte schon immer ein Auge auf Linux auf Mobilgeräten.
 - Meistens sind die Projekte nach großen Ambitionen im Sande verlaufen.
- So war auch mein Interesse an der Firma Jolla und ihrem Sailfish OS eher marginal.
 - Zumal die Jolla Hardware (Jolla Phone und Tablet) auch wieder auf ein totes Gleis geriet.
- Doch dann erfuhr ich, dass Sailfish OS auf meinem gerade ausrangierten Sony Handy läuft.

Was ist Sailfish OS?

- Diese Frage war der Anlass zu einer Reise in die Linux [Ahnentafel](#) ([Vektorgrafik](#) aus Wikipedia). Schön sind auch die Darstellungen bei [DistroWatch](#), aber nicht ganz so akribisch.
- Dabei konnte ich feststellen, dass Wikipedia zu jeder Linux-Distribution die „Vorfahren“ auflistet.
- Schauen wir also bei Sailfish OS nach.
- Das sieht nach einer bewegten Historie aus.
- Die Ahnenreihe verrät aber nicht alles.
- Die Geschichte enthält bekannte Namen
 - Nokia
 - Microsoft
 - Intel, Samsung
- Schauen wir es uns an.
- Am besten chronologisch rückwärts.

Sailfish OS	
	
Entwickler	Jolla
Lizenz(en)	GPL und andere Lizenzen
Akt. Version	4.5.0 (Struven Ketju) (9. Februar 2023) ^[17]
Abstammung	GNU/Linux <ul style="list-style-type: none"> ↳ Red Hat Linux <ul style="list-style-type: none"> ↳ Fedora <ul style="list-style-type: none"> ↳ Moblin <ul style="list-style-type: none"> ↳ MeeGo <ul style="list-style-type: none"> ↳ Sailfish OS
Architektur(en)	ARM, Atom, MIPS
sailfish.org	

Sailfish OS: Vorfahr MeeGo

- Zitat Wikipedia: *MeeGo is a discontinued Linux distribution hosted by the Linux Foundation, using source code from the operating systems Moblin (produced by Intel) and Maemo (produced by Nokia).*
- MeeGo war in erster Linie gedacht für Smartphones, Handhelds, Tablets, Netbooks, Smart-TVs und Autos, also für Endverbraucher.
- Im September 2011 wurde bekannt, dass MeeGo zusammen mit LiMo im neuen Projekt Tizen (Samsung) aufgehen wird.
- Das Ende kam durch die Microsoft-Übernahme 2013.
- Ehemalige Mitarbeiter von Nokia gründeten das Unternehmen Jolla, welches MeeGo unter dem Namen SailfishOS aktiv weiterentwickelt.

MeeGo

MeeGo



Screenshot von MeeGo (Netbook-Oberfläche)

Entwickler	Intel, Linux Foundation, Nokia, MeeGo-Gemeinschaft
Lizenz(en)	GPL und andere Lizenzen
Akt. Version	1.2 (19. Mai 2011)
Abstammung	GNU/Linux ↳ Debian ↳ Maemo ↳ MeeGo GNU/Linux ↳ Red Hat Linux ↳ Fedora ↳ Moblin ↳ MeeGo
Architektur(en)	ARM, Atom
Sonstiges	Entwicklung eingestellt ^[1]

meego.com

Mer: zwischen MeeGo und Sailfish OS

- Das ursprüngliche Mer war eine Neuimplementierung von Maemo, portiert auf das Nokia Tablet N800. Wurde eingestellt als MeeGo erschien.
- Nokia wollte MeeGo 2010 als primäres Handy OS bis zum Strategiewechsel 2011. So blieb Intel alleine damit. Das Ende kam im September 2011 zugunsten von Tizen (Intel mit Samsung).
- Daraufhin wurde das Mer Projekt von der Community wieder aktiviert um die Codebasis und die Tools von MeeGo weiterzuentwickeln.
- Jolla, finnisches Startup aus den Resten von Nokia, nahm Mer als Basis für ein neues Betriebssystem: Sailfish OS. Ende 2013 kam das Jolla Phone heraus.
- Es gab ein weiteres Mer-Derivat namens Nemo Mobile.



OS family	Linux
Working state	Discontinued (merged with Sailfish OS)
Source model	Open source
Marketing target	Mobile
Package manager	RPM Package Manager
Platforms	ARM, x86 and MIPS
Kernel type	None (not shipped with Mer)
License	Free software <small>[clarification needed]</small>
Official website	merproject.org

Vorgänger von MeeGo: 1. Maemo

- Wikipedia: *Maemo ist ein Betriebssystem für mobile Endgeräte und eine Entwicklungs-Plattform, um Applikationen für die Nokia Internet Tablets und andere Maemo-konforme Handhelds zu programmieren. Die von Nokia initiierte Plattform besteht zum großen Teil aus angepasster freier Software, die sie mit vielen Linux-Distributionen gemein hat. Maemo war als Alternative zu Palm OS bzw. dem Nachfolger webOS, Windows CE, Symbian OS oder Android gedacht.*
- Maemo baut auf dem Linux-Kernel sowie dem freien Desktop Gnome auf. Nokia trat in diesem Zusammenhang auch als Förderer des GNOME-Projektes auf.
- Mit der Übernahme von Trolltech im Jahre 2008 hatte Nokia den GTK+-Konkurrenten Qt im eigenen Hause. Seit Maemo 5.0 wird Qt parallel zu GTK+ angeboten.

Maemo

maemo™



Entwickler	Nokia
Lizenz(en)	GPL und andere
Akt. Version	5 (Fremantle) PR 1.3.1 (Oktober 2010)
Abstammung	GNU/Linux ↳ Debian ↳ Maemo
Architektur(en)	ARM
Sonstiges	Entwicklung eingestellt

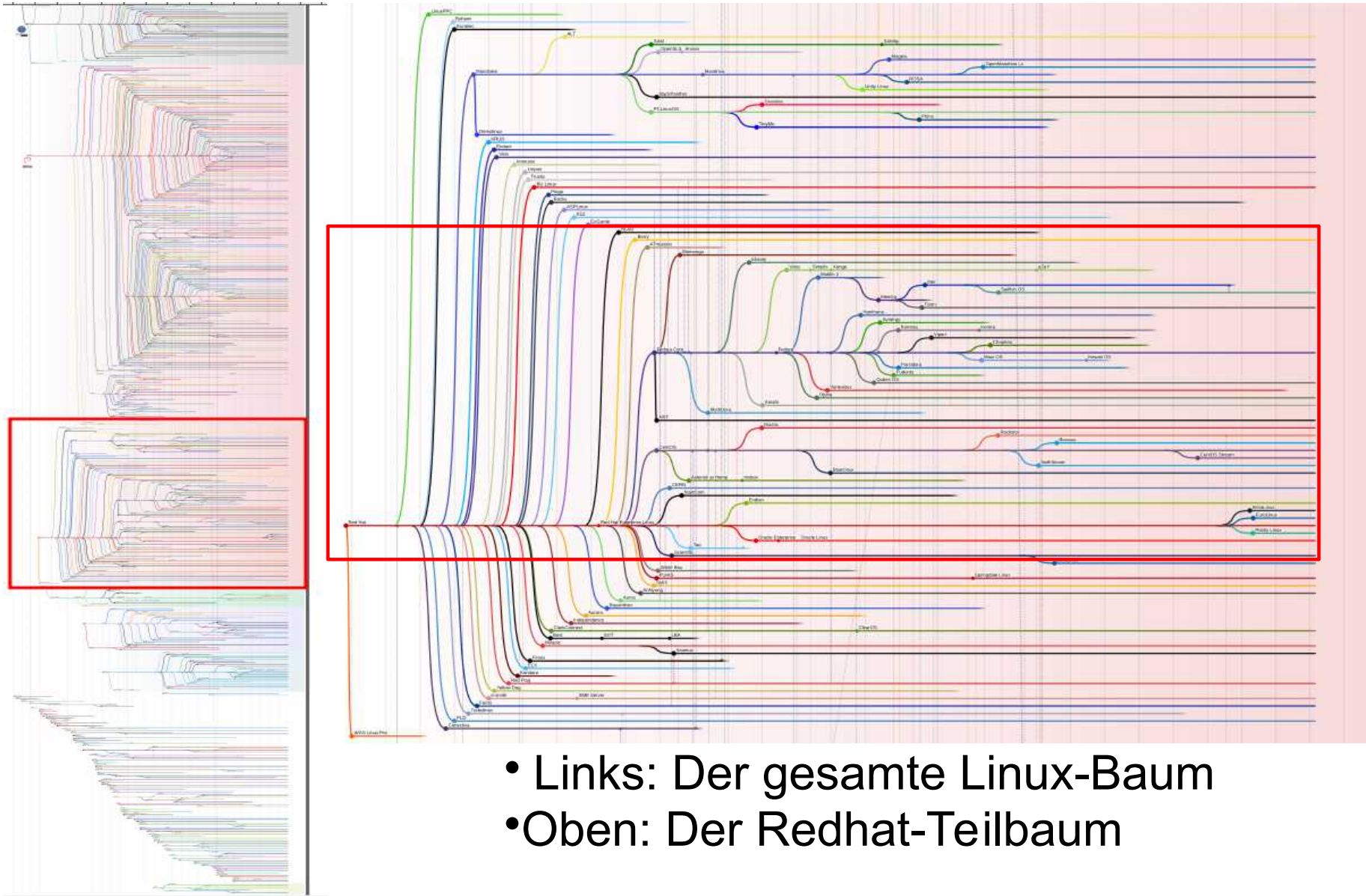
www.maemo.org

Vorgänger von MeeGo: 2. Moblin

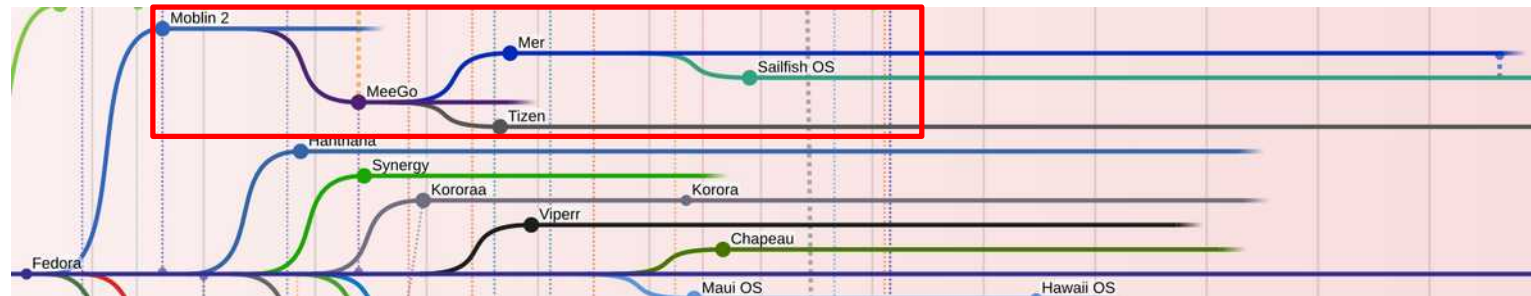
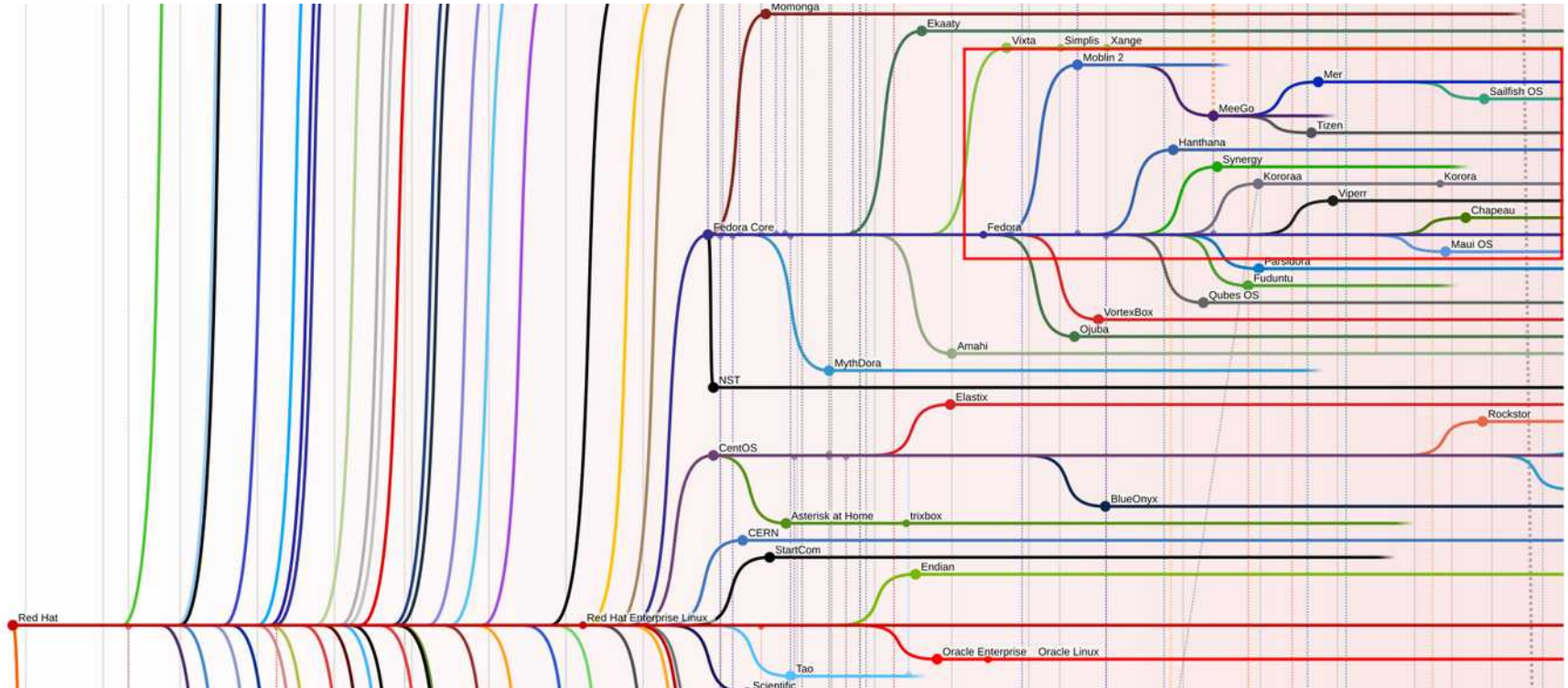
- Wikipedia: *Moblin war ein Open-Source-Projekt, das sich die Entwicklung einer speziell auf Mobile Internet Devices (MIDs) und andere Geräteklassen wie Netbooks und Nettops angepasste Linux-Distribution zum Ziel gesetzt hat. Intel startete die Internetseite im Juli 2007 und erweiterte die Seite signifikant im April 2008 mit der Veröffentlichung der Intel-Atom-Prozessorfamilie auf dem Intel Developer Forum in Shanghai. Die Intel-Atom-Prozessorfamilie zielt auf MIDs, Nettops und andere Marktsegmente, in denen geringer Stromverbrauch und geringe Kosten entscheidend sind. Auf der Moblin.org-Webseite findet sich ein Software Development Kit, der Quellcode für alle Hauptkomponenten der Moblin-Plattform sowie Dokumentation und Beispielcode.*
- Das Moblin-Projekt wurde eingestellt und zu MeeGo überführt.

Moblin	
Entwickler	Intel
Lizenz(en)	GPL, LGPL und andere
Akt. Version	2.1 vom 4. November 2009
Abstammung	GNU/Linux <ul style="list-style-type: none"> ↳ Red Hat Linux ↳ Fedora ↳ Moblin
Sonstiges	Aufgegeben zugunsten von MeeGo
moblin.org	

Sicht von oben (1): die Nadel im Heuhaufen

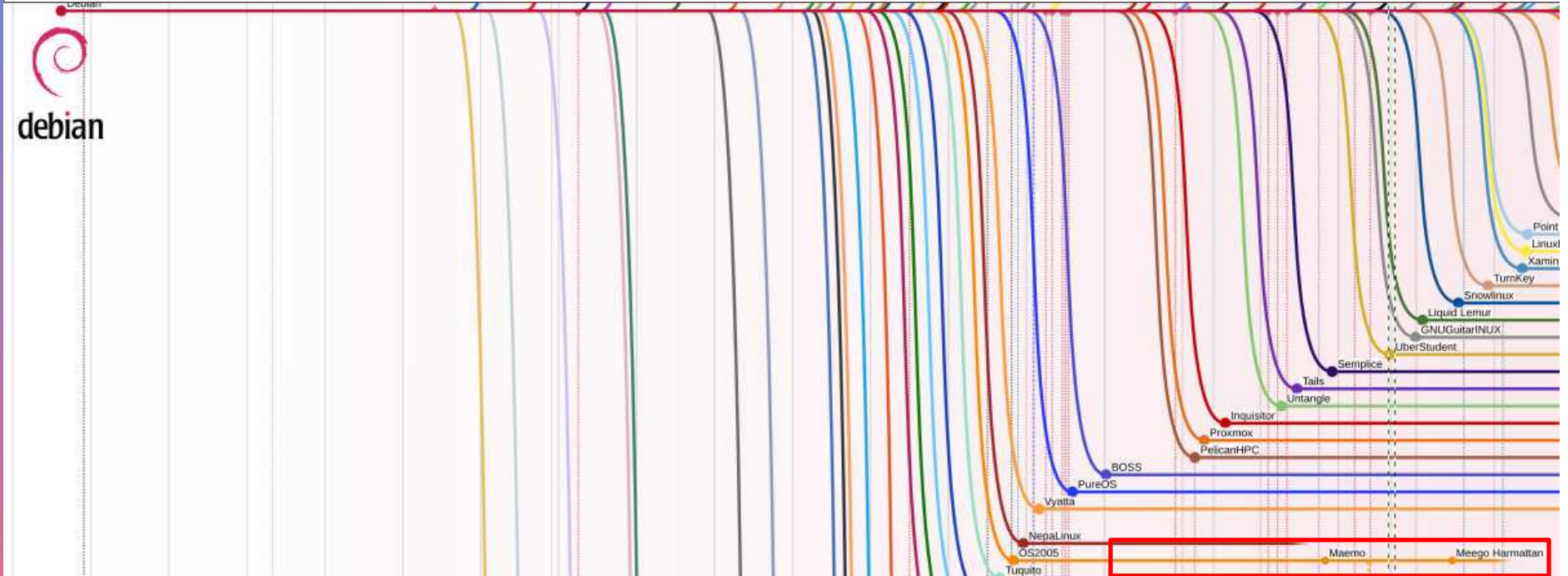


Sicht von oben (2): die Nadel im Heuhaufen



Oben: Position Fedora Teilbaum - Unten: Nadel gefunden

Maemo im Debian-Baum



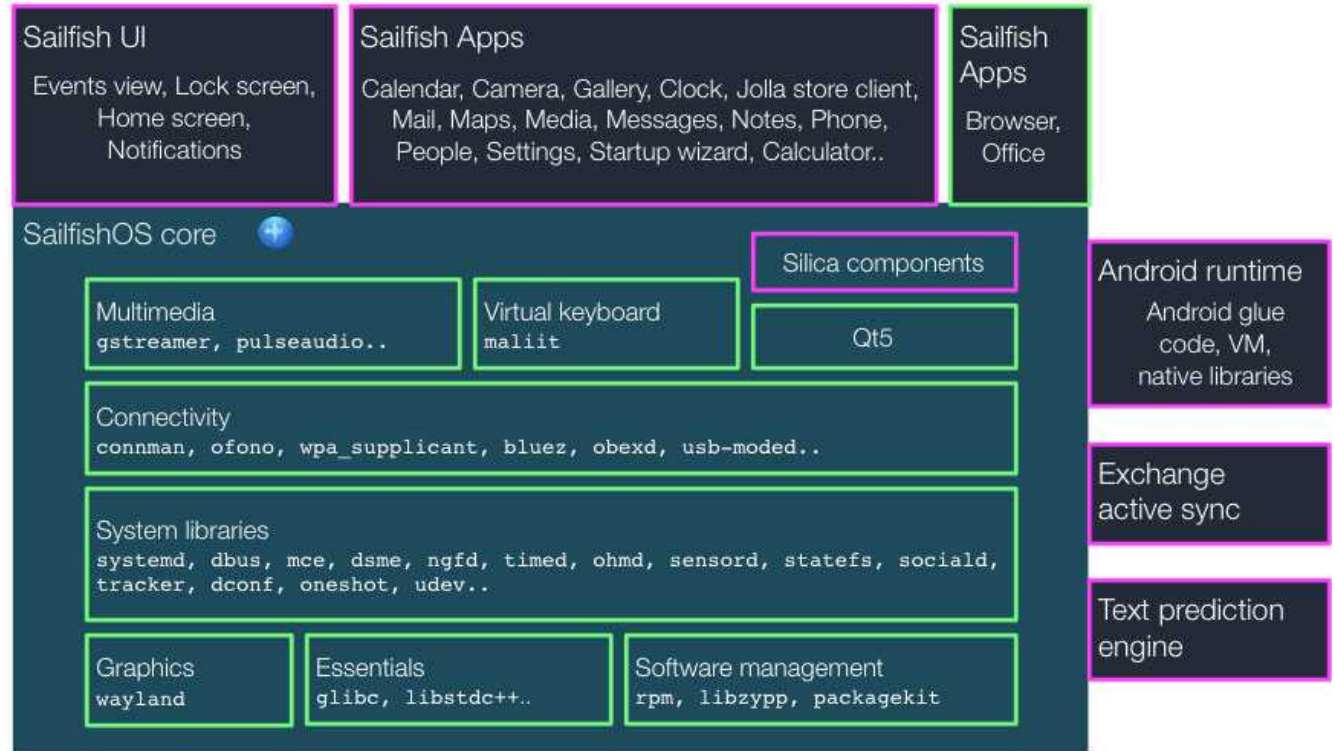
Jolla heute

- So stellen sie sich vor:
- <https://jolla.com/> - (sic: ohne „www“)
 - The deep tech OS company
 - We are Jolla, a deep tech company from Finland with a focus on innovative operating system solutions. Jolla was founded in 2011 continuing the heritage of Nokia and MeeGo.
 - Since then we have been developing Sailfish OS, the only European mobile operating system available in the market today, as well as AppSupport, which allows Android™ apps to run on any Linux platform. Since 2019 we have productized AppSupport specifically for the automotive sector.

Sailfish OS Werdegang

- Mit eigenem, separatem Webauftritt:
- <https://sailfishos.org/> - (auch hier: ohne „www“)
 - The mobile OS with built-in privacy
 - Sailfish OS is a **European** alternative to dominating mobile operating systems, and the only mobile OS offering an exclusive licensing model for local implementations.
 - Version History
 - 1.0 Nov. 2013
 - 2.0 Okt. 2015 (mit Jolla Tablet)
 - 3.0 Okt. 2018 (Sailfish X community, Sony Xperia)
 - 4.0.1 Feb. 2021 („new enablers to support different ecosystem projects“)

Architektur



Sailfish OS architecture designed to easily support multiple hardware targets

Hardware adaptation separate from the platform



Installation

- Download unter <https://shop.jolla.com/>
 - Die Version zum Download heißt Sailfish X
 - Es gibt sie kostenlos und kostenpflichtig, aber nur für „**Authorized Countries**“: EU, UK, Norwegen, Schweiz

What you get	Free	Xperia 10 II & III	Xperia 10, XA2	Xperia X	Gemini PDA
Sailfish OS to flash to your device	✓	✓	✓	✓	✓
Clear instructions and support for downloading & installation	✓	✓	✓	✓	✓
Software updates as long as device is supported		✓	✓	✓	✓
Customer support as long as device is supported		✓	✓	✓	✓
MS Exchange support		✓	✓	✓	✓
Predictive text input		✓	✓	✓	
Android AppSupport		11	11	4.4	
VoLTE Beta		✓			

Installation (1)

- Man nehme:
 - Unterstütztes Handy (z.B. Sony Xperia XA2 Ultra)
 - Verifiziere, dass funktionsfähig unter Android (SIM mit mobilen Daten, Kamera, Touch Display, Lautsprecher, Mikro, GPS)
 - USB-C Datenkabel
 - Internetverbindung
 - Linux Rechner
 - Download Image von Sailfish X.
 - `Sailfish_OS-Jolla-4.5.0.21-h3213-1.0.0.12.zip`

Installation (2)

- Vorgehensweise – Anleitungen etwas versteckt unter <https://jolla.com/sailfishxinstall/> (für Win, Lin, Mac):
 - 1) Sicherstellen, dass nicht Android 10 installiert ist (9 oder 8.1 sind OK)
 - 2) **Fastboot Tools** auf Linux installieren
 - Ubuntu: universe Repository
 - apt-get install android-tools-fastboot
 - 3) Extrahieren zip Datei
 - 4) Unlock Bootloader via **Unlock Code von Sony**. URL im Dokument stimmt nicht mehr: [aktuell](#)
<https://developer.sony.com/open-source/aosp-on-xperia-open-devices/get-started/unlock-bootloader>

Installation (3)

<https://developer.sony.com/open-source/aosp-on-xperia-open-devices/get-started/unlock-bootloader>

Open devices

Get started ▾

Guides ▾

Downloads ▾

More information ▾

Unlock your device

To unlock your device you need an **unlock code**, which you get by entering your device IMEI, IDID or MEID number in the generator below.

Note: Dual SIM devices have two IMEI numbers. Use IMEI 1 in order to get the correct unlock code for your device.

Select your device from the list below to get the **unlock code** to start unlocking it.

If you need assistance, please see [How to unlock bootloader](#).

Note: New devices XQ-CT62 (1IV US variant) and XQ-CQ62 (5IV US variant) do not support bootloader unlock.

How to unlock bootloader ›

Select your device:

Xperia XA2 Ultra



Installation (4)

[Open devices](#) [Get started](#) [Guides](#) [Downloads](#) [More information](#)

Connect to Fastboot ↑

1. Turn off your device.
2. Connect a USB-cable to your computer.
3. On your device, press the Fastboot key (**Volume up** for most devices, see [Useful key combinations](#)) at the same time as you connect the other end of the USB-cable.
 - **Windows users:** Open the **Devices and Printers** directory, right-click on the fastboot driver and press **Update**. Browse and point towards the location of the new `android_winbus.inf` file.
4. When your device is connected, open a command window on your computer and go to the **platform-tools** folder within the Android SDK folder.
5. Enter the following command:

```
fastboot devices
```

6. Verify that you get an answer back without any errors.

Enter unlock key ↑

WARNING! The command below contains your unlock key. If you perform this step, you will unlock the bootloader. This may void your warranty and/or any warranty from your operator.

Note: In this step you will need the [unlock code](#) for your device.

1. If you still want to unlock the bootloader of your device, enter the following command:

```
fastboot oem unlock 0x
```

Tatsächlich: `sudo fastboot -i 0x0fce oem unlock 0x`

2. Verify that you get an answer back without any errors.

Done! You have now unlocked the bootloader of your device.

Installation (5)

- Tatsächlich fehlen bei der Sony-Beschreibung einige Vorab-Schritte (Internet Verbindung erforderlich):
 - Aktiviere “**Developer options**” via “Settings > System > About Phone” – tippe mehrmals auf “Build Number” bis angezeigt wird “You are now a developer!”
 - Aktiviere “**OEM unlocking**” und “**USB debugging**” via “Settings > System > Developer options”
 - Telefon ausschalten
 - USB-Kabel am Computer anschließen (nur da)
 - Drücke und halte „Lauter“-Taste – so USB-Kabel am Telefon anschließen. LED muss blau werden.
 - Jetzt kann der **unlock**-Befehl (im Verzeichnis der entpackten Jolla-Datei) ausgeführt werden.

Installation (6)

- Als nächstes braucht man ein **Binary von Sony**, offenbar der proprietäre Anteil aus dem Diagramm, z.B.:
 - <https://developer.sony.com/file/download/software-binaries-for-aosp-oreo-android-8-1-kernel-4-4-nile-v16/> - [Link](#)
 - Entpacken ins gleiche Verzeichnis, wo schon Sailfish OS liegt

The screenshot shows a web browser window with the URL <https://developer.sony.com/file/download/software-binaries-for-aosp-oreo-android-8-1-kernel-4-4-nile-v16/>. The page header features the Sony logo, a 'Develop' dropdown menu, a language selector set to 'English', and a 'Log in' button. The main heading is 'Software binaries for AOSP Oreo (Android 8.1) – Kernel 4.4 – Nile (v16)'. Below this, the text reads: 'Software binaries for Xperia XA2, Xperia XA2 Ultra and Xperia XA2 Plus. To be used with the Android™ Open Source Project (AOSP) for Xperia devices. **SHA256SUM** 97120C2CB34CD5280FFA38A17E98A8F4BB93DA8140C8AF67AFCE7290FC6140AB.' The release date is 'OCTOBER 11, 2018 | 88.11 MB' and the filename is 'SW_BINARIES_FOR_XPERIA_ANDROID_8.1.6.4_R1_V16_NILE.ZIP'. A blue 'Download' button is located at the bottom left of the page content.

Installation (7)

- Jetzt kann (endlich) geflasht werden.
 - Gleiche Prozedur wie beim Unlock (15 Sekunden auslassen), wieder muss LED blau werden.



-
-
- **flash**-Befehl (im Verzeichnis der entpackten Jolla-Datei): `sudo bash ./flash.sh`
- Den Anweisungen folgen.
- Am Ende sollte kommen

- `Flashing completed.`
- Remove the USB cable and boot up the device by pressing the power key.



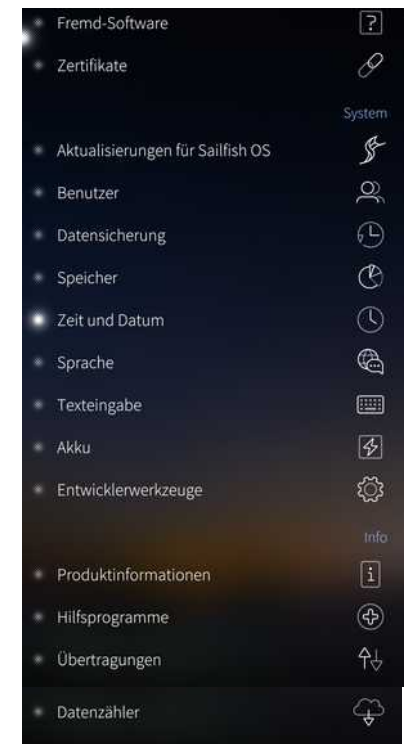
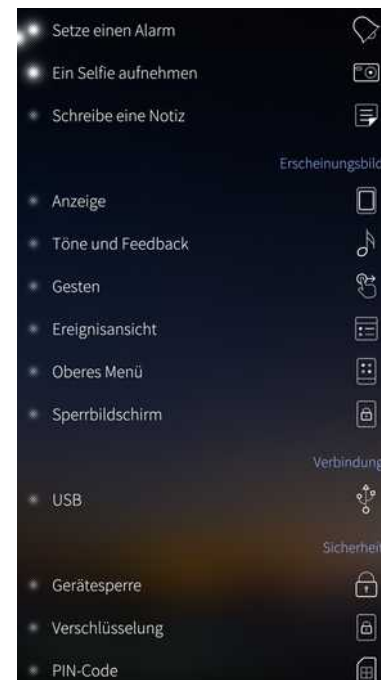
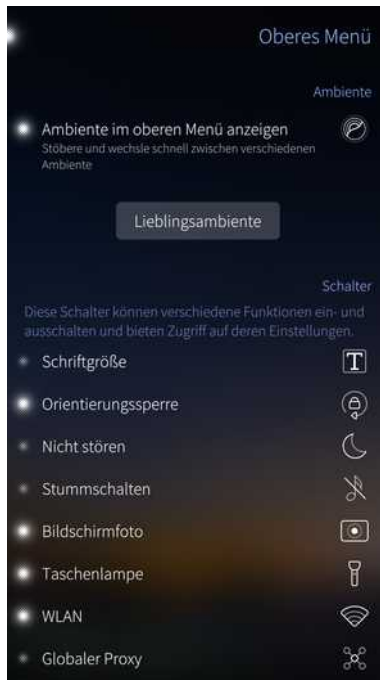
- Nach Reboot startet Salifish OS.

Anmerkungen

- Es gibt einen Weg zurück zu Android mit dem „Emma“ Flashing Tool.
- Die kostenlose Version ohne Android Support gilt als „Trial Version“. Die Kaufversion kostet 49,90 €.
- Da offenbar der Hardwarehersteller mitspielen muss, ist die Auswahl sehr beschränkt. Eigene Hardware von Jolla ist krachend gescheitert. Das Tablet wurde z.B. in einer dreistelligen Stückzahl verkauft.
- Außer den 5 Sony-Modellen wird noch ein [Gemini PDA](#) von Planet Computers aus 2018 auf der Website gelistet.

Oberes Menü

- Vom Rand nach unten wischen
- Besonderheiten:
 - Ambiente
 - Screenshot Icon
 - Sehr viele Optionen



SSH

- Man kommt nur rein, wenn das Handy entsperrt ist.
- SCP entsprechend, aber auch SCP-Server auf dem Handy via HTTP: <http://192.168.1.228:9537/>
 - Statt `sudo` heißt es hier übrigens `devel-su` für Rootrechte.
 - Screenshot nächste Seite

You must unlock the device lock and the encryption of your Sailfish phone before you can make an SSH connection to the phone. In other words, the phone must be on and unlocked. See these articles: [device lock](#) and [encryption](#).

```
guenter@guenter-Mint-T470:~$ ssh defaultuser@192.168.1.228
defaultuser@192.168.1.228's password:
Last login: Tue Jan 23 14:31:01 2024
,---
| Sailfish OS 4.5.0.21 (Struven ketju)
'---

[defaultuser@XperiaXA2Ultra ~]$ ls
Desktop      Downloads  Pictures   Public     Videos
Documents    Music      Playlists  Templates

[defaultuser@XperiaXA2Ultra ~]$ pwd
/home/defaultuser

[defaultuser@XperiaXA2Ultra ~]$ ls Pictures/
Camera/      Default/   QRclip/    Screenshots/  screenshots/
```



PC-Treff-BB Aidingen

Sailfish OS

© 2024 Günter Waller



192.168.1.228:9537

Directory listing for /

Durchsuchen...

Keine Datei ausgewählt.

upload

RootPage

HomePage

PreviousPage

.config/	4.0KB	2023-07-10 13:00:57
.fs-resized	0.0bytes	1975-10-12 09:23:29
afs/	4.0KB	2022-11-06 09:36:52
bin/	200.0bytes	2024-01-23 14:54:44
boot/	40.0bytes	2024-01-23 14:31:15
bt_firmware/	16.0KB	1970-01-01 01:00:00
bugreports@	50.0bytes	2023-02-13 17:17:35
cache@	11.0bytes	2023-02-13 17:17:35
config/	0.0bytes	1970-01-01 01:00:01
d@	17.0bytes	2023-02-13 17:17:35
data/	4.0KB	2023-07-10 13:00:53
default.prop@	23.0bytes	2023-02-13 17:17:35
dev/	7.8KB	2024-01-23 14:30:24
documentation.list	0.0bytes	2022-11-06 09:38:16
dsp/	4.0KB	1970-01-01 01:00:00
etc/	460.0bytes	2024-01-23 14:54:44
fimage/	4.0KB	2023-07-11 11:30:38
firmware/	16.0KB	1970-01-01 01:00:00

Demotime – aber vorher...

- Für Demos wäre ein Tool wie `scrcpy` (für Android) ganz gut. Das spielt aber nur mit Android zusammen via ADB (Android Debug Bridge).
- Gefunden habe ich **Screencast**, und dabei gelernt, dass es (neben dem Jolla App Store) ein **chum repository** gibt – außerdem die Fundstelle **openrepos.net**.

<https://openrepos.net/content/nephros/screencast-now-chum>

User login

Username or e-mail *

Password *

- [Create new account](#)
- [Request new password](#)

Log in

Categories

- [Coding Competition](#)
- ▼ [Applications](#)
 - [Application](#)
 - [Adult Content](#)
 - [Ambience & Themes](#)

Screencast - now in chum

Submitted by [nephros](#) on Thu, 2021/06/17 - 17:12



Rating:



Your rating: None Average: 5 (1 vote)

This is now available from the chum repository. Please update to that version.

Taking over from Maemo/Meego/SailfishOS/AuroraOS legend [coderus](#), this is the maintenance version of his Screencast application.

Please consider these versions experimental until I get to grips with how everything works.

- Original software: <https://openrepos.net/content/coderus/screencast>
- Source for this version: <https://gitlab.com/nephros/coderus-screencast/>

Weg zu Screencast (1)

- Also auf zu **SailfishOS:chum** community repository:
- <https://github.com/sailfishos-chum/main>
- Hat den Anspruch, primäre SW-Plattform für Sailfish zu werden.
- Verteilt RPM Pakete (keine Binaries) via OBS (Open Build Service).
- Eigentlicher Link:
<https://build.sailfishos.org/project/show/sailfishos:chum>
- Kann benutzt werden entweder
 - mit der App **SailfishOS:Chum GUI application**
 - oder via CLI package management, **pkcon** oder **zypper**

Weg zu Screencast (2)

- Ich entscheide mich für die App – nicht wirklich der einfachere Weg, denn jetzt brauche ich den **SailfishOS:Chum GUI Installer**:

<https://github.com/sailfishos-chum/main>

☰ README 📄 MIT license

User's guide

There are two different ways of using the SailfishOS:Chum repository:

- with the [SailfishOS:Chum GUI application](#)
- with the usual command line tools for package management, `pkcon` or `zypper`

How to install the SailfishOS:Chum GUI application

The easiest way to install the SailfishOS:Chum GUI application built for the CPU-architecture of a device and its installed Sailfish OS release fully automatically is [the SailfishOS:Chum GUI Installer](#). Because it is [hosted at OpenRepos](#), it can be conveniently [installed via Storeman](#). The SailfishOS:Chum GUI Installer can also be manually downloaded from [OpenRepos](#), [GitHub](#) or [SailfishOS:Chum](#) and then installed, e.g., by `pkcon install-local <local path of package>`.

Alternatively a version of the SailfishOS:Chum GUI application for a specific CPU-architecture and Sailfish OS release can be manually selected and downloaded at [chumrpm.netlify.app](#) for manual installation.

Furthermore SailfishOS:Chum GUI application's individual RPMs are also provided at the [SailfishOS:Chum repository](#), where they can be fully manually selected and fetched from, as the SailfishOS:Chum GUI Installer does in a fully automated manner and the web-interface at [chumrpm.netlify.app](#) semi-automated.

Important note: If you experience issues while or after installing the SailfishOS:Chum GUI application, [do read the installation notes!](#)

Weg zu Screencast (3)

- Nun also **Storeman**, der Installer für den Installer, bzw. **Storeman Installer**, der Installer für den Installer für den Installer:

<https://openrepos.net/content/olf/storeman-installer>



The screenshot shows the OpenRepos website interface. At the top, there is a search bar and navigation links: About OpenRepos, News, Publishers, Last Added, Global Stats, Donate, and Feedback. The main content area is titled "Home" and features a "User login" form with fields for "Username or e-mail" and "Password", along with links for "Create new account" and "Request new password". A "Log in" button is also present. To the right, the "Storeman Installer" page is displayed, showing it was submitted by "olf" on Sun, 2023/07/23 - 16:17. The page includes a rating section with five stars and the text "Your rating: None Average: 5 (11 votes)". Below the rating, the title "Storeman Installer" is followed by a description: "The Storeman Installer for SailfishOS performs the initial installation of the **Storeman OpenRepos client application**. Storeman Installer selects, downloads and installs the correct variant of the Storeman application built for the CPU-architecture of the device and the installed SailfishOS release from the SailfishOS-OBS."

Weg zu Screencast (4)

- Nun also doch ein rpm installieren:
 - harbour-storeman-installer-2.2.4-release5.noarch.rpm
 - **„Allow untrusted software“ !!**
 - Man muss nach dem Download mit einem Filemanager die Stelle finden, wohin es geladen wurde. Bei mir die SD-Karte, war schwierig.

Installation instructions

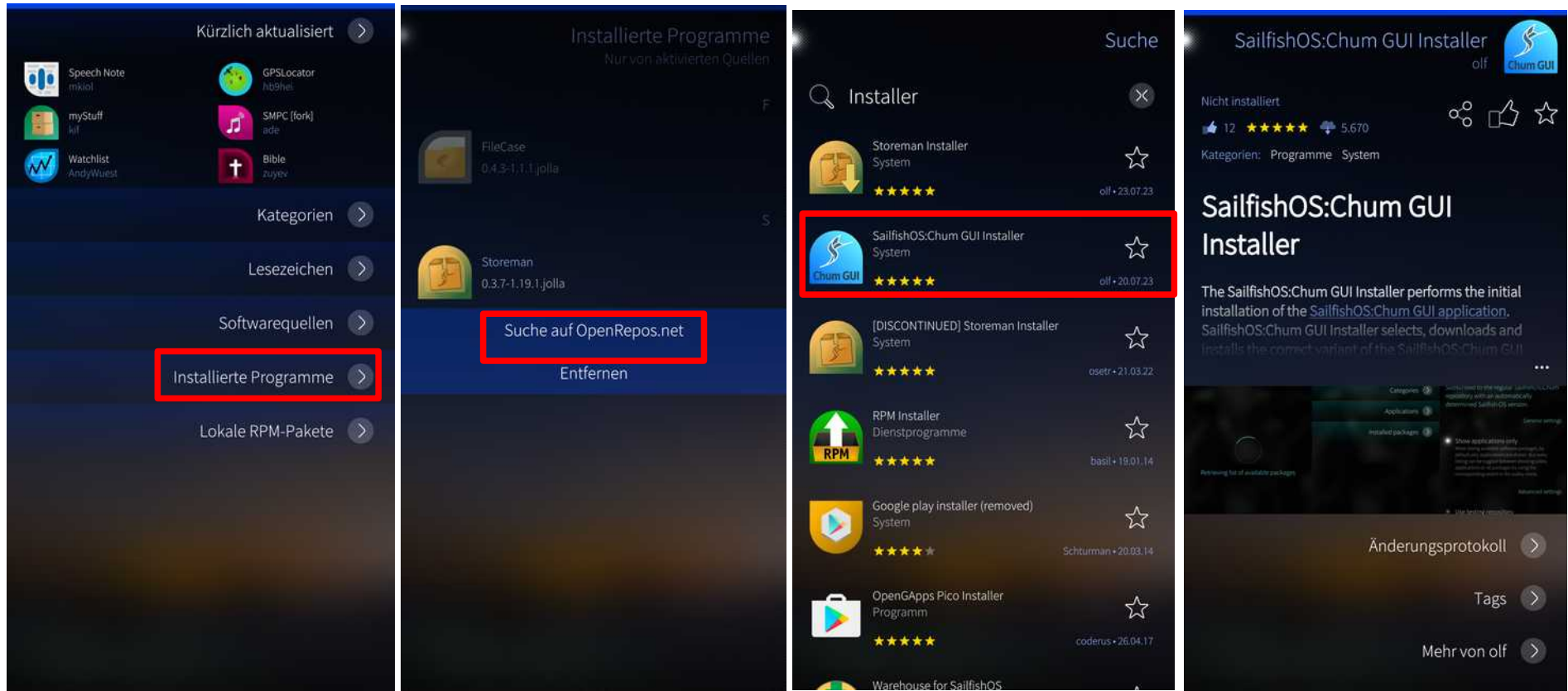
- Initial installation without having Storeman already installed
 1. Enable "System → Security → Untrusted software → Allow untrusted software" in the SailfishOS Settings app.
 2. Download the current Storeman Installer RPM from [OpenRepos.net](https://openrepos.net) (i.e., here, below) or its "latest release" page at [GitHub](https://github.com).
 3. Tap on the "File downloaded" notification on your SailfishOS device or select the downloaded RPM file in a file-manager app and choose "Install" in its pulley menu; then confirm the installation.
 4. Preferably disable "Allow untrusted software" again.

Weg zu Screencast (5)

- Nun ist Storeman ein startbares Icon. Damit kann jetzt der SailfishOS:Chum GUI Installer installiert werden.

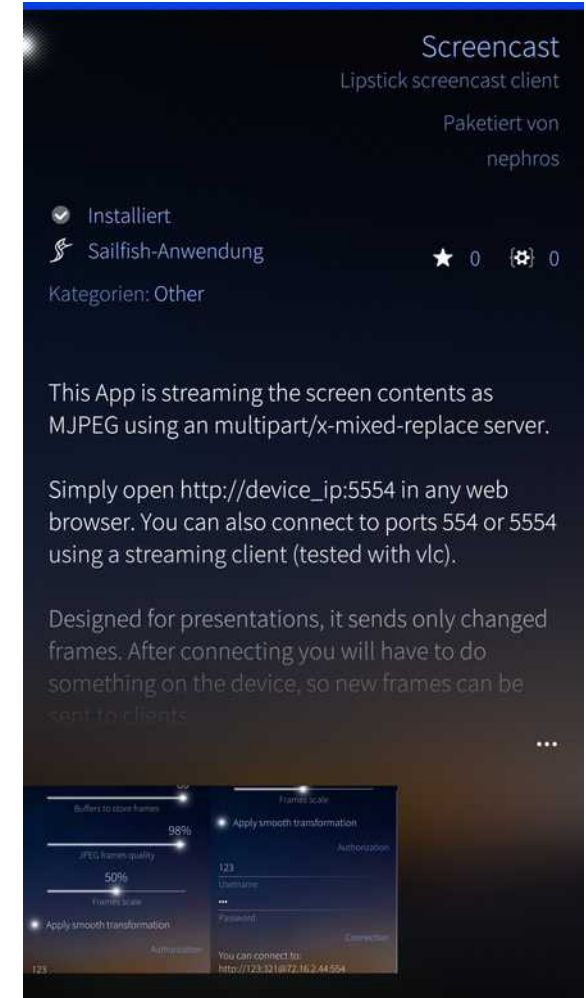
- Installation via Storeman

- Search for *Installer*.
- Select the *SailfishOS:Chum GUI Installer* by *olf*.
- Enable *olf*'s repository in the top pulley menu.
- Install *SailfishOS:Chum GUI Installer*.



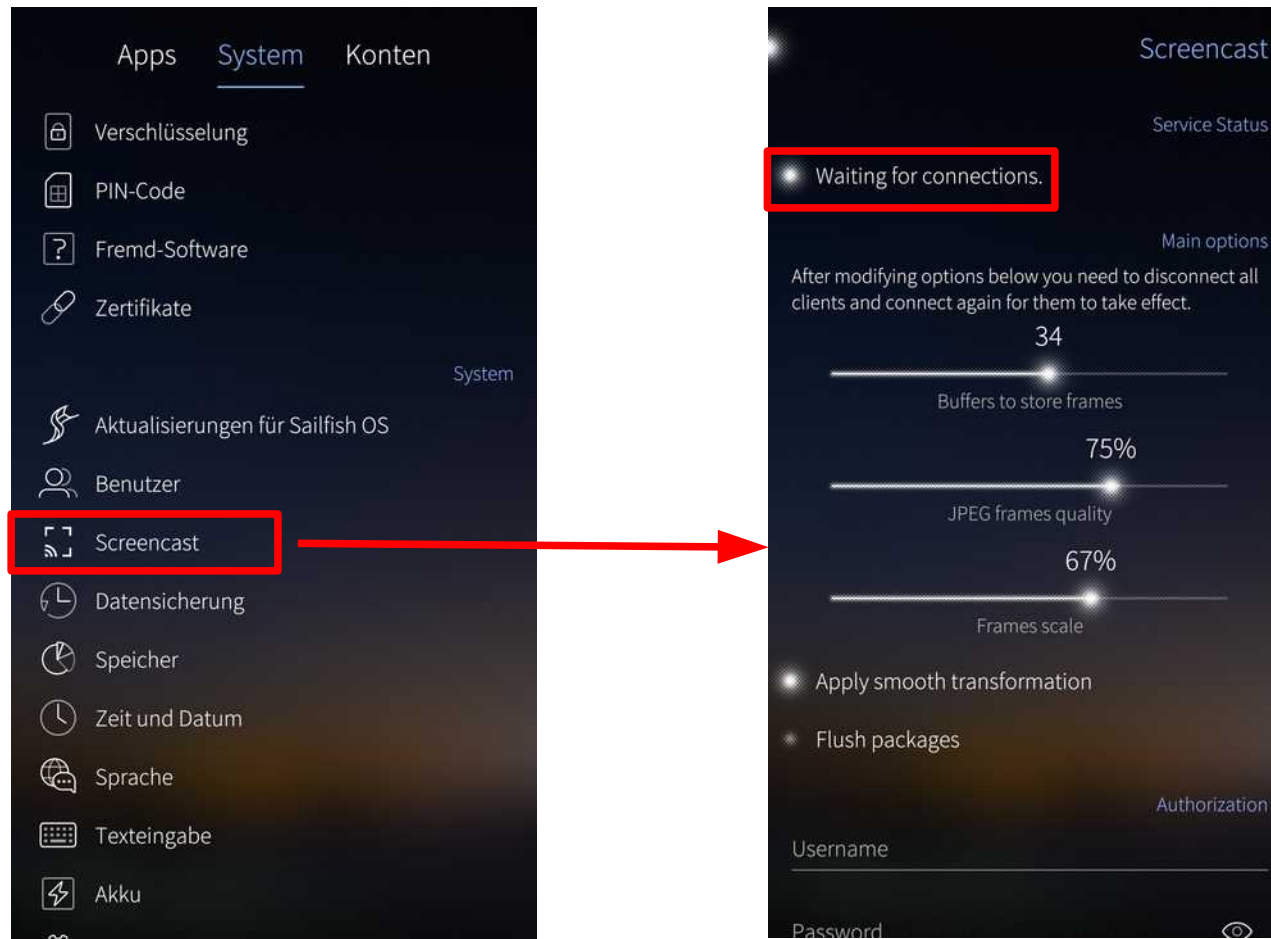
Weg zu Screencast (6)

- Der SailfishOS:Chum GUI Installer führt nun tatsächlich, wenn man ihn startet, vollautomatisch die Installation des SailfishOS:Chum GUI Repositories nebst der zugehörigen App aus.
- Damit kann man dann endlich Screencast installieren.
- Fazit: *Die spinnen, die Römer.*



Screencast

- Wer nun glaubt, eine weitere clickbare App vorzufinden, sieht sich getäuscht. Es gibt nur einen neuen Eintrag in den Systemeinstellungen. Port ist 5554, Handy muss entsperrt sein. Kann mit Uid/Pw geschützt werden.



Letzte Anmerkungen vor den Demos

- Nachträgliche Überlegung zu den 3 Repositories:
 - Jolla Store – der offizielle App Store vom Hersteller
 - Chum – die um Sailfish OS herum entstandene Community und ihre Apps für das OS
 - Storeman – freie Wildbahn, Apps für alle möglichen Plattformen (mit allen möglichen Dependencies)

Erkenntnisse, was hat nicht geklappt

- Der Standarduser heißt `defaultuser`
- Unterstützung für Samba/CIFS

– Sollte eigentlich gehen ... aber

- `[defaultuser@XperiaXA2Ultra ~]$ devel-su mount -t cifs -o username=guenter,vers`
- `=2.0 '//192.168.1.1/nas3' /mnt/nas3 --verbose`
- Password:
- Password for guenter@//192.168.1.1/nas3: `*****`
- `mount.cifs kernel mount options: ip=192.168.1.1,unc=\\192.168.1.1\nas3,vers=2.0,user=guenter,pass=*****`
- `mount error(22): Invalid argument`
- Refer to the `mount.cifs(8)` manual page (e.g. `man mount.cifs`)
- `[defaultuser@XperiaXA2Ultra ~]$`

Erkenntnisse

- Manches geht gut, manches zäh (Handy ist einige Jahre alt).
- Vieles ist sehr spartanisch.
- Andererseits ist manches Standard im OS, was bei Android Erweiterungen benötigt (z.B. CalDAV).
- Es gibt einen Gastzugang.

Demo

- Tutorial
- App Stores
- Browser
- Kamera
- Kontakte
- Galerie
- Uhr
- Bücher (ePub)
- Kalender
- Mail
- GSM/Map
- Podcasts
- Mediathek
- File Manager
- Newsmedien
- ARD Audiothek
- ... Wünsche

Links

- <https://jolla.com/>
 - <https://jolla.com/appsupport/>
 - <https://sailfishos.org/>
 - <https://blog.jolla.com/>
 - <https://forum.sailfishos.org/>
- <https://openrepos.net/>
- https://de.wikipedia.org/wiki/Jolla#Sailfish_OS
- https://en.wikipedia.org/wiki/Sailfish_OS