

## ArchivistaVM Mini: 100 MByte ISO-Datei\*, 700 MByte RAM und 20 Sekunden bis zum Produktivbetrieb

**Pfaffhausen, 17. September 2012:** Vor etwa **einem Jahr wurde die erste RAM basierte ArchivistaVM-Lösung** veröffentlicht. In den letzten 12 Monaten konnte die ISO-Datei von ArchivistaVM Mini von ca. 330 MByte auf unter 100 MByte reduziert, der Speicherbedarf von 2 GByte auf nunmehr unter 700 MByte minimiert, und auch die Startzeit deutlich nach unten korrigiert werden (derzeit noch ca. 20 Sekunden). In diesem Blog werden die Hintergründe dazu erläutert. Eines vorweg, eine schlanke Virtualisierungslösung im RAM ist keine Hexerei, sondern beruht vielmehr daher, dass ArchivistaVM konsequent auf moderne Standardkomponenten setzt.

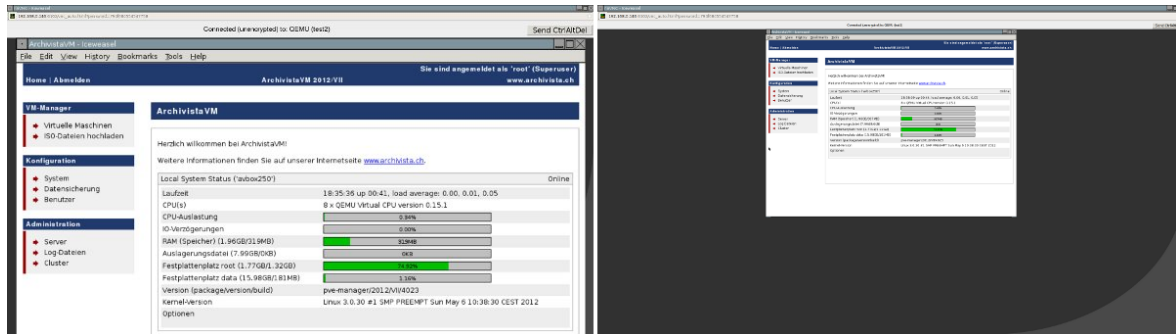


### HTML5-Browser und noVNC halten Einzug

Kernstück der neuen ArchivistaVM Mini bildet ein aktualisierter Browser (Icewasel alias Firefox). Dank HTML5-Unterstützung in ArchivistaVM konnte **noVNC** erst integriert werden. Warum ist dies zentral? Bisher war es bei der Virtualisierung mit ArchivistaVM notwendig, dass ein VNC-Java-Applet auf dem Browser installiert sein musste, um die virtualisierten Gäste darstellen zu können. Die dazugehörige Technologie hat mittlerweile viele Jahre auf dem Buckel. Ganz allgemein kann gesagt werden, Applets haben sich (vielleicht mit der Ausnahme von Flash) nicht wirklich durchgesetzt. HTML5-fähige Browser bieten eine sogenannte Canvas-Funktion an. Mit dieser kann nun ein VNC-Viewer direkt im Browser dargestellt werden — ohne dass dazu zusätzliche Applets installiert sein müssen.

Alle bekannteren **Browser (Firefox, Internet Explorer, Chrome, Opera und Safari)** bieten seit mehreren Jahren das **Canvas-Feature** an. Im Falle der ArchivistaBox bedeutete dies, dass der Browser zwar einige MByte mehr an Platz im RAM benötigt, dass dafür aber **auf Java verzichtet werden konnte, womit in etwa 110 MByte auf der RAM-Disk gespart** werden konnten.

Die moderate Laufgeschwindigkeit von Java ist allseits bekannt, jene von JavaScript wird laufend optimiert. Und daher lassen sich mit JavaScript heute Dinge realisieren, die noch vor ein zwei Jahren undenkbar gewesen wären. Gerne sei an dieser Stelle auf den **Linux-PC-Emulator** verwiesen (inkl. virtuellem C-Compiler), der komplett in JavaScript realisiert wurde.



Von daher erstaunt es nicht mehr weiter, wenn sich heute auch anspruchsvollere Dinge wie das Übertragen ganzer virtualisierter Bildschirme (letztlich überträgt ein VNC-Vierwer genau dies) in sehr guter Geschwindigkeit realisieren lassen. Die **Vorteile** liegen auf der Hand, es müssen **keine Applets mehr** vorgehalten werden, ArchivistaVM Mini kann ca. **110 MByte an "Ballast abwerfen"** und weiter kann der dargestellte virtuelle **Bildschirm im Browser beliebig skaliert** werden (Ctrl+Scrollrad, linke Abbildung vergrößerte Ansicht, rechte Abbildung verkleinerte Darstellung).

Für die Umsetzung des VNC-Clients im Browser kommt wie bereits erwähnt **noVNC** zum Einsatz, weil die Anforderungen bei noVNC im Vergleich zu anderen Lösungen (**z.B. Guacamole**) sehr bescheiden sind. Ebenfalls in Betracht gezogen wurde der Einsatz von **Spice**. Weil Spice (so schnell es auch sein mag) aber nur mit Applets machbar ist, und gerade diese sollten ja vermieden werden, wurde aktuell auf Spice verzichtet. Oder anders herum gesagt, Spice ist zwar schnell, aber derzeit leider keineswegs ein Standard.

### **Darum ist die Virtualisierung mit ArchivistaVM für kluge Köpfe erste und beste Wahl**

Etwas ketzerisch gefragt: Welche andere Virtualisierungsprodukt kommt mit einer ISO mit unter 100 MByte daher (OS, Server, Client zusammen), welche andere Lösung lässt sich standalone in ca. 20 Sekunden aufsetzen, gibt es überhaupt ein Produkt, das sich komplett im RAM betreiben lässt? Oder kennen Sie eine Open Source Lösung, bei der Cluster (inkl. DRBD) ohne manuelle Handarbeit aufgesetzt werden können? Welche andere Lösung lässt sich sowohl beim Hochfahren wie auch im Betrieb automatisieren? Sie werden viele Produkte für die Virtualisierung vorfinden, aber einzig ArchivistaVM wird all diese Features erfüllen.

Open Source und Linux bei ArchivistaVM bedeuten ferner nicht, dass Linux-Kenntnisse notwendig sind, **CD/Stick einschieben und los geht es.**

**Installation und Updates gehören dank dem einzigartigen RAM-Modus der Vergangenheit** an. ArchivistaVM ist weiter die einzige Plattform für die Virtualisierung, bei der die Gäste sowohl auf dem lokalen Server als auch über Intra- und Internet verwaltet werden können, und dies zu 100% ohne zusätzliche Client-Software. **ArchivistaVM läuft in jedem HTML5-fähigen Browser, weitere Plugins (Applets) sind keine erforderlich.** ArchivistaVM lässt sich zudem sowohl beim Hochfahren des Systems als auch bei der Administration automatisieren.

Aus all diesen Gründen wählen kluge Köpfe ArchivistaVM als Plattform zur Virtualisierung. Noch klügere Köpfe erwerben für den harten Produktiv-Alltag entweder **ArchivistaVM Light** oder eine **ArchivistaBox-Hardware**. Die klügsten Köpfe aber sind jene, welche die ArchivistaBox schon im Einsatz haben. Einmal weil sie schon seit Jahren von den tollen Features profitieren, die es nur bei ArchivistaVM gibt, und weiter, weil es ohne sie ArchivistaVM gar nicht gäbe. Falls Sie sich noch nicht zu den klugen bzw. klügsten Köpfen zählen können, so wäre es jetzt äusserst klug, die **ISO-Datei zu beziehen**. Viel Spass!

**\* Update: Für alle jene, welche es gerne noch etwas kleiner hätten, gibt es die ArchivistaVM Mini auch ohne GUI, diese ISO-Datei ist noch 72 MByte gross. Dabei wird nochmals deutlich weniger RAM benötigt (<512 MByte). Mehr dazu findet sich [hier](#). Wer die ArchivistaCDs (ab 2012-10-28) über **PXE booten möchte, findet hier Informationen.****



Facebook



In 20 Sekunden zum VM-Server

Twitter