

H. Neuroth, A. Oßwald, R. Scheffel, S. Strathmann, K. Huth (Hrsg.)

# nestor Handbuch

Eine kleine Enzyklopädie  
der digitalen Langzeitarchivierung

Version 2.3

Kapitel 8.4

Emulation

nestor Handbuch: Eine kleine Enzyklopädie der digitalen Langzeitarchivierung  
hg. v. H. Neuroth, A. Oßwald, R. Scheffel, S. Strathmann, K. Huth  
im Rahmen des Projektes: nestor – Kompetenznetzwerk Langzeitarchivierung und  
Langzeitverfügbarkeit digitaler Ressourcen für Deutschland  
nestor – Network of Expertise in Long-Term Storage of Digital Resources  
<http://www.langzeitarchivierung.de/>

Kontakt: [editors@langzeitarchivierung.de](mailto:editors@langzeitarchivierung.de)  
c/o Niedersächsische Staats- und Universitätsbibliothek Göttingen,  
Dr. Heike Neuroth, Forschung und Entwicklung, Papendiek 14, 37073 Göttingen

Bibliografische Information der Deutschen Nationalbibliothek  
Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen  
Nationalbibliografie; detaillierte bibliografische Daten sind im Internet unter  
<http://www.d-nb.de/> abrufbar.

Neben der Online Version 2.3 ist eine Printversion 2.0 beim Verlag Werner Hülsbusch,  
Boizenburg erschienen.

Die digitale Version 2.3 steht unter folgender Creative-Commons-Lizenz:  
„Namensnennung-Keine kommerzielle Nutzung-Weitergabe unter gleichen Bedingungen 3.0  
Deutschland“  
<http://creativecommons.org/licenses/by-nc-sa/3.0/de/>



Markenerklärung: Die in diesem Werk wiedergegebenen Gebrauchsnamen, Handelsnamen,  
Warenzeichen usw. können auch ohne besondere Kennzeichnung geschützte Marken sein und  
als solche den gesetzlichen Bestimmungen unterliegen.

URL für Kapitel 8.4 „Emulation“ (Version 2.3): [urn:nbn:de:0008-20100305134](http://nbn-resolving.de/urn/resolver.pl?urn=urn:nbn:de:0008-20100305134)  
<http://nbn-resolving.de/urn/resolver.pl?urn=urn:nbn:de:0008-20100305134>



*Gewidmet der Erinnerung an Hans Liegmann (†), der als Mitinitiator und früherer  
Herausgeber des Handbuchs ganz wesentlich an dessen Entstehung beteiligt war.*

## 8.4 Emulation

*Stefan E. Funk*

Mit Emulation (Nachbildung, Nachahmung, von lat. aemulator = Nacheiferer) versucht man die auftretenden Verluste einer Datenformatmigration zu umgehen, indem man die originale Umgebung der archivierten digitalen Objekte nachbildet. Emulation kann auf verschiedenen Ebenen stattfinden:

- zum einen auf der Ebene von Anwendungssoftware,
- zum anderen auf der Ebene von Betriebssystemen und zu guter Letzt
- auf der Ebene von Hardware-Plattformen.

So kann zum Beispiel die originale Hardware des digitalen Objekts als Software mit einem Programm nachgebildet werden, welches das archivierte Betriebssystem und die darauf aufbauenden Softwarekomponenten laden kann (Emulation von Hardware-Plattformen). Ein Beispiel für die Emulation von Betriebssystemen wäre ein MS-DOS-Emulator<sup>9</sup>, der die Programme für dieses schon etwas ältere Betriebssystem auf aktuellen Rechnern ausführen kann. Ein Beispiel für den ersten Fall wäre etwa ein Programm zum Anzeigen und Bearbeiten von sehr alten Microsoft Word-Dateien (.doc), die das aktuelle Word nicht mehr lesen kann. Auf diese Weise wird die Funktionalität dieser alten und nicht mehr verfügbaren Soft- oder Hardware emuliert und die Inhalte bzw. die Funktionalität der damit erstellten Dokumente erhalten. Im Gegensatz zur Migration, bei der jeweils eine neue und aktuellere Version des digitalen Objektes selbst erzeugt wird, werden die originalen Objekte bei der Emulation nicht verändert. Stattdessen muss man für jede neue Hardwarearchitektur die Emulationssoftware anpassen, im schlechtesten Fall muss diese jedes Mal neu entwickelt werden. Wenn das aus irgendeinem Grund nicht geschieht, ist der komplette Datenbestand der betroffenen Objekte unter Umständen nicht mehr nutzbar und damit für die Nachwelt verloren.

### Emulation von Anwendungssoftware

Da es um die Darstellung der digitalen Dokumente geht, die wir vorhin beschrieben haben, ist die Emulation der Software, die mit diesen Dokumenten arbeitet, eine erste Möglichkeit. So kann auf einem aktuellen System ein Pro-

---

<sup>9</sup> DOS – Disc Operating System, näheres unter: [http://www.operating-system.org/betriebssystem/\\_german/bs-msdos.htm](http://www.operating-system.org/betriebssystem/_german/bs-msdos.htm)

gramm entwickelt werden, das archivierte digitale Objekte in einem bestimmten Format öffnen, anzeigen oder bearbeiten kann, auf die mit aktueller Software auf diesem System nicht mehr zugegriffen werden kann, weil vielleicht die Originalsoftware nicht mehr existiert oder auf aktuellen Systemen nicht mehr lauffähig ist.

Wenn wir zum Beispiel eine PDF-Datei aus dem Jahr 1998, Version 1.2, darstellen möchten, und der aktuelle Acrobat Reader 7.0 stellt das Dokument nicht mehr richtig dar, müssen wir einen PDF-Reader für diese PDF-Version auf einem aktuellen Betriebssystem programmieren, sprich: einen alten PDF-Reader emulieren. Dieser sollte dann alle PDF-Dateien der Version 1.2 darstellen können. Für jeden Generationswechsel von Hardware oder Betriebssystem würde so ein PDF-Reader benötigt, um den Zugriff auf die PDF-Dokumente in Version 1.2 auch in Zukunft zu gewährleisten. Die genaue Kenntnis des PDF-Formats ist hierzu zwingend erforderlich.

## **Emulation von Betriebssystemen und Hardware-Plattformen**

Bei einigen Anwendungen kann es sinnvoll sein, eine komplette Hardware-Plattform zu emulieren, zum Beispiel wenn es kein einheitliches Format für bestimmte Anwendungen gibt. Hier ist der Commodore-64 ein gutes Beispiel. Die Spiele für den C-64 waren eigenständige Programme, die direkt auf dem Rechner liefen, soll heißen, es wird direkt die Hardware inklusive des Betriebssystems<sup>10</sup> benötigt und nicht ein Programm, das diese Spiele ausführt (wie ein PDF-Viewer).

Es muss also ein Commodore-64 in Software implementiert werden, der sich genau so verhält wie die Hardware und das Betriebssystem des originalen Commodore-64 und auf einem aktuellen Computersystem lauffähig ist. Diese C-64-Emulatoren gibt es für nahezu alle aktuellen Computersysteme und auch weitere Emulatoren für andere ältere Systeme sind erhältlich.<sup>11</sup>

Die Emulation eines Betriebssystems oder einer Hardware-Plattform ist eine sehr komplexe Sache, die schon für einen C-64-Emulator sehr viel Arbeit bedeutet. Man kann jedoch auch die Hardware eines PC in Software nachbilden, um dann auf einem solchen virtuellen PC beliebige Betriebssysteme und die auf ihnen laufenden Anwendungsprogramme oder auch Spiele zu starten (die Be-

---

10 Eine Trennung von Hardware und Betriebssystem ist beim Commodore-64 nicht nötig, da diese beiden Komponenten sehr eng zusammenhängen. Auch andere „Betriebssysteme“ des C-64, wie zum Beispiel GEOS, setzen direkt auf das Betriebssystem des C-64 auf.

11 Hier einige Adressen im Internet zum Thema Emulatoren: <http://www.aep-emu.de/>, <http://www.homecomputermuseum.de/>

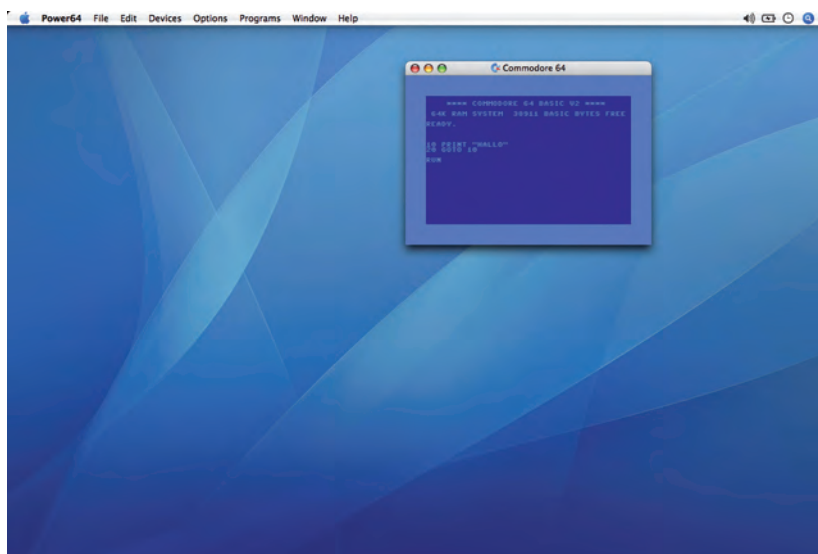


Abbildung 1: Power 64, ein Commodore-64 Emulator für Mac OS X

triebssysteme wie auch die Programme bleiben dann im Originalzustand). Dies bedeutet im Allgemeinen, dass eine gute Performanz auf der aktuellen Hardware vorhanden sein muss. Eine Emulation eines Commodore-64 auf einem aktuellen PC ist jedoch keine performanzkritische Anwendung. Für zukünftige Computersysteme, die unsere heutigen emulieren sollen, wird im Allgemeinen davon ausgegangen, dass deren Performanz weitaus höher ist als heute, sodass auch hier die Performanz für eine erfolgreiche Emulation ausreichen dürfte.

#### *Beispiel: Migration und Emulation alter C-64 Programme*

- Da der Commodore 64 ein sehr beliebter und weit verbreiteter Homecomputer war, gibt es sehr viele Emulatoren für nahezu alle aktuellen Computersysteme. Viele Videospiele, die es für den C-64 gab, sind im Internet als C-64 Disk-Image zu finden. Die darin enthaltenen Programme können dann mit den Emulatoren geladen und genutzt werden. Als alter C-64 Nutzer stand ich also nicht vor dem Problem, meine Spiele von alten 5,25-Zoll Disketten auf neuere Datenträger migrieren zu müssen. Ein Emulator für den Apple unter Mac OS X ist Power64<sup>12</sup>, siehe Abbildung 1.
- Anders sah es hingegen für die Programme aus, die ich vor mehr als 20 Jahren auf dem C-64 selbst programmiert habe. Es handelt sich hier

---

12 <http://www.infinite-loop.at/Power64/index.html>

um viele Programme in Commodore-64 BASIC. Die Frage, die sich mir stellte, war nun die, ob und wie ich diese Daten von meinen alten (auf dem Original C-64 noch laufenden) 5,25 Zoll-Disketten von 1982 bis 1987 auf die Festplatte meines PC kopieren und ich diese Daten auch für den C-64-Emulator nutzen kann.

- Der erste Versuch, einfach ein vor einigen Jahren noch gebräuchliches 5,25 Zoll-Laufwerk<sup>13</sup> an den PC anzuschließen und die C-64 Daten am PC auszulesen, schlug zunächst einmal fehl. Grund hierfür waren die unterschiedlichen Dichten und die unterschiedlichen Dateisysteme der 5,25 Zoll-Disketten. Auf eine Diskette des C-64 war Platz für 170 KB, damals einfache Dichte (single density). Die Disketten für den PC hatten jedoch doppelte Dichte (double density) oder gar hohe Dichte (high density), sodass das mit zur Verfügung stehende Diskettenlaufwerk die C-64 Disketten nicht lesen konnte.
- Nach kurzer Recherche entdeckte ich eine Seite im Internet (die Community für den C-64 ist immer noch enorm groß), die Schaltpläne für einige Kabel abbildete, mit denen man seinen PC mit den Diskettenlaufwerken seines C-64 verbinden konnte. Mit Hilfe des Programmes Star Commander<sup>14</sup>, das unter DOS läuft, kann man damit seine Daten von C-64 Disketten auf seinen PC kopieren und auch gleich Disk-Images erstellen. Inzwischen kann man solche Kabel auch bestellen und muss nicht selbst zum LötKolben greifen (Für die Nutzung dieses Programms muss natürlich eine lauffähige DOS-Version zur Verfügung stehen, ist keine verfügbar, kann evtl. eine emuliert werden :-)
- Nach diesen Aktionen kann ich nun meine alten selbst erstellten Programme auf vielen C-64 Emulatoren wieder nutzen, weiterentwickeln und spielen, wie in Abbildung 2 und 3 zu sehen ist (und das sogar auf mehreren virtuellen Commodore-64 gleichzeitig).

---

13 Den ersten Versuch unternahm ich vor etwa vier Jahren, 5,25-Zoll-Diskettenlaufwerke waren nicht mehr wirklich gebräuchlich, aber noch erhältlich. Heute werden selbst die 3,5-Zoll-Laufwerke schon nicht mehr mit einem neuen Rechner verkauft. Neue Medien zum Datenaustausch und zur Speicherung sind heute USB-Stick, DVD, CD-ROM und Festplatte.

14 <http://sta.c64.org/sc.html>

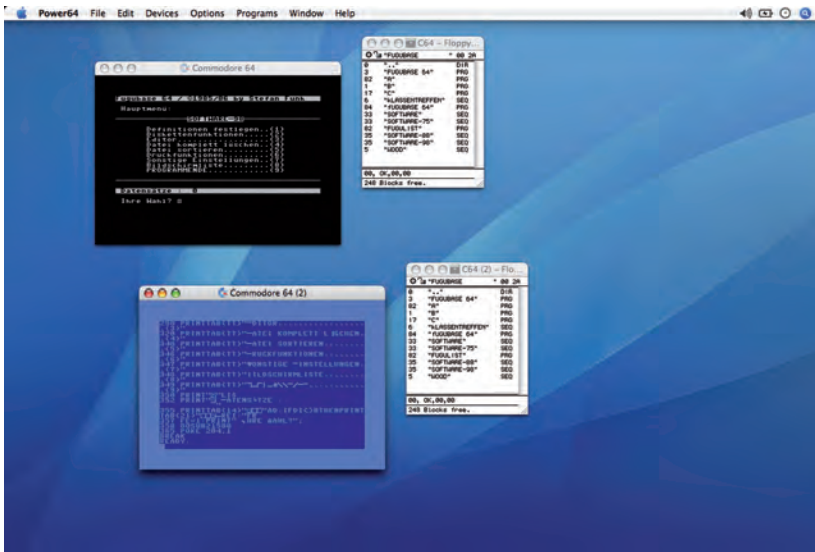


Abbildung 2: Fugabase 64, ein Datenverwaltungs-Programm in Basic für den C-64, emuliert unter Mac OS X (S. E. Funk, 1985/86)

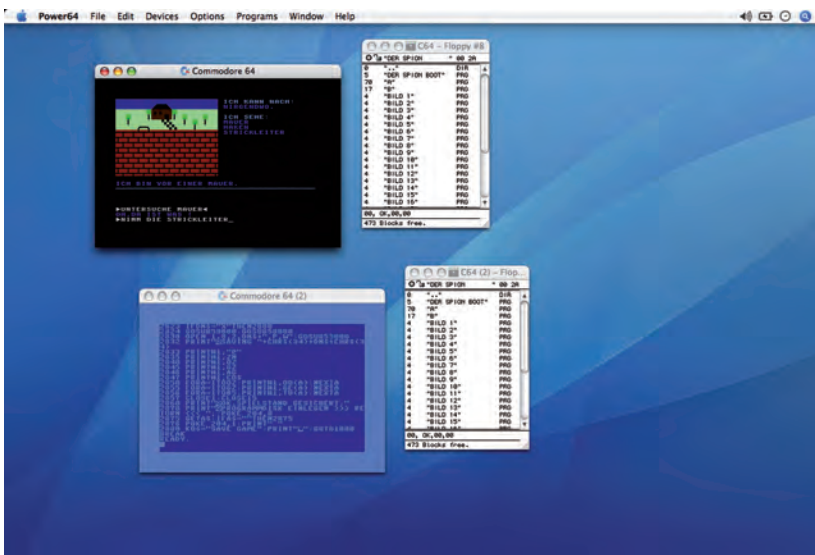


Abbildung 3: Der Spion, ein Adventure in Basic für den C-64, emuliert unter Max OS X (S. E. Funk, 1987)

Beispiel: Eine Emulation in der Emulation



- Es ist nun auch möglich, einen Emulator wiederum zu emulieren, wenn ein weiterer Generationswechsel einer Hardwareplattform ansteht. Ein praktisches Beispiel ist ein Apple Notebook, das unter Mac OS X, einem Unix-basierten Betriebssystem, arbeitet. Auf diesem werden zwei Emulatoren und ein weiteres originales Betriebssystem gestartet.
- Auf diesem Rechner wird das Programm Q gestartet<sup>15</sup>, das eine Hardware-Plattform emuliert (einen Pentium x86 mit diversen Grafik-, Sound- und weiteren Hardwarekomponenten). Es basiert auf dem CPU-Emulator QEMU.<sup>16</sup>
- Auf dieser virtuellen Hardwareplattform kann nun ein originales Windows 98 installiert werden, so dass man ein reguläres, altbekanntes Windows 98 auf diesem nicht-Windows-Rechner nutzen kann. Das installierte Windows 98 kann selbstverständlich alle Programme für Windows 98 ausführen, da es sich tatsächlich um ein originales Windows 98 handelt. Sogar ein Windows-Update über das Internet ist möglich.
- Jetzt kann natürlich auch ein C-64 Emulator für Windows, hier der VICE<sup>17</sup>, gestartet werden. Darauf laufen nun alle altbekannten und beliebten Commodore-64 Programme.
- Probleme kann es bei dieser Art von Emulation zum Beispiel bei der Performanz geben und je nach Qualität der Emulatoren auch mit hardware-spezifischen Dingen wie Grafik, Sound und angeschlossener Peripherie (Mäuse, Joysticks, etc.). Der C-64 Emulator muss schließlich durch Windows über die virtuelle Hardware (Emulation QEMU) auf die reale Hardware des Notebooks zugreifen. Bei steigender Komplexität solcher Emulationsszenarien wird die Anzahl der möglichen Fehler stark ansteigen. Als Beispiel siehe Abbildung 4.

## Der Universal Virtual Computer (UVC)

Mittlerweile gibt es einen elaborierteren Ansatz der Emulation, den Universal Virtual Computer (UVC) von IBM. Der UVC ist ein wohldokumentierter virtueller Computer, der auf unterschiedlichen (auch zukünftigen) Architekturen nachgebildet werden kann. Aufgebaut ist er ähnlich wie ein heute existierender Computer, der beispielsweise Speicherzugriff ermöglicht. Mit Hilfe dieser Dokumentation ist es einem Programmierer auch auf zukünftigen Systemen

---

15 <http://www.kju-app.org/>

16 <http://www.nongnu.org/qemu/>

17 <http://www.viceteam.org/>

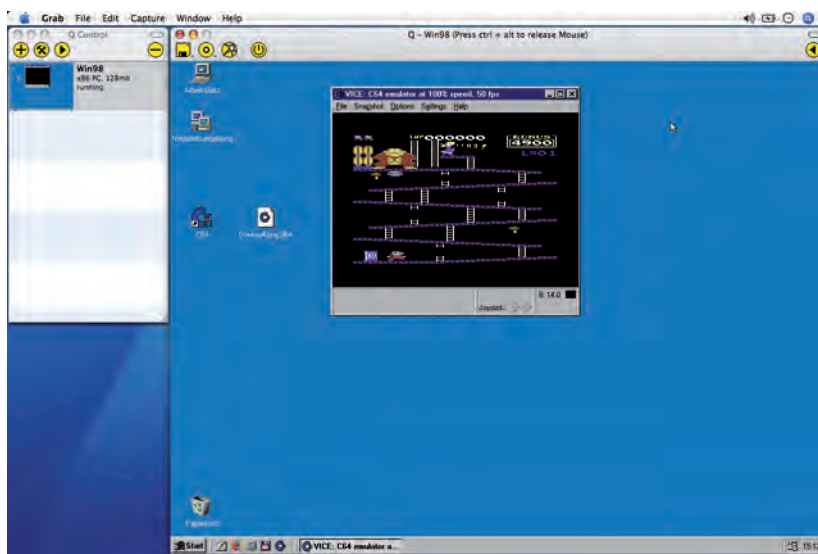


Abbildung 4: Das Videospiel Donkey Kong auf einem C-64 Emulator auf einem Windows 98 auf einem virtuellen Pentium auf einem Apple PowerBook unter Mac OS X

möglich, diesen virtuellen Computer zu implementieren. Auf diesem virtuellen Computer aufbauend können nun Programme geschrieben werden, die zum Beispiel eine PDF-Datei lesen oder Grafiken darstellen können.

Archiviert wird jetzt der PDF-Reader (der Bildbetrachter), der für den UVC programmiert wurde, sowie das originale PDF-Dokument (oder die originale Grafik) selbst. Ein zukünftiger Nutzer kann dann auf einer zukünftigen und wahrscheinlich hoch entwickelten Hardware auch in ferner Zukunft noch mit Hilfe der Dokumentation des UVC einen solchen implementieren und mit Hilfe dieses virtuellen Computers den PDF-Reader starten, mit dem das archivierte PDF-Dokument dargestellt wird. Die Dokumentation muss selbstverständlich erhalten bleiben und lesbar sein.

Ein Problem dieser Idee ist sicherlich, dass bei zunehmendem Anspruch an die Emulation, die auf dem UVC laufen soll, eine Programmierung derselben immer schwieriger wird. Es wird sehr kompliziert, wenn für den UVC ein Betriebssystem wie Linux oder Windows programmiert werden soll, mit dessen Hilfe dann die Applikationen von Linux oder Windows genutzt werden können. Schon eine nachprogrammierte Version eines Textverarbeitungsprogrammes wie zum Beispiel Word, mit dem später alte Word-Dokumente (.doc) auf dem UVC gelesen und bearbeitet werden können, ist ein höchst umfangreiches Un-

ternehmen. Zumal hier nicht nur die Formatbeschreibung, sondern auch alle Programmfunktionen bekannt sein müssen.

## **Zusammenfassung: Vor- und Nachteile von Emulation**

### *Vorteile von Emulation*

- Bei der Emulation bleiben die Originalobjekte unverändert.
- Eine Konvertierung der Objekte ist nicht nötig.
- Für die Emulation wird weniger Speicherplatz benötigt, da keine Migrationen gespeichert werden müssen.

### *Nachteile von Emulation*

- Für komplizierte Objekte/Systeme (wie Betriebssysteme oder Anwendungsprogramme) sind Emulatoren technisch schwer zu implementieren.
- Es entsteht ein hoher Aufwand pro Hardware-Generationswechsel. Es müssen für jede Plattform neue Emulatoren entwickelt werden.
- Die Spezifikationen für die zu emulierenden Objekte/Systeme sind nicht immer hinreichend bekannt.

## **Literatur**

- Lorie, Raymond (2002): *the UVC: a method for preserving digital documents – proof of concept*, <http://www-5.ibm.com/nl/dias/resource/uvc.pdf>
- Nationaal Archief (2005): *Technical Description of the Universal Virtual Computer (UVC) - Data preservation process for spreadsheets*, <http://www.digitaleduurzaamheid.nl/bibliotheek/docs/TDUVCv1.pdf>
- Oltmans, Eric; Nanda Kol (2005): *A Comparison Between Migration and Emulation in Terms of Costs*, <http://www.worldcat.org/arcviewer/1/OCC/2007/08/08/0000070519/viewer/file959.html#article0>