

Robert Musil und das Matroschka-Prinzip

Linux, Wine und VirtualBox werden als Open Source-Werkzeuge eingesetzt, um die digitale Edition der Klagenfurter Ausgabe besser zugänglich zu machen. Gleichzeitig ist dabei ein kulturelles Umdenken in der Arbeitshaltung zu beobachten. Harald Jele

Die Begriffe Linux und Wine sind üblicherweise welche, die eher im technischen Umfeld Verwendung finden. Über Umwege und in durchaus überraschender Weise gelangten diese mit dem hier beschriebenen Ansatz in den Kontext literaturwissenschaftlicher Überlegungen.

Papier, Microfiche oder digital?

Bereits 1984 wurde damit begonnen, den literarischen Nachlass des österreichischen Schriftstellers Robert Musil (1880 - 1942) in elektronischer Form zu erfassen und entsprechend den vorhandenen Möglichkeiten zu publizieren. Zu diesen Möglichkeiten zählten in den 80er-Jahren durchaus noch solche, die maschinenschriftlichen, handgeschriebenen oder am Computer transkribierten Texte letztlich nur in Papier- oder eventuell auch Microficheform zu veröffentlichen. Letztlich kam es 1992 jedoch zu einer CD-ROM-Edition, die im Verlag Rowohlt vertrieben wurde.

Obwohl bei allen Überlegungen der Umstand stets mitbedacht wurde, die durch die geförderten Projekte entstehenden Leistungen auch dauerhaft konservieren zu wollen, so wurde dieser Aspekt, bezogen auf die digitale Form, eher sträflich vernachlässigt. Leider musste bereits zum Erscheinen der CD-ROM-Ausgabe der Umstand wahrgenommen werden, dass diese kaum mehr an die Frau oder den Mann zu bringen war. Die eingesetzten Technologien war in fast jeder Hinsicht für den bevorstehenden technischen Wandel einfach nicht vorbereitet.

Dass diese Erkenntnis zu keiner wesentlichen Neuausrichtung, und erst sehr viel später zu einem Umdenken geführt hat, mag manche in Erstaunen versetzen. Literaturwissenschaftler sind sich jedoch in weiten Teilen nicht (immer) bewusst, in welche Käfige und Mühlen sie sich mit ihren Tätigkeiten begeben, wenn sie den Umstand verneinen, sich um die Einhaltung möglichst offener Standards kümmern oder diese gar einfordern zu müssen.

Eine Windows-Anwendung als Lösung?

Auf der Basis der Erstellungsdateien konnte jedoch in weiterer Folge, und durch einigermaßen viele händische Nachbearbeitungen, ein Textkorpus in elektronischer Form erstellt werden, der für einen Neuanfang dienlich sein konnte.

Dieses Textkorpus wurde 1999 in die, nur für das Betriebssystem Windows verfügbare, Software *FolioViews* importiert und in weiterer Folge auch in dieser weiter bearbeitet [1]. Perspektivisch gesehen mutet eine solche Entscheidung heute, nicht nur aus rein technischer Sicht, als eine sehr unreflektierte an. Die damit verbundenen Nachteile folgten auf die Hand. Zum einen Teil lagen diese sofort vor, zum anderen Teil mussten sie schleichend zur Kenntnis genommen werden und einige schlummern wohl noch immer unerkannter Weise vor sich hin.

Sofort erkannt wurden die technisch wie technologisch engen Grenzen, in die man sich mit einer solchen Lösung begeben hatte:

- Mehrplatzfähigkeit war, und ist auch weiterhin, nicht gegeben. Dieser Umstand führt in einer Arbeitsumgebung, in der mehrere oder gar viele Kolleginnen und Kollegen zusammenarbeiten müssen, immer zu einem nicht unerheblichen Mehraufwand. Microsoft musste zum damaligen Zeitpunkt mit Windows in solchen Umgebungen ohnehin meist passen. Wieso sollte eine einzelne Anwendung für dieses Betriebssystem daher anderes vorsehen?
- wollte die Software irgendwann einmal nicht (mehr) arbeiten, so konnte man im besten Fall nur erahnen, warum dies der Fall ist und „probieren“, wie man den Fehler selbst beheben kann

Schleichend stellte sich ein, dass man

- von einer technischen Dokumentation der internen Abläufe nur träumen konnte und weiterhin kann. Man weiß daher im Grunde bei den einzelnen Vorgängen der texteditorischen Arbeit nicht mehr, als dass man mit „einem Programm“ arbeitet
- beim Import und Export von Daten an die vom Hersteller vorgegebenen Formate mit Gedeih und Verderb gebunden war und ist
- über die Form, wie Daten intern gespeichert sind, völlig im Dunkeln gelassen wird. Diese könnten eigentlich nur durch aufwändiges Reengineering ermittelt werden. Welche Datenstrukturen oder gar Zeichensätze (intern) verwaltet werden (können), ist und bleibt unbekannt
- was Benutzerinnen und Benutzern mitgeteilt werden muss, wenn sie nicht mit Windows arbeiten wollen oder können, liegt

- auf der offenen Hand
- im Fall, dass ein technologischer Wandel ins Haus steht, die Daten nicht einfach mitnehmen kann, sondern (wieder einmal) grundlegend mit einigem manuellen Aufwand und findiger Programmierung zu Werke gehen muss

Diese Umstände sind im Wesentlichen, zumindest für einen Kundigen mit ein wenig grundlegendem Open Source-Verständnis, eigentlich nicht überraschend. Es gilt wohl sogar das Gegenteil: Diese Zusammenhänge sind vorhersagbar. Und vorhersagbar ist zugleich wohl auch der täglich anzutreffende Umstand, dass alle Proponenten, die für eine solche, geschlossene Lösung eintreten, diese auf Dauer gravierenden Nachteile stoisch verneinen. Als Rechtfertigung dient in der Regel das ohnehin bekannte und äußerst beliebte Argument, dass man sich mit einer solchen Entscheidung zumindest nicht von anderen unterscheidet und die breite Mehrheit wohl (in der Herde) für „irgendeinen“, wengleich auch noch unbekanntes, Fortbestand sorgen wird.

Die Konsequenz aus einer solchen Haltung mag für ein Produkt wie Microsoft Word stimmen. Die Frage, ob man mit einer solchen Haltung auch in einem eher kleinen Nischenmarkt überleben kann, ist fraglich und wurde, bezogen auf die digitale Edition des Nachlasses von Robert Musil, bereits schon (mindestens) einmal beantwortet.

Zwischenlösung in Sicht

2009 ist eine gründlich überarbeitete und wesentlich erweiterte Edition zum Nachlass von Robert Musil erschienen. Diese wurde, wie viele andere literarische Werkeditionen auch, als reines Windows Produkt der bereits genannten Software *FolioViews* am Robert Musil-Institut für Literaturforschung der Universität Klagenfurt erstellt und ist über die entsprechende Online-Seite erhältlich [2].

Seit dem Erscheinen dieser DVD sind die erwartbaren und zugleich vorhersagbaren Umstände schließlich auch eingetroffen. Diese sind den Rückmeldungen entsprechend überwiegend folgende:

- interessierte Benutzerinnen und Benutzer können nicht oder nur mit einem zusätzlichem finanziellen und zeitlichen Aufwand zu den Inhalten gelangen, weil sie nicht mit Microsoft Windows arbeiten
- institutionelle Kunden können ihren Mitarbeiterinnen und Mitarbeitern die DVD nicht über ihr vorhandenes Netzwerk anbieten
- die Inhalte sind immer nur für eine Interessierte oder einen Interessierten, und diese

nur an einem Gerät, zugänglich. Alle weiteren müssen warten

Um die davon betroffene, im Großen eigentlich unbekannt, aber keinesfalls zu vernachlässigende, potentielle Klientel mittelfristig in ihren Bedürfnissen ansprechen zu können, wurde aufgrund der bisher gewonnenen Erfahrungen eine Zwischenlösung angestrebt, die durch Kapselung der Windows-Anwendung nach dem in der Informatik durchaus üblichen Prinzip einer „Matroschka“ (Матрёшка) – quasi in einem „Container“ – arbeiten soll.

Der Vorgang der Implementierung (das Einpacken der kleineren Teile in die größeren) ist dabei exakt der umgekehrte, wie die entgeltliche Präsentation gegenüber den Benutzerinnen und den Benutzern:

- eine virtuelle Maschine wird installiert,
- darin ein Betriebssystem mit ausgeprägten Netzwerk- und Mehrbenutzerfähigkeiten,
- diesem Betriebssystem werden die Systembibliotheken von Wine hinzugefügt, um die Windows-Systemaufrufe dem nativen Betriebssystem zu vermitteln,
- und innerhalb dieser Umgebung wird die Windows-Anwendung entsprechend eingerichtet.

Die wesentliche Grundvoraussetzung für das Design der Implementierung lieferten dafür die bereits vermittelten Überlegungen zu den damit angesprochenen Themen: Offenheit. Im technischen Sinn am besten umgesetzt mit Werkzeugen und Methoden, die im Quelltext vorliegen und die mit einer dazu passenden Lizenz entsprechend frei und kostenlos eingesetzt und verbreitet werden dürfen.

So fiel die Entscheidung für die Wahl einer Software zur Virtualisierung relativ einfach. *VirtualBox* verfügt über die notwendigen Eigenschaften, die man zeitgemäß erwarten darf und ist zudem passend (dual) lizenziert. Für mehrere Betriebssysteme existieren fertige Installationspakete. Für viele weitere sind Anleitungen zum Kompilieren im Internet auffindbar.

Der Ansatz, dass Linux als das darin installierte Betriebssystem zur Anwendung gelangt, mag zwar wenig überraschend anmuten, war aber letztlich in der Entscheidungsfindung nicht unbedingt zwingend. Die ausschlaggebenden Gründe für die Wahl von Linux waren

- die außerordentlich gute Dokumentation in eigentlich allen Belangen,
- die große Auswahl an unterschiedlichen Distributionen mit relativ hoher Bekannt-

- heit,
- die vorhandenen Möglichkeiten, mit einfachen und doch effizienten Mitteln das System, auch über einen längeren Zeitraum, auf dem letzten Stand zu halten,
- umfangreiche Softwarerepositories, die das eigene Kompilieren unnötig machen
- ein für den Anwender einfaches Bedienungskonzept, mit dem auch Benutzer anderer Betriebssysteme zurecht kommen können

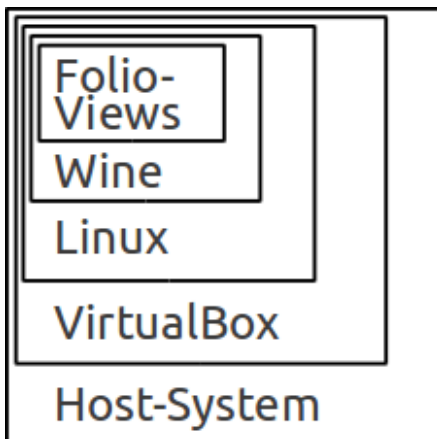


Abb. 1: Schema der einzelnen Containerschichten

Ausgewählt wurde letztlich eine Langzeit unterstützte Ausgabe von Ubuntu (10.04 LTS).

Wine wurde in der für die Distribution von Ubuntu vorliegenden Version (1.2.2) installiert. Die vorhandenen Installationspakete sind relativ aktuell und, wie sich in weiterer Folge zeigte, für die Anwendung aktuell genug. Die Beschäftigung mit den Wine-Server-Instanzen ist eine sehr zeitraubende. Dieser Umstand rührt aber nicht daher, dass die Prozesse von Wine schwierig oder umständlich zu benutzen wären, sondern vielmehr daher, dass der eingesetzten Windows-Software keine technische Dokumentation beigelegt ist. Das Installieren, die Inbetriebnahme und die Konfiguration sind damit einem Blindflug durch die Windows-Technologie gleichzusetzen. Zwar ist im Debugging-Mode von Wine vieles zu erkennen, trotzdem kann auf die in den Ausgaben des Debuggers festgestellten Probleme nicht immer angemessen reagiert werden. So vergehen oft Wochen, bis eigentlich kleine Unannehmlichkeiten gelöst und entfernt sind. Damit bleibt letztlich in vielen Fällen nur die Erkenntnis, genau zu wissen, warum die Software unter bestimmten Umständen (auch nativ unter Windows) nicht funktioniert. Eine Behebung durch die Herstellerfirma wird im Grunde jedoch bloss ignoriert und bleibt, wie im Closed Source-Bereich häufig üblich, trotz des oft eindeutigen Fehlercharakters aus.

Herausforderungen für die Benutzer

Im Fall der Auslieferung der Klagenfurter Ausgabe in Form einer Windows-DVD müssen die Benutzerinnen und Benutzer selbst für eine ordnungsgemäße Installation sorgen. Auch wenn das mitgelieferte Setup-Programm die meisten Fallstricke umschiffet, so bleibt ein gewisses Restrisiko bestehen, dass aufgrund der Konfiguration des betreffenden Endgerätes auch eine vorschriftsmäßige Installation nicht immer stets zum Ziel führt und die Anwendung in weiterer Folge nicht gestartet werden kann.

Mit der Auslieferung der Ausgabe in einer fertig installierten und konfigurierten virtuellen Maschine scheinen diese Fallstricke gebannt.

Trotzdem können im Vorhinein nicht alle möglichen Probleme als gelöst oder in jedem Fall als einfach zu lösen angenommen werden:

- ein an der USB-Schnittstelle des Wirtssystems angeschlossener Drucker mag im besten Fall ordentlich an das Gastsystem vermittelt werden. Liegen für diesen Drucker jedoch keine Linux-Treiber vor, so kann der Drucker als Ausgabegerät nativ nicht angesprochen werden. Gedruckt kann dann eigentlich nur über den Umweg der Erstellung von PostScript- oder PDF-Dateien werden
- USB-Geräte, so unter anderem auch der ausgelieferte USB-Stick, müssen durch die virtuelle Maschine hindurch dem Gastsystem zugänglich gemacht werden. Dieser Umstand zeigt sich beim Einsatz von VirtualBox als ein im Grunde bewältigbarer; trotzdem bedarf es einer gewissen Aufmerksamkeit für die Wahrnehmung dieser Problematik durch die Benutzer und einer Anleitung, was zu tun ist, wenn Geräte zwar erkannt, dem Gastsystem aber auf Anhieb nicht weitergereicht werden
- zum Austausch von Dateien mit dem Gastsystem können für die virtuelle Maschine Verzeichnisse im Wirtssystem eingerichtet werden, die letztlich beiden (dem Gast und dem Wirt = „Shared“) zugänglich sind. Deren Einrichtung kann nach den bisher gemachten Erfahrungen durchaus mühsamer sein, als dies laut der entsprechenden Dokumentation zu erwarten ist
- die Installation und Einrichtung von VirtualBox ist ausreichend dokumentiert, sodass diese Vorgänge wohl als lösbar bzw. machbar angesehen werden können. Hinweise bezüglich der ausgelieferten Konfiguration erscheinen jedenfalls angebracht

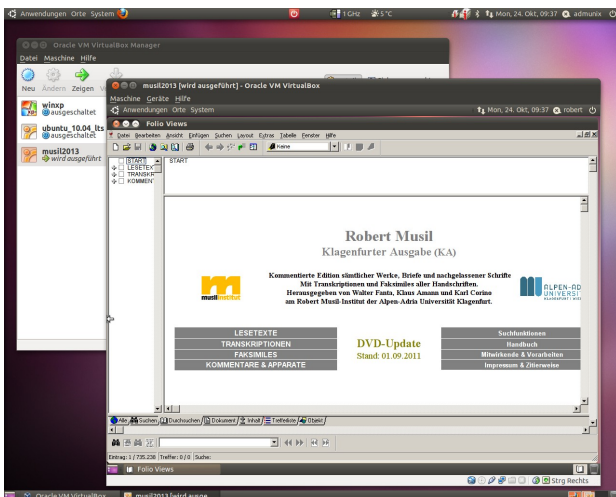


Abb. 2: Screenshot zu den einzelnen Containerschichten

Netzwerkfähigkeit

Mit der Realisierung dieses Ansatzes ist im Grunde genommen auch eine Integration der Anwendung in ein lokales Netzwerk möglich. Linux, als das „Parade-Netzwerk-Betriebssystem“ schlechthin, bietet eine Fülle an Möglichkeiten, Anwendungen, die auf einem Server laufen, über unterschiedlichste Protokolle, auf einem Client-Rechner zur Anzeige zu bringen.

Allein der Umstand, dass für den Client nicht davon ausgegangen werden kann, dass dieser all diese Fähigkeiten auch mitbringt, schränkt die Anzahl der sinnvollen Möglichkeiten sofort auf eine Hand voll Ansätze ein.

Der einfachste Ansatz, nämlich die Anzeige über einen SSH-Tunnel einem lokalen X-Server zu übergeben, scheitert an den Erfahrungen, dass diese auch unter dem Betriebssystem Mac OS-X nicht immer sehr stabil zu funktionieren scheinen. Einem Administrator von Microsoft Windows ist wohl ohnehin nicht zu vermitteln, dass dies ein gangbarer Ansatz sein könnte.

Der Einsatz des Protokolls NX der Firma NoMachine erscheint im Moment als nicht unbedingt ratsam:

- nicht alle Programmteile, die für eine solche Implementierung notwendig sind, liegen im Quelltext vor,
- die Zukunft dieses Produkts wurde zudem in der Vergangenheit mehrfach von unterschiedlichen Seiten angezweifelt,
- der dabei entstehende Aufwand ist gegenüber dem Nutzen für eine einzelne Anwendung wohl nicht in jedem Fall rechtfertigen.

Letztlich ist die Entscheidung dafür gefallen, die graphische Ausgabe über VNC-Verbindungen dem Client zu vermitteln, die freizugebenden Verzeich-

nisse über eine einfache Konfiguration von Samba herzustellen und für die Ausgabe an Druckgeräten die Netzwerkfähigkeit von CUPS zu nutzen.

Dass diese Entscheidungen nicht unbedingt der Weisheit letzter Schluss sein müssen, ist verständlich. Weitere Lösungsansätze sollten noch mit der anzusprechenden Klientel besprochen werden. Für Anregungen oder Hinweise sind wir sehr dankbar.

Zukunftsansichten

Dass dieser Lösungsansatz bloß als einer für den Übergang angesehen werden kann, liegt zudem auf der Hand:

Die Probleme, die sich langfristig aus der Datenhaltung in einer solchen Umgebung zeigen, sind groß und akut genug, um bereits mittelfristig eine dauerhafte Lösung anzustreben.

Die Entscheidung für eine Zielplattform ist noch zu treffen. Die wesentlichsten Parameter, die bei der Entscheidungsfindung zu berücksichtigen bleiben, sind:

- das Datenformat ist in einer offenen Form, möglichst unter Einhaltung entsprechender Standards, zu wählen. Der XML-Dialekt der TEI (= Text Encoding Initiative) [3] wird im Besonderen zu beachten sein. Ein diesbezüglich passendes System, das in den letzten Jahren an Bekanntheit gewonnen hat, könnte z. B. *TextGrid* [4] sein
- gleiches gilt für den Zeichensatz: Für diesen ist das heute übliche Unicode Character Set UTF-8 anzustreben. Die Migration dorthin wird ohne intellektuelle Nacharbeit nicht zu bewältigen sein, da die bestehende Kodierung auf der Basis des Zeichensatzes Windows 1252 in nicht eindeutiger Weise abgebildet wurde
- die Anwendung wird als eine Web-Applikation zu realisieren sein, die eine vollständige Bedienung durch die Benutzerinnen und Benutzer im Web-Browser ermöglicht und keine besonderen Installationen auf der Seite des Clients verlangt

Auch wenn das Ziel einer Migration der Klagenfurter Ausgabe bekannt sein wird und entsprechende Methoden und Werkzeuge zur Umstellung hergestellt sind, so wird dieser Prozess sich letztlich doch über einen längeren Zeitraum hinziehen. Der Umfang der Edition hat die Fülle von 12000 bearbeiteten Seiten aus dem Nachlass, die dicht mit Transkriptionszeichen gefüllt sind, längst überschritten. Zudem ist noch unbekannt, ob sämtliche Informationen, die in der Datenbank Infobase von FolioViews mit rein maschinellen Mitteln und zudem in exakter Weise exportiert werden können.

Quellen

- [1] Fanta, Walter: Zur Immortalität elektronischer Korpora am Beispiel der Musil-Edition, Abs. 32. In: Forum Computerphilologie: [<http://computerphilologie.tu-darmstadt.de/jg09/fanta.html#FNRef21>]
- [2] Robert Musil : Klagenfurter Ausgabe. (KA-DVD 2009): [<http://wwwg.uni-klu.ac.at/musiledition>]
- [3] Konsortium zur *Text Encoding Initiative* (TEI) [<http://www.tei-c.org>]
- [4] Forschungsverbund *TextGrid* [<http://textgrid.de>]

Der Autor



Dr. Harald Jele ist Mitarbeiter an der Universität Klagenfurt und beschäftigt sich zur Zeit (auch) mit den technischen Aspekten der Klagenfurter Ausgabe zum Nachlass von Robert Musil.