

Praktikumsstelle: CCC Campus-Computer-Center GmbH  
Praktikant: Matthias Jauernig  
Zeitraum: 26.09.05 – 10.02.06

# Praktikumsbericht

Leipzig, den 10.02.06

## Praktikumsaufgabe

### **Entwicklung einer Software zur Update-Automatisierung von dezentralen Softwareinstallationen in einer Client-Server-Umgebung**

#### Aufgabenbeschreibung:

Ziel meines Praktikums bei der CCC Campus-Computer-Center GmbH bestand darin, ein System für automatische Software-Updates zu entwickeln. Clients, die über ein Netzlaufwerk auf die Produktdaten Zugriff haben, sollen damit ihre Produkte auf den neuesten Stand halten können. Dazu war zunächst eine Administrationskomponente zu entwickeln, mit welcher zu aktualisierende Projekte angelegt werden können. Als zweiter Schritt stand dann die Entwicklung der Client-Komponente an, die eine Projekt-Datei ausliest und die Aktualisierung anhand der darin angegebenen Daten durchführt.

Bei beiden Anwendungen sollte es sich um komplette Neu-Entwicklungen handeln, die Client-Komponente dabei ein bereits bestehendes einfacher gestricktes Programm ersetzen und über die Abbildung dessen Funktionalität hinausgehen.

Was zunächst nach einer einfachen und schnell zu erledigenden Aufgabe aussah, stellte sich komplexer dar als erwartet. Eine Vielzahl von Konfigurationsmöglichkeiten und Optionen, hohe Flexibilität und verschiedene Erweiterungen ließen die Entwicklung zu einer praktikumsfüllenden Aufgabe avancieren.

#### Begriffsklärung:

Das gesamte Softwarepaket trägt den Namen Kamajii, der auch schon zuvor für das Projekt (zumindest den Client) verwendet wurde. Kamajii ist eine Figur aus dem Zeichentrick „Chihiros Reise ins Zauberland“ und ist dort für die Wasserverteilung in einem Badehaus zuständig, ähnlich wie es für die Datenverteilung mittels der Client-Komponente möglich sein sollte.

Von diesem Begriff leiten sich die Namen der Teilprodukte ab:

- *KaVe* - Kamajii Verwaltung: die Administrationskomponente,
- *KaSy* - Kamajii Synchronisator: die Clientkomponente.

#### Dieser Bericht:

Ich möchte den Bericht in wöchentliche Einheiten unterteilen, in denen ich kurz auf die darin verrichteten Tätigkeiten eingehen will.

Am Ende möchte ich anderen Studenten in einer kleinen Einschätzung meine Erfahrungen vermitteln, die ich während des Praktikums bei der CCC GmbH gemacht habe.

### **Woche 1: 26.09. – 30.09.05**

Die erste Woche bzw. die ersten beiden Wochen standen im Zeichen der Einarbeitung in die Entwicklungsumgebungen Visual Studio .NET sowie Visual Studio 6.0, da die Verwaltungskomponente KaVe mit C# und .NET entwickelt werden sollte, die Clientkomponente KaSy hingegen mit VC++ 6.0 und MFC, um auf den Clients nicht nachträglich das .NET-Framework installieren zu müssen.

Die erste Aufgabe bestand darin, eine simple Version der Administrationskomponente zu entwickeln, in der ein Dateibaum erstellt werden kann. Dieser sollte dann in eine XML-Datei gespeichert und aus dieser wieder geladen werden können. Diese Aufgabe umfasste also die Einarbeitung in C#, den VS .NET - Designer sowie die XML-Verarbeitung in .NET.

Nach Abschluss dieser Aufgabe sollte eine zweite Testanwendung erstellt werden, die eine vereinfachte Client-Komponente darstellt, jedoch schon einiges von der Funktionalität der beabsichtigten Anwendung enthält. Dazu habe ich mich zunächst in VC++ 6.0 und die MFC eingearbeitet. Auch das Auslesen von XML-Daten in der MFC war ein Thema, da Projekte, die mit dem vereinfachten Administrationstool erstellt wurden, von der Testanwendung geladen werden sollten.

### **Woche 2: 04.10. – 07.10.05**

In diesem Zeit-Rahmen habe ich die zweite Testanwendung weitergeführt und zum Abschluss gebracht. Vordergründig ging es hier um das Kopieren von Dateien, um Prüfen von Dateiattributen, um das Auseinandersetzen mit verfügbaren ZIP-Bibliotheken und den Ablauf der Testanwendung in 2 Threads, um den Abbruch des Kopier-Vorgangs zu ermöglichen.

Am Ende entstand ein Programm, welches ein Projekt aus einer XML-Datei laden kann, ein direktes oder indirektes Kopieren über das Temp-Verzeichnis erlaubt und den aktuellen Status des Vorgangs in einem Fortschrittsbalken anzeigt. Ebenso können damit gepackte Dateien automatisch entpackt werden, wenn es in der Administrationskomponente festgelegt wurde.

### **Woche 3: 10.10. – 14.10.05**

Nach 2 Wochen Einarbeitung wurde diese erste Phase einvernehmlich als abgeschlossen erklärt und ich konnte damit beginnen, die Administrationskomponente zu entwickeln. Zunächst hab ich mir mit Markus, meinem Betreuer, einen Überblick über die Anforderungen verschafft, woraufhin er ein erstes Schema für die zu erstellende XML-Datei entwickelt hat. Wenn auch nicht gefordert, hielt ich es für notwendig, zumindest für mich ein grobes Klassendiagramm zu erstellen, um ein Gesamtkonzept zu besitzen und die Anforderungen im Auge zu behalten.

Im ersten Schritt der Programmierung habe ich die Klassen meines Klassendiagramms implementiert und mit grundlegenden Funktionen versehen. Nachfolgend machte ich mir Gedanken um ein Designkonzept und begann damit, dieses umzusetzen.

### **Woche 4: 17.10. – 21.10.05**

Es galt, das erstellte Designkonzept umzusetzen und die ersten Programmfunktionalitäten zu implementieren. Am meisten ist hier das Anlegen des Projekt-Inhalte-Baumes hervorzuheben, über den die auf den unterschiedlichen Clients zu installierenden/aktualisierenden Elemente festgelegt werden können. Mit dem Umsetzen der einzelnen Controls einher ging auch die Einbindung der Funktionalitäten in die jeweiligen Klassen.

### **Woche 5: 24.10. – 28.10.05**

Es wurden weitere Controls zur Anwendung hinzugefügt und die entsprechenden Klassen mit Funktionalität versehen: die Projekt-Optionen, über welche optionale Einstellungen (z.B. zu erstellende Backups, anzulegende Log-Dateien, Start nach Beendigung der Synchronisation, ...) definiert werden können und z.B. die Location-Verwaltung, über welche sich Pfade von Quell- und Ziel-Verzeichnissen festlegen lassen. Um Code-Redundanzen möglichst zu vermeiden, habe ich weiterhin gemeinsam genutzte Funktionen in eigene Controls ausgelagert und diese mit Events versehen, um deren Funktionalität nutzen zu können.

Weiterhin habe ich das Einlesen der Projektdaten aus einer XML-Datei und das Speichern des Projekts in die Datei implementiert, was einen wichtigen Schritt auf dem Weg zum fertigen und testbaren Produkt dargestellt hat.

### **Woche 6: 01.11. – 04.11.05**

Mit dem Ende der 5. Woche war der Hauptteil der Arbeit am Admin-Tool eigentlich schon getan, nun ging es ans Testen, Bugfixen, Erweitern und Verfeinern der Anwendung. So z.B. die Einführung einer Funktion, um zu erkennen, wenn Änderungen im Projekt durchgeführt wurden, um den Benutzer bei Programmende bei nicht gespeichertem Projekt darauf hinzuweisen, dass geänderte Daten vorliegen. Weitere größere Erweiterung war die Einführung einer Toolbar, um direkt auf die wichtigsten Funktionen zugreifen zu können.

Am Ende der Woche fand eine Besprechung im kleineren Rahmen über das weitere Vorgehen statt, wo noch einmal vorzunehmende Erweiterungen und Änderungen diskutiert wurden.

### **Woche 7: 07.11. – 11.11.05**

Diese Woche wurde die Feinabstimmung fortgesetzt. Die bei der Besprechung vorgeschlagenen Änderungen und Verbesserungen wurden umgesetzt, z.B. das Einführen einer MRU-Liste (most recently used), um schnellen Zugriff auf die zuletzt geöffneten Projekte zu haben sowie viele weitere Änderungen, welche die Benutzerfreundlichkeit und die Darstellung verbessert haben.

### **Woche 8: 14.11. – 18.11.05:**

Es ging weiter mit dem Bugfixen und Verbessern der Admin-Komponente. Vor allem aber auch wurden viele Testprojekte erstellt und die damit entstehenden Daten und XML-Dateien auf Konsistenz geprüft, was zu einigen bösen Überraschungen führte, die nach langwierigem Suchen ihrer Ursache schließlich aber schnell beseitigt werden konnten.

Wichtig war die ebenfalls in dieser Woche erstellte Quelltext-Dokumentation, die viel Zeit in Anspruch nahm.

Ende der Woche folgte eine konkretere Besprechung zum Client mit dem Festlegen konkreter Synchronisationsabläufe und der Behandlung evtl. auftretender Probleme.

### **Woche 9: 21.11. – 25.11.05**

Los ging es mit der Programmierung des Clients KaSy. Wie schon bei KaVe hielt ich es für sinnvoll, ein Klassendiagramm zu erstellen, was sich als sehr wichtig erwies, auch in Bezug

auf die Frage, wie sich was am besten umsetzen lässt. Im Anschluss daran konnte es an die Umsetzung der Klassenstruktur gehen. Im gleichen Zug wurde auch schon Funktionalität eingebunden und das korrekte Auslesen von XML-Dateien umgesetzt. Ende der Woche konnte dann schon mit der Implementierung der Kopier Routinen begonnen werden.

### **Woche 10: 28.11. – 02.12.05**

In dieser Woche wurden die Kopier Routinen weiter umgesetzt und die optionalen Projekteinstellungen wie das Erstellen von Backups und Log-Dateien implementiert. Die Programmierung wurde von nun an stets umrandet von Tests und Bugfixing, sowohl in KaVe als auch in KaSy.

### **Woche 11: 05.12. – 09.12.05**

In KaSy wurde weitere Funktionalität eingebracht. So z.B. das Packen und Entpacken von Dateien, das Rollback bei Fehlern oder Abbruch durch den Benutzer und die korrekte Anzeige des Fortschritts im Dialog. Ebenso fand ein analytischer Code-Review durch mich selbst statt, mit welchem viele zuvor unentdeckte Fehler erkannt werden konnten.

### **Woche 12: 13.12. – 16.12.05**

Auch diese Woche wurde weiter getestet, erweitert und Fehler beseitigt. Sowohl in KaSy als auch in KaVe wurden dabei viele Detailanpassungen vorgenommen, um höchstmögliche Bedienbarkeit und Funktionalität zu erreichen.

### **Woche 13: 19.12. – 23.12.05**

Zunächst habe ich den Quelltext von KaSy dokumentiert, was viel Arbeit bedeutet hat. Dass sich diese Arbeit am Ende lohnt steht allerdings außer Frage, zudem konnten durch kritische Betrachtung noch einige Fehler gefunden werden.

Im Anschluss daran wurden erste „härtere“ Tests durchgeführt und auch real existierende Projekte synchronisiert (Knowledge DB). Dabei wurden einige weitere Fehler erkannt und beseitigt.

### **Woche 14: 27.12. – 30.12.05**

Weihnachtsurlaub.

### **Woche 15: 02.01. – 06.01.06**

Zu Beginn des neuen Jahres wurden weitere Testprojekte erstellt, welche sich fast anstandslos synchronisieren ließen, Fehler wurden sofort bereinigt. Weitere Anpassungen in KaVe sorgten für noch mehr Flexibilität und Konfigurierbarkeit.

In dieser Woche wurde auch die Benutzerhilfe für KaVe erstellt, welche sich mit Anwendungsbeispielen und ausführlichen Programmbeschreibungen auf ein über 50 Seiten langes Dokument ausdehnte.

### **Woche 16: 09.01. – 13.01.06**

Zunächst wurde die Hilfe erweitert, dann durch weitere Tests kleinere Fehler gefunden und zudem weitere kleine Features eingebaut. Als wichtige Funktionalität gilt in dieser Woche die Einführung einer „Sicherheits-Option“ zu nennen, durch welche eine manuelle Änderung der Projektdatei über definierbare Bereiche unmöglich gemacht werden kann, indem eine Art Checksumme über diese Bereiche definiert wird.

### **Woche 17: 16.01. – 20.01.06**

Für KaSy habe ich eine Entwickler-Dokumentation erstellt, welche die grundlegenden Konzepte des Programms auflistet und erklärt. Weitere optionale Funktionalitäten wurden hinzugefügt, z.B. dass nur eine Log-Datei mit vorzunehmenden Änderungen erstellt wird, anstatt das Projekt an sich zu synchronisieren.

Zum Ende der Woche hin habe ich damit begonnen, die Präsentation des Projekts zu erstellen, die am 27.01. durchzuführen war.

### **Woche 18: 23.01. – 27.01.06**

In dieser Woche habe ich Bugs aus dem Projekt beseitigt und weitere Detailanpassungen eingebracht, die Kollegen bei Tests aufgefallen sind. Sekundär recherchierte ich für meine Bachelorarbeit (die nichts mit dem Praktikumsthema zu tun hat), um einen ersten Überblick darüber zu gewinnen.

Am 27.01. folgte die Präsentation der Software und damit einhergehend die Erstellung einer Liste zu bearbeitender Aufgaben und anzupassender Funktionalitäten.

### **Woche 19: 30.01. – 03.02.06**

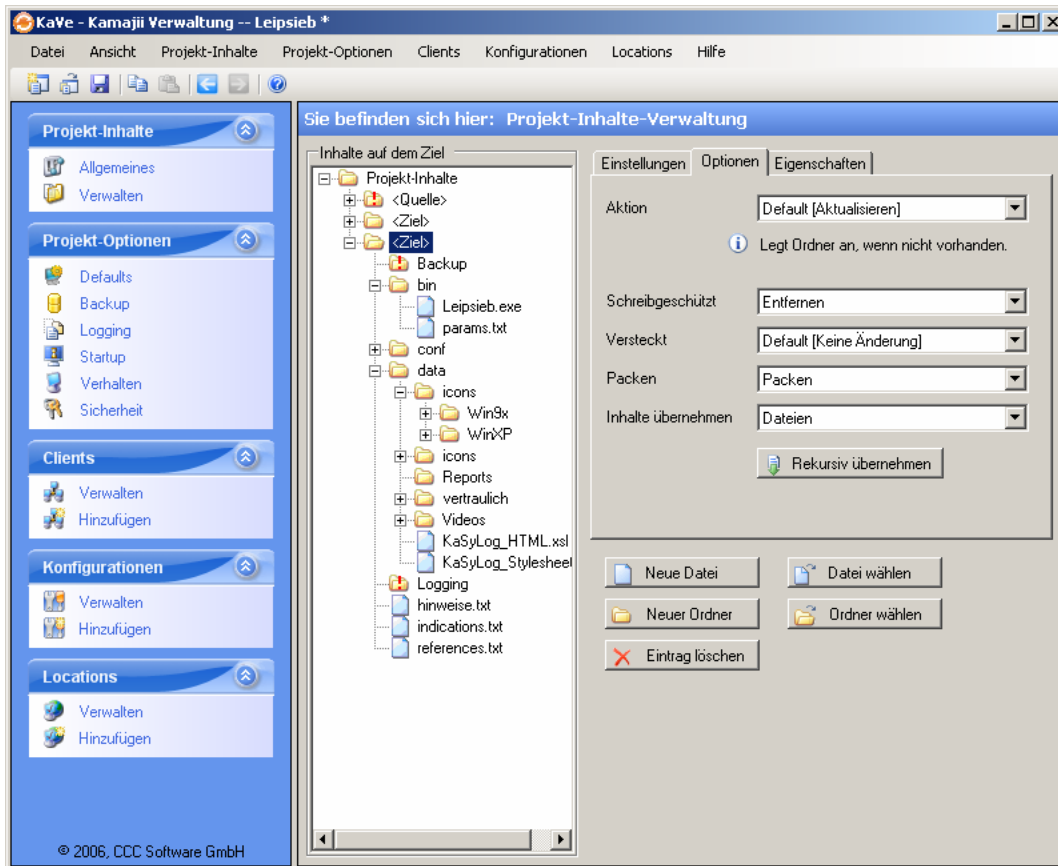
Als wichtiger Schritt dieser Woche ist die Umsetzung des Protokolls der Praktikumspräsentation zu nennen. Im Anschluss daran natürlich Tests, Fehlerbeseitigungen und das Anpassen von Quelltext-Dokumentation und Hilfen. Ende der Woche begann ich mit der Erstellung und Implementierung von XSL-Stylesheets für die erstellten Log-Dateien, auch um die Log-Dateien automatisch durch KaSy formatieren zu können.

### **Woche 20: 06.02. – 10.02.06**

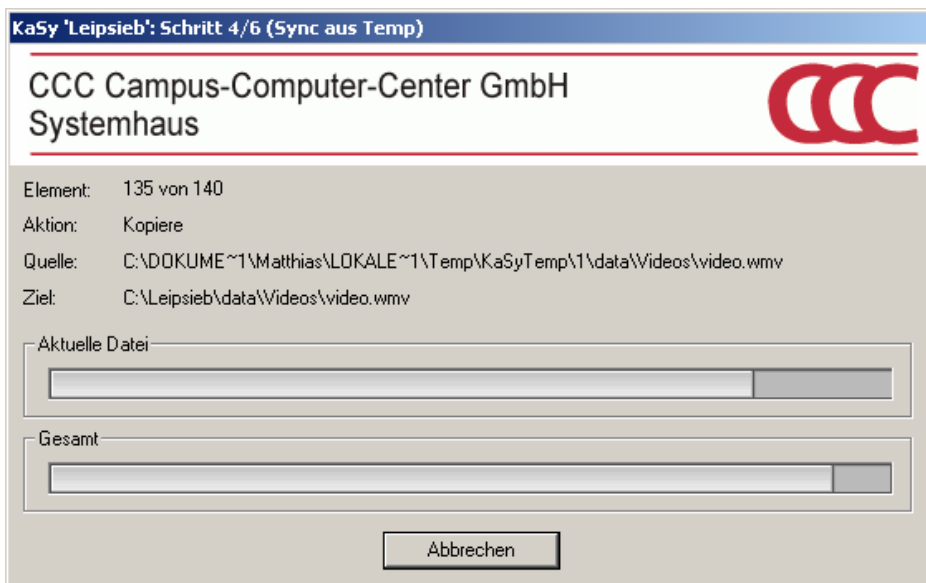
Anfang der Woche stand die Umstellung einer ersten firmen-internen Software auf das neue Produkt im Mittelpunkt, um den Mitarbeitern automatisch die aktuellste Version zu überspielen. Dies lief nahezu reibungsfrei. Weitere Tests des Programms lieferten weitere Detailverbesserungen und das Beseitigen letzter erkannter Fehler. Freitag fanden dann die Projektübergabe an Markus und letzte Verfeinerungen statt.

## Screenshots der endgültigen Versionen:

KaVe:



KaSy:



## **Persönliche Einschätzung des Praktikums:**

Ich kann für mich festhalten, dass das Praktikum zu meiner vollsten Zufriedenheit verlaufen ist. Im fast familiären Kreise der kleinen Entwicklergemeinschaft von CCC zu arbeiten hat mir viel Freude bereitet und umfangreiche Erfahrungen vermittelt.

Die Betreuung in der Firma verlief großartig: ich konnte mich jederzeit mit Fragen an Markus, meinen Betreuer, wenden und bekam sie beantwortet. Das hat mir bei vielen Problemen sehr geholfen. Doch nicht nur auf Markus, auch auf die anderen Mitarbeiter konnte man stets mit konkreten Fragen zugehen, ohne dass man abgewiesen wurde. Der exzellenten Planung von Markus und seinem Verständnis für das Projekt ist es auch zu verdanken, dass die Programmierung überhaupt ein definiertes Ziel bekam, Änderungen sich im Rahmen hielten, Erweiterungen schnell hinzuzufügen waren und das Projekt schließlich erfolgreich beendet werden konnte.

Das Praktikum bei CCC hat mir eine Menge Wissen vermittelt, das ich nicht missen möchte. Neben dem speziellen Wissen zu den Programmiersprachen C++ und C# sind da besonders auch die allgemeinen Konzepte des Programmierens zu nennen, die Planung der Entwicklung, das Testen der Anwendung. Vor allem zum Testen kann ich sagen, dass ich es hätte schneller durchführen können, wenn ich mir persönlich ein besseres Konzept erstellt hätte. In diesem und anderen Bereichen habe ich Erfahrungen gesammelt und aus „Fehlern“ gelernt, sodass ich versuchen kann es das nächste Mal besser zu machen.

Zum Schluss möchte mich noch einmal bei meinem Betreuer Markus Maspfuhl, dem Geschäftsführer Jens Heinrich sowie allen Mitarbeitern der Campus-Computer-Center GmbH für die Zeit des Praktikums und das entgegen gebrachte Vertrauen bedanken und hoffe, auch in Zukunft – vor allem im Rahmen der anstehenden Bachelor-Arbeit – auf eine fruchtbare Zusammenarbeit.

---

Matthias Jauernig,  
Praktikant

---

Markus Maspfuhl,  
Praktikumsbetreuer CCC