

Colibri: ein SAS-Server für validierte Statistiksoftware

Andreas Mangold
HMS Analytical Software GmbH
Rohrbacherstr. 26
69115 Heidelberg
andreas.mangold@hms-heidelberg.de

Zusammenfassung

Wie schafft man es, schnell und flexibel SAS-Programme zu entwickeln und einzusetzen und dabei die hohen Qualitätsanforderungen in der pharmazeutischen und medizintechnischen Industrie einzuhalten? Dieser Beitrag stellt das praktisch einsetzbare Werkzeug „Colibri“ vor, das nicht nur die validierte Entwicklung von Programmen und auch die Verwaltung von Analysen im regulierten Umfeld unterstützt, sondern mit dem man auch einfache oder komplexe SAS-Programme ohne Java-Programmierung ins Web bringen kann.

Schlüsselworte: Computer Software Validierung, 21 CFR Part 11, Makro, Web, Versionsverwaltung

1 Einleitung

Produkte der Pharma- und Gesundheitsbranche müssen ohne Kompromiss äußerst hohe Anforderungen hinsichtlich Wirksamkeit und Freiheit von Nebenwirkungen sicher erfüllen, da sie am Menschen angewandt werden. Für die Zulassung von solchen Produkten fordern die entsprechenden Behörden daher Qualitätsnachweise: die Produkte müssen diese Anforderungen nicht nur erfüllen, sondern die Unternehmen müssen dies auch beweisen können. Dies wirkt sich nicht zuletzt auf die Entwicklung und Anwendung von Software aus, auch von Software zur Datenanalyse, die beim Qualitätsnachweis für Pharma- und Gesundheitsprodukte eine große Rolle spielt.

HMS hat unter dem Namen „Colibri“ auf Basis von SAS[®] eine neuartige Anwendungsplattform für Hersteller von Pharma- und Gesundheitsprodukten entwickelt. Colibri unterstützt die systematische Validierung von SAS-Programmen und verwaltet Versionen und Konfigurationen, so dass jederzeit nachweisbar ist, mit welchen Programmen welche Ergebnisse erzielt wurden. Da alle Analysen auf dem Server gespeichert werden, kann sichergestellt werden, dass die Analyseergebnisse nicht ohne Nachweis geändert werden und nach Freigabe gar nicht mehr änderbar sind.

Der vorliegende Beitrag beschäftigt sich im Kapitel 2 mit den Herausforderungen, die man bei der Entwicklung und Anwendung von SAS-Programmen im regulierten Umfeld hat und zeigt Lösungen auf. Kapitel 3 stellt das Rollenkonzept von Colibri vor – Entwickler, Tester, Administrator und Anwender. Die Kapitel 4 bis 7 zeigen die Arbeitsweise der verschiedenen Benutzerrollen mit Colibri. Kapitel 8 gibt einen Einblick in den Aufbau von Colibri.

2 Analytical Software für Pharma / Health

2.1 Herausforderungen und Lösungen – Übersicht

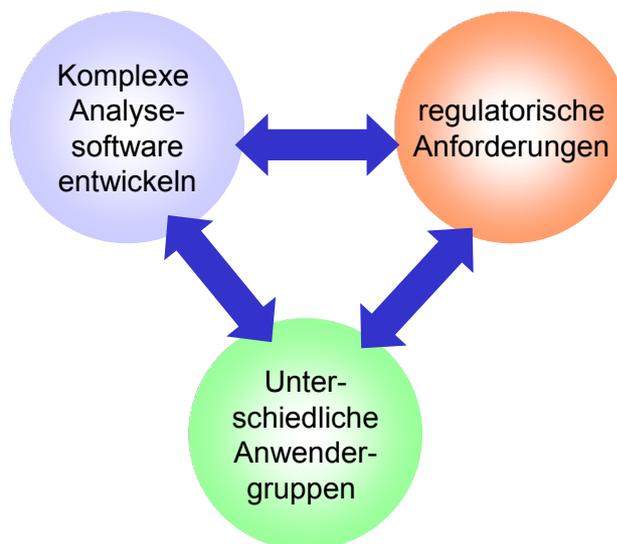


Abbildung 1: Im Spannungsfeld verschiedener Anforderungen

Die größte Herausforderung ist sicherlich, allen in Abbildung 1 genannten Anforderungen gleichzeitig gerecht zu werden, d.h. in kurzer Zeit komplexe Software zu entwickeln, dabei allen regulatorischen Anforderungen gerecht zu werden und die Bedürfnisse unterschiedlicher Anwendergruppen im Auge zu behalten.

In den folgenden Abschnitten werden die drei in Abbildung 1 genannten Problemkreise genauer betrachtet, die wesentlichen Herausforderungen genannt und Lösungsansätze skizziert, die in die Konzeption von Colibri eingeflossen sind.

2.2 Komplexe Analytische Anwendungen entwickeln

Beispiele für analytische Softwareanwendungen in der Pharma- und Gesundheitsbranche sind:

- Berechnung von Dosis-Wirkungs-Kurven in der Präklinik und Toxikologie,
- Auswertung von klinischen Studien,
- Kalibrierung von diagnostischen Messinstrumenten,
- Qualitätsüberwachung in der Produktion.

Die SAS-Software ist bewährt und bietet Werkzeuge für alle erdenklichen Analyseverfahren an. Kennzeichnend für die oben genannten Anwendungsbereiche sind aber auch folgende Punkte:

- Programme auf der Basis von SAS müssen jeweils maßgeschneidert entwickelt werden und gehen über den reinen Prozeduraufruf weit hinaus.
- Die Softwareentwicklung muss mit der Produktentwicklung stets Schritt halten. Programme müssen daher häufig angepasst werden.

Schon aus diesen Gründen ist ein systematisches Softwareengineering mit Versions- und Konfigurationskontrolle und dokumentierten Tests unabdingbar.

2.3 Hohe Qualitätsanforderungen (21 CFR Part 11)

Um medizinische Produkte in den USA vertreiben zu dürfen, müssen die Entwicklungsverfahren und Produktionsanlagen den Bestimmungen der FDA (U. S. Food and Drug Administration) entsprechen. Insbesondere muss die eingesetzte Software dem Bundesgesetz „Title 21 CFR Part 11“ von 1997 entsprechen. Im Allgemeinen wird davon ausgegangen, dass eine Einhaltung der strengen US-amerikanischen Gesetze auch dazu führt, dass entsprechende Produkte in anderen Teilen der Welt problemlos vermarktet werden können.

Das Ziel dieses Gesetzes ist es, elektronischer Aufzeichnungen gleichwertig zu schriftlichen Aufzeichnungen zu machen und festzulegen, was bei der Erstellung und Verarbeitung, bei der Speicherung und beim Abruf solcher elektronischen Aufzeichnungen zu beachten ist.

In der folgenden Tabelle 1 sind die zentralen Forderungen dieses Gesetzes und die Unterstützung, die Colibri jeweils bietet, dargestellt.

Tabelle 1: Zentrale Forderungen der FDA und Lösungsmöglichkeiten in Colibri

Forderung der FDA	Lösung in Colibri
Validierte Software: Nachweis, dass Software auch das tut, was sie tun soll	Verwaltung von Referenzen von jedem Softwaremodul auf die zugehörige Spezifikation (außerhalb von Colibri), Dokumentation von Testspezifikationen und Tests, Freigabe durch elektronische Unterschrift, Verwaltung von Versionen und Konfigurationen
Daten müssen dauerhaft gespeichert werden und jederzeit abrufbar sein	Zentrale Speicherung aller Auswertungsschritte und -ergebnisse auf dem Server, Zugriff jederzeit wieder möglich, Absicherung durch Prüfsummen
Zugriffsbeschränkung auf autorisierte Personen	Anmeldung mit Passwort, Rollenkonzept, strukturierte Zugriffsberechtigungen über Gruppenzuordnung
Nachvollziehbarkeit von Änderungen	Alle Schritte, die zu einem bestimmten Auswertungsergebnis geführt haben, bleiben nachvollziehbar, auch die jeweils verwendete Softwareversion. Manuelle Änderungen an Daten (Fehlerkorrekturen, Eliminieren von Ausreißer) werden als Audit Trails dokumentiert, Versionsverwaltung für Programme
Elektronische Unterschrift für die Freigabe / Genehmigung	Auswertungsergebnisse können, entsprechende Benutzerrechte vorausgesetzt, schreibgeschützt werden. Für bestimmte Schritte kann die Eingabe des Passworts vorgeschrieben werden

2.3 Viele Anwender mit unterschiedlichen Anforderungen

Auswertesoftware wird in unterschiedlichen Unternehmensbereichen und in unterschiedlichen Abteilungen eingesetzt. Dabei gilt es, vielen unterschiedlichen Anforderungen kostengünstig gerecht zu werden.

Die folgende Tabelle 2 enthält die Herausforderungen, die sich dadurch ergeben und zeigt, welche Lösungen Colibri hierfür bietet.

Tabelle 2: Probleme und Lösungen im Multi-user-Betrieb

Herausforderung	Lösung in Colibri
Viele Projekte bei verschiedenen Produkten in unterschiedlichen Produktphasen	Breiter einheitlicher Softwareeinsatz spart Kosten. Das Konzept der Colibri-Applikation ermöglicht es, Colibri für die unterschiedlichsten Auswertungsaufgaben zu konfigurieren. Die Wiederverwendung von Softwaremodulen in unterschiedlichen Applikationen wird unterstützt
Unterschiedliche Benutzertypen	Frei kombinierbare Auswertungsbausteine für Experten, fertig vorkonfigurierte einfach bedienbare Vorlagen für Laboranwender
Unterschiedliche Standorte (im Hause, bei Kliniken, Partnern, Homeoffice)	Aufruf aller Funktionen über Intranet oder Internet, für das Frontend wird nur ein Webbrowser benötigt
Unterschiedliche Rollen in der Softwareentwicklung	Für jede Rolle (Entwickler, Tester, Administrator, Anwender) gibt es eine Benutzeroberfläche, zusammengehalten durch ein zentrales Repository

3 Der Software-Lebenszyklus in Colibri

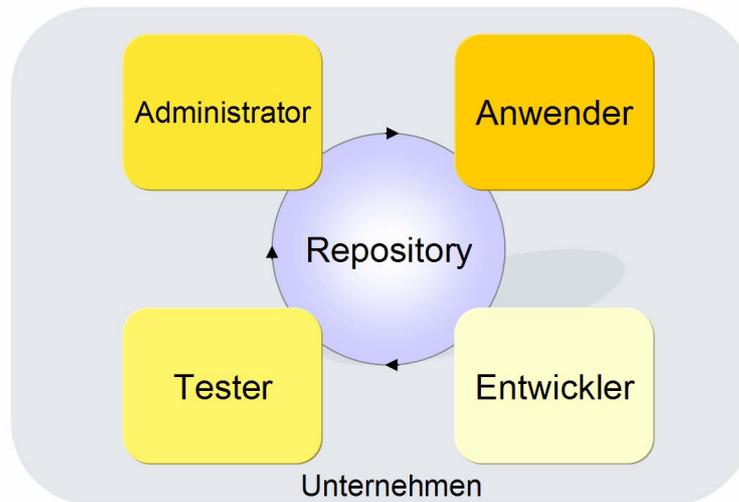


Abbildung 2: Der Software-Lebenszyklus in Colibri

Colibri unterstützt den kompletten Software-Lebenszyklus für analytische Softwaremodule: Die Entwicklung, den Test, den Programmeinsatz und die Anwendung einschließlich der Speicherung von Ergebnissen. Jeder Mitarbeiter arbeitet im Webbrowser mit einer Benutzeroberfläche, die auf seine Rolle zugeschnitten ist und die ihm genau die Daten und Funktionen zur Verfügung stellt, die er für seine Aufgabe benötigt.

Den Entwicklern nimmt Colibri viel Arbeit ab: Es verwaltet die Softwaremodule, erstellt die Benutzeroberfläche für den komfortablen Aufruf der Funktionen und unterstützt die Teamarbeit. Die Testumgebung von Colibri verringert den Aufwand, Tests zu spezifizieren, umzusetzen und zu dokumentieren. Der Administrator kann die mit Colibri entwickelten Analyse-Anwendungen einfach für diejenigen Benutzer freischalten, die sie benötigen. Eine Installation an den Arbeitsplätzen ist nicht erforderlich. Die Anwender können von jedem Standort aus sicher über das Firmennetz oder das Internet mit ihren Colibri-Workspaces arbeiten. Colibri organisiert die Ablage der Analyseergebnisse und sichert sie gegen unerwünschte Änderungen.

Das zentrale Repository ermöglicht diese durchgängige Unterstützung, indem es Informationen über alle Softwaremodule und über die Benutzer und deren Rechte speichert. Dies erleichtert es dem Unternehmen, bessere Software in kürzerer Zeit zu entwickeln und die Forderungen der FDA hinsichtlich Softwareentwicklung und Datenspeicherung zu erfüllen.

4 Colibri für den Anwender

4.1 Colibri-Funktionen komfortabel aufrufen und parametrieren

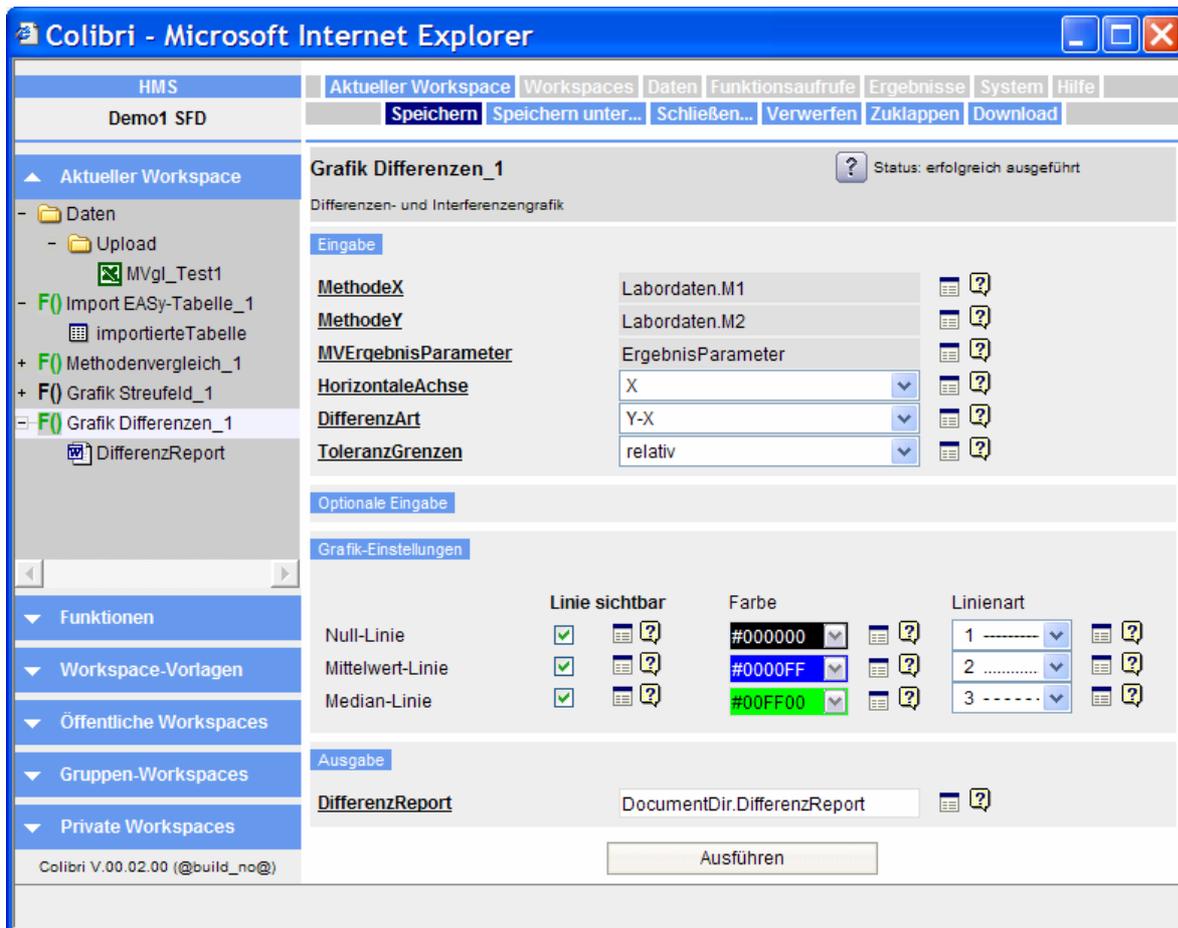


Abbildung 3: Aufruf einer Colibri-Funktion in der Anwenderoberfläche

Der Anwender wählt eine Colibri-Funktion (letztlich also ein SAS[®] Programm) aus, verbindet sie mit Daten und stellt die notwendigen Parameter ein.

Die Eingabedaten können entweder per Upload auf den Server geschrieben werden (wie im Beispiel eine Datei im Microsoft Excel Format) oder es kann auf Daten zugegriffen werden, die sich auf Datenbankservern oder auf Fileservern im Netz befinden.

Die Aufrufreihenfolge der verschiedenen Colibri-Funktionen ist im linken Bereich des Bildschirms (aktueller Workspace) in einer Baumstruktur sichtbar. Dort sieht

man auch, welche Colibri-Funktion welches Ergebnis erzeugt hat. Eine Farbcodierung zeigt, welche Aufrufe erfolgreich waren, welche noch parametrisiert oder ausgeführt werden müssen und wo ein Fehler aufgetreten ist. Der Ablauf der verschiedenen Aufrufe ist streng sequenziell. Das hat den Vorteil, dass jederzeit nachvollziehbar ist, wie die Ergebnisse zustande kommen.

Da in den Unternehmen für viele Aufgaben fest stehende Abläufe vorgeschrieben werden, ist es auch möglich, mit Vorlagen zu arbeiten, die nur noch mit Daten versorgt werden müssen.

4.2 Ergebnisse ansehen

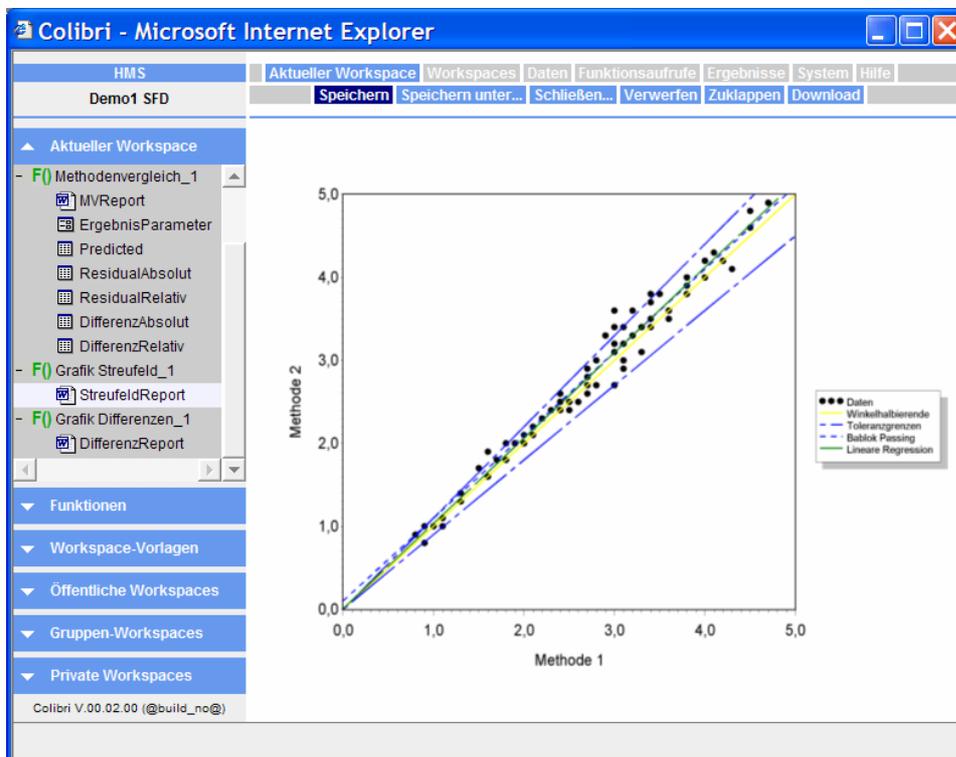


Abbildung 4: Ergebnisse einer Colibri-Funktion ansehen

Nachdem die Colibri-Funktion ausgeführt wurde, wird das Ergebnis angezeigt (Abbildung 4). Das Format der Ausgabe hängt von dem jeweiligen Programm ab und kann normalerweise direkt im Browser angezeigt werden.

HTML-Dateien und Grafiken können vom Browser ohne weiteres angezeigt werden. Ausgabeformate, für die spezifische PC-Anwendungen benötigt werden, können in der Regel auch im Browser angezeigt werden, z.B. Dateien im Format von Microsoft

Word und von Microsoft Excel oder Dateien im Adobe Portable Document Format (PDF).

Erzeugt die Colibri-Funktion eine SAS[®] Tabelle oder neue Spalten in einer vorhandenen SAS[®] Tabelle, so können diese in einem Tabellenviewer angesehen werden.

Alle Ergebnisse verbleiben normalerweise auf dem Server und werden nur zur Anzeige geöffnet. Bei Bedarf können die Ergebnisse aber auch herunter geladen und auf dem PC des Anwenders weiter verarbeitet werden.

4.3 Analysen im „Workspace“ verwalten

Alle zusammenhängenden Analyseschritte des Anwenders, also alle Datenzugriffe, Aufrufe von Colibri-Funktionen, Parametrierungen und alle Analyseergebnisse werden in einem „Colibri-Workspace“ gespeichert. Man kann sich den Workspace wie eine Arbeitsmappe vorstellen.

Im Beispiel in Abbildung 5 lädt der Anwender eine Excel-Datei hoch und speichert diese damit im Workspace auf dem Server. Danach ruft er die Colibri-Funktion „ImportExcel“ auf und gibt die Excel-Datei und das zu importierende Rechenblatt als Eingabe an. Als Resultat wird eine SAS[®] Tabelle erzeugt, die wieder im Workspace abgelegt wird.

Anschließend berechnet der Anwender mit Hilfe der Colibri-Funktion „Regression“ eine lineare Regression zweier Spalten. Die Ergebnisse, ein Report als HTML-Datei sowie die beiden Kennwerte „Steigung“ und „Intercept“ (Y-Achsen-Abschnitt) werden wieder im Workspace abgelegt.

Schließlich werden die Datenpunkte als Scatterplot ausgegeben und die Regressionsgerade mit der berechneten Steigung darin angezeigt.

Dieses Beispiel zeigt, wie die Colibri-Funktionen wie Bausteine kombiniert werden können. Die Aufteilung der Funktionalität auf die Colibri-Funktionen ist Sache des Entwicklers. Beispielsweise könnten die beiden Funktionen „Regression“ und „Scatterplot“ auch in einer Funktion vereinigt werden.

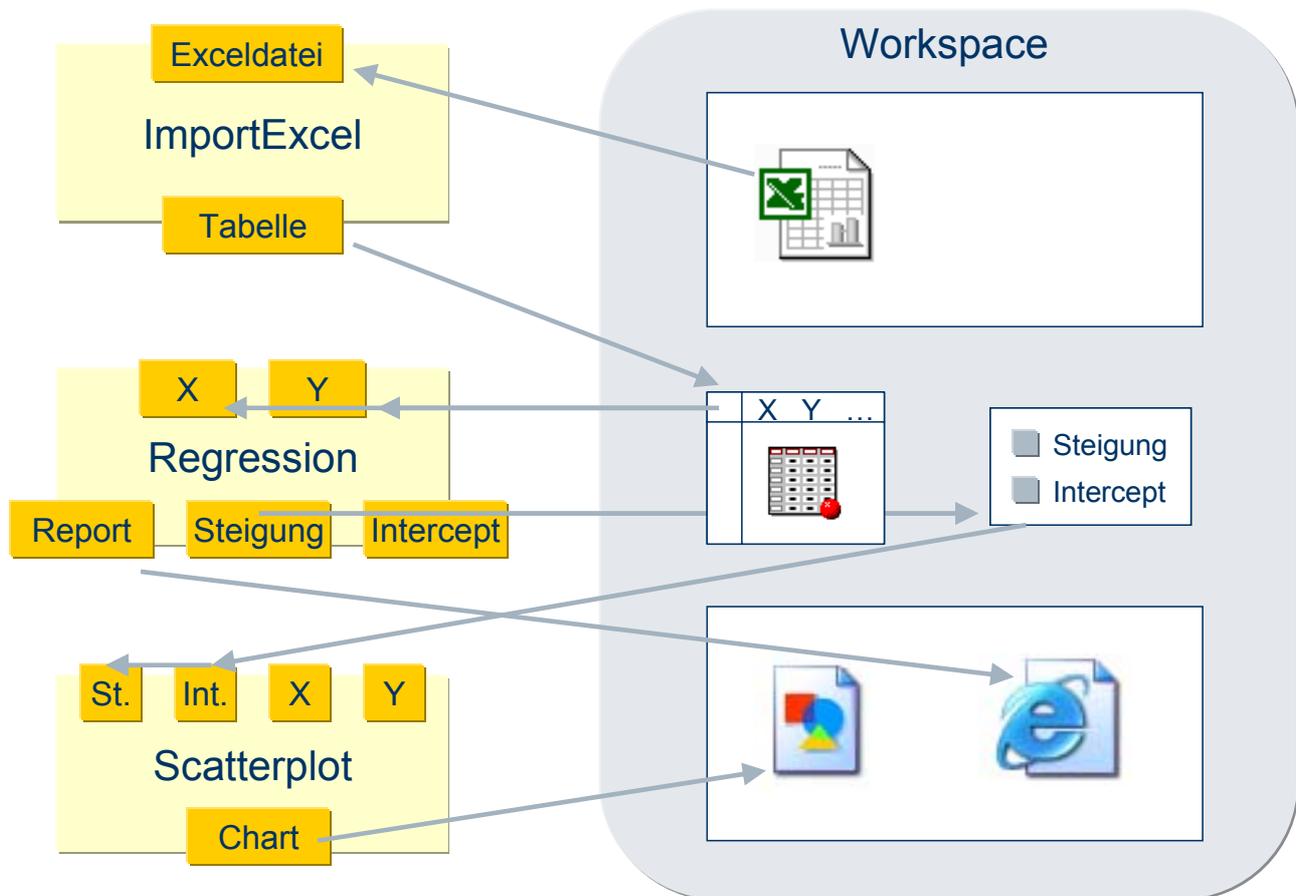


Abbildung 5: Das Workspace-Konzept von Colibri

4.4 Analysen sicher speichern und ablegen

Jeder Workspace erhält einen Namen und wird in einer Bibliothek abgelegt. Es gibt öffentliche Bibliotheken und gruppenspezifische Bibliotheken für die gemeinsame Ablage sowie private Bibliotheken für die private Ablage. Öffentliche Bibliotheken können mit einer Ordnerstruktur versehen werden, so dass zum Beispiel Workspaces, die zu einem bestimmten Produkt, zu einem bestimmten Labor etc. gehören, gemeinsam abgelegt werden können. Eine Suchfunktion ergänzt diese Systematik, so dass man Workspaces lokalisieren und jederzeit wieder öffnen kann.

Workspaces, die nicht mehr geändert werden dürfen, können mit einem Schreibschutz versehen werden. Man kann sie dann nicht mehr ändern, aber noch anschauen und unter einem anderen Namen speichern.

Da Colibri eine serverbasierte Anwendung ist, muss man damit rechnen, dass die Verbindung zum Server auch einmal gestört sein kann. Daher werden die aktuellen Inhalte des Workspaces, der gerade bearbeitet wird, auf dem Server zwischengespei-

chert. Sollte die Verbindung einmal verloren gehen, meldet sich der Anwender neu an und kann seine Arbeit einfach fortsetzen.

5 Colibri für den Entwickler

5.1 Softwarekonfigurationen und –versionen verwalten

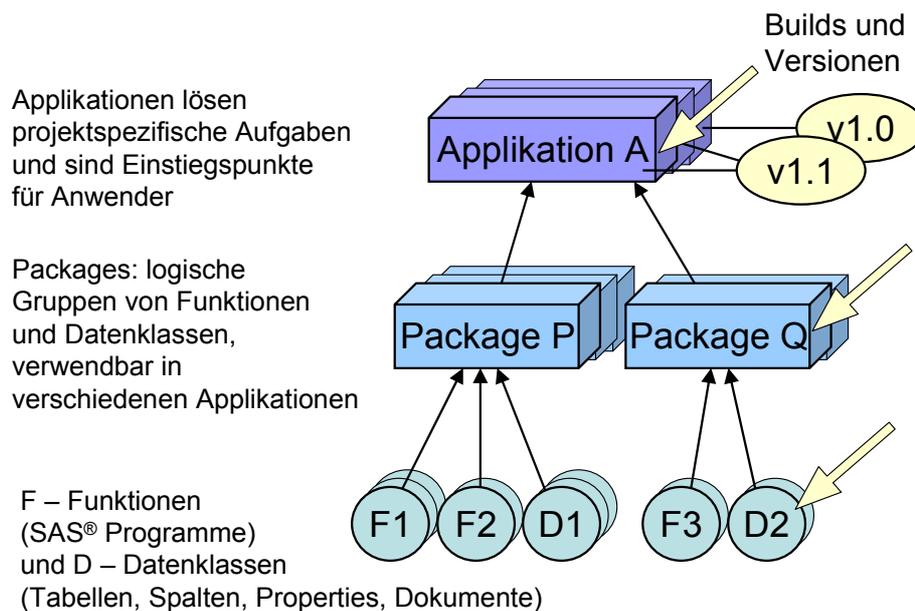


Abbildung 6: Konfiguration und Versionsverwaltung

Der Anwender arbeitet in Colibri mit Applikationen. Applikationen lösen spezifische Aufgaben, z.B. bei der Auswertung von Studien oder bei der Analyse von Daten aus der Produktion oder der Qualitätskontrolle.

Um die Wiederverwendung von Softwaremodulen zu vereinfachen, werden SAS® Programme (genannt „Colibri-Funktionen“) und Datenstrukturen (genannt „Datenklassen“) zu Packages zusammengefasst. Jedes Package kann in verschiedenen Applikationen verwendet werden. Es gibt allgemein verwendbare Packages mit grundlegenden Funktionen z.B. für den Datenimport, die Datenumformung oder für die Erstellung von Grafiken. Daneben werden für jede Applikation spezifische Packages entwickelt, die die erforderlichen Algorithmen für die jeweilige Aufgabe enthalten.

Wurde ein Softwaremodul (Applikation, Package, Funktion) einmal für die Anwendung freigegeben, erzeugt jede Änderung automatisch ein neues Build. Grundsätzlich kann kein freigegebener Programmcode überschrieben werden. Auch nach Jahren

können früher durchgeführte Berechnungen mit der gleichen Konfiguration nochmals durchgeführt werden.

5.2 Arbeiten im Team

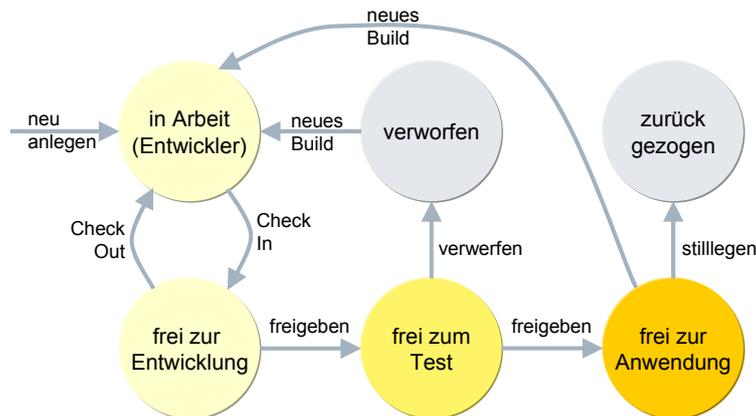


Abbildung 7: Lebenszyklus eines Softwaremoduls

Die Entwicklung im Team wird durch ein eingebautes Revision Control System unterstützt. Ein Programm kann einem Entwickler zugeordnet werden („Check Out“), der für die Fertigstellung verantwortlich ist. „Check In“ stellt das Programm anderen Entwicklern für die weitere Bearbeitung zur Verfügung.

Sobald ein Entwickler das Programm zum Test freigegeben hat, tritt der Tester in Aktion. Ergibt der Test, dass das Programm noch fehlerhaft ist oder nicht der Spezifikation entspricht, wird das Build verworfen und ein neues Build erzeugt. Wenn zwei Tester entschieden haben, dass das Programm in Ordnung ist, geben sie es zur Anwendung frei (Vier-Augen-Prinzip). Der Administrator entscheidet nun, wer mit der Anwendung arbeiten darf.

Wenn nötig können mehrere Versionen einer Anwendung parallel betrieben werden. Nicht mehr benötigte Versionen und Applikationen können stillgelegt werden.

Dieses Verfahren wird auf allen Ebenen der Konfiguration angewandt, also für Applikationen, Packages, Colibri-Funktionen und Datenklassen.

Alle Statusänderungen werden im Hintergrund protokolliert. Dadurch ist jederzeit nachvollziehbar, wer zu welchem Zeitpunkt an welcher Version des Programms gearbeitet hat. Änderungs- und Freigabekommentare vervollständigen die Dokumentation des Entwicklungsprozesses.

5.3 SAS[®] Code verwalten

Der Entwickler verwaltet den Quellcode seiner SAS[®] Programme in Colibri. Die Programme werden im Repository gespeichert.

In jedem Programm gibt der Entwickler durch entsprechende Makroaufrufe an, welche Datenstrukturen als Eingabe erwartet werden und welche Ergebnisse als Ausgabe bereitgestellt werden. So wird es möglich, den Programmablauf in Teilschritte zu zerlegen, die der Anwender später wie Bausteine zusammenfügen und parametrieren kann.

Die Parameterseiten, die der Anwender im Webbrowser zu sehen bekommt, werden automatisch von Colibri aus den Angaben des Entwicklers zusammengestellt. Bei Bedarf kann das Layout auch geändert werden. Java-Programmierung ist dafür nicht notwendig.

Die Entwickleroberfläche von Colibri unterstützt bei der Verwaltung der Programmversionen und ermöglicht auch die mehrsprachige Übersetzung von Beschreibungstexten und Hinweisen.

6 Colibri für den Softwaretester

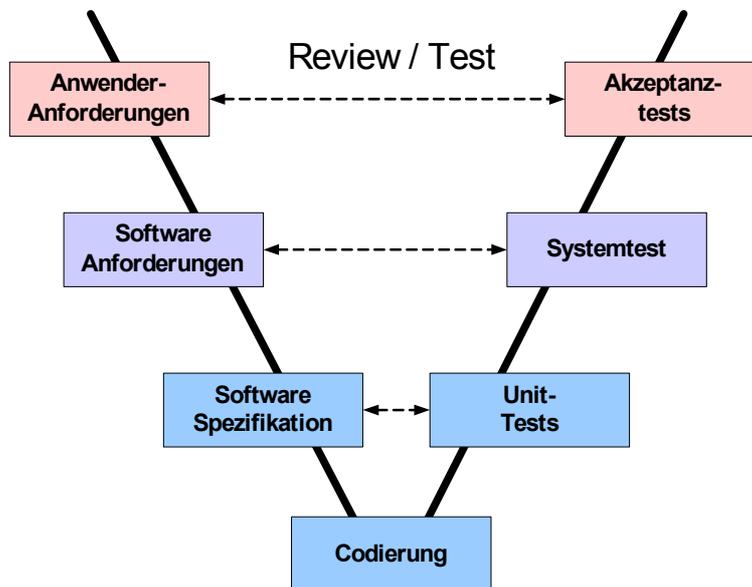


Abbildung 8: Validierung nach dem V-Modell

Für die Validierung nach dem V-Modell sind Reviews und Tests auf unterschiedlichen Ebenen erforderlich. Voraussetzung ist dabei, dass Anforderungsdefinitionen und Spezifikationen vorliegen. Die Erzeugung und Verwaltung dieser Dokumente

erfolgt außerhalb von Colibri. Colibri unterstützt aber bei den Tests. Unit-Tests sind dafür da, einzelne Colibri-Funktionen bzw. die dahinter stehenden SAS[®] Programme zu testen. Systemtests prüfen Packages und Akzeptanztests prüfen Applikationen.

Für die Testspezifikation und –durchführung bietet die Testumgebung von Colibri folgende Unterstützung: Alle Testdaten und Testaufrufe werden gemeinsam in Test-szenarien gespeichert. Der Tester dokumentiert das erwartete und das tatsächliche Ergebnis und seine Bewertung. Schließlich kann ein kompletter Testbericht ausgedruckt werden.

Wenn Programme weiterentwickelt werden, stehen die einmal erstellten Testszenarien weiterhin zur Verfügung und können ebenfalls weiterentwickelt werden. Dadurch spart man Zeit bei der Revalidierung von geänderter Software und die Testumgebung bleibt übersichtlich.

Am Ende des Testzyklus steht die Freigabe der geprüften Software. Der Administrator kann sie nun für die Anwender frei schalten.

7 Colibri für den Administrator

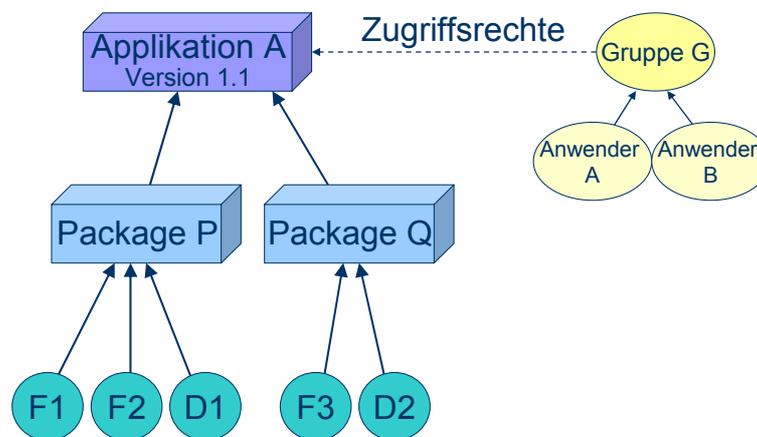


Abbildung 9: Zuordnung von Zugriffsrechten zu Applikationen

Da Colibri auf dem Server ausgeführt wird und alle Anwender im Webbrowser arbeiten, ist die Softwareverteilung sehr einfach: Der Administrator definiert Anwender und Anwendergruppen, legt ihre Rechte fest und ordnet den Gruppen diejenigen Applikationen zu, mit denen die Gruppenmitglieder arbeiten wollen. Wenn sich ein Anwender dann an Colibri anmeldet, stehen ihm automatisch alle Softwaremodule der Applikation zur Verfügung.

Auf dem PC des Anwenders muss nur ein aktueller Webbrowser installiert sein, weitere Installationsarbeiten sind nicht erforderlich. Auf diese Weise können auch Anwender außerhalb des Firmennetzes mit Colibri arbeiten, sofern ihnen der Zugriff über das Internet mit Hilfe einer sicheren Verbindung (SSL) eingeräumt wird. Weitere Aufgabe des Administrators ist die Zuordnung von SAS[®] Servern zu Anwendergruppen und Applikationen, um die Rechenlast zu verteilen. Auch der Administrator selbst arbeitet in der browserbasierten Benutzeroberfläche von Colibri und hat dort immer alle Benutzer, Gruppen und Rechte im Blick.

8 Der Aufbau von Colibri

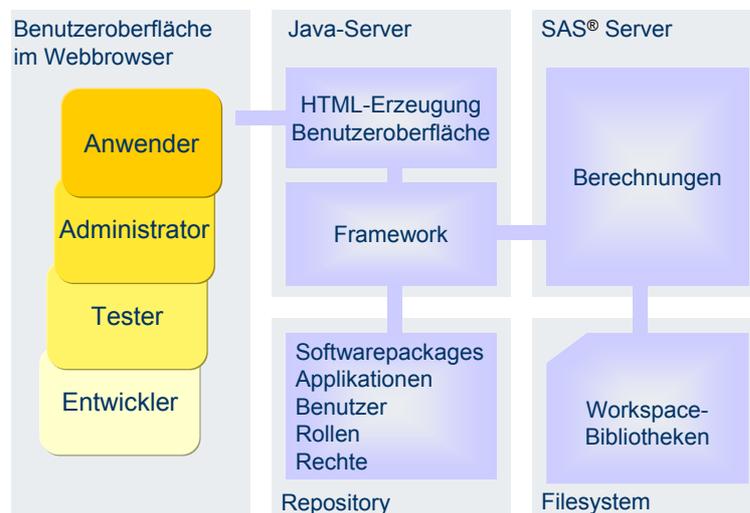


Abbildung 10: Softwarearchitektur von Colibri

Für jede Benutzerrolle gibt es in Colibri eine Benutzeroberfläche. Die Rolle wird bei der Anmeldung ausgewählt.

Die Colibri-Serverinstallation besteht aus vier Komponenten:

- Der SAS[®] Server greift auf die Workspace-Strukturen zu und nimmt die eigentlichen Berechnungen vor. Es können beliebig viele SAS[®] Server parallel eingesetzt werden, um die Rechenkapazität zu vergrößern.
- Das Repository wird auf einem relationalen Datenbanksystem gespeichert (derzeit wird Oracle bevorzugt).
- Die Workspacebibliotheken werden in einer Ordnerstruktur im Filesystem gespeichert.

A. Mangold

- Die Logik von Colibri („Framework“) läuft auf einem Java-Server. Dort laufen auch die Java-Programme, die die browserbasierte Benutzeroberfläche antreiben. Als Java-Server kann Tomcat oder ein Java-Container wie IBM Websphere eingesetzt werden.