

# Validierung von SAS-Programmen für die Auswertung und Dokumentation klinischer Daten - Prozess, Umfang und Dokumentation der Validierung

Claudia Meurer  
Accovion GmbH  
Helfmann-Park 10  
65760 Eschborn  
claudia.meurer@accovion.com

## Zusammenfassung

Jede Erstellung eines SAS<sup>TM</sup>-Programmes zur Auswertung klinischer Daten im regulatorischen Umfeld wirft die Frage auf, wie das Programm adäquat zu validieren ist und wie diese Validierung ausreichend dokumentiert wird. Bei Accovion wird in der Planungsphase jedes Projekts festgelegt, welches Validierungsmodell für die Validierung eines einzelnen SAS<sup>TM</sup>-Programmes angewendet werden soll. Dabei werden fünf Validierungsmodelle unterschieden. Die Auswahl des Validierungsmodells hängt von der Relevanz und dem zugrunde liegenden Risiko eines SAS<sup>TM</sup>-Programmes ab. Das jeweilige Validierungsmodell legt fest, wie und in welchem Umfang zu validieren ist und ob eine unabhängige Validierung durch einen zweiten Programmierer erfolgen soll.

Im Vortrag werden der Prozess der Validierung, die unterschiedlichen Validierungsmodelle und die Dokumentation der Validierung von SAS<sup>TM</sup>-Programmen beschrieben.

**Schlüsselwörter:** Validierung, SAS<sup>TM</sup>-Programm, Validierungsaufwand, Validierungsprozess, Validierungsdokumentation.

## 1 Vorstellung der Abteilung und Arbeitsinhalte

Die Firma Accovion GmbH hat in Deutschland zwei Standorte, der Hauptsitz ist in Eschborn bei Frankfurt, der zweite Sitz befindet sich in Marburg. Z.Zt. arbeiten in Eschborn 15 und in Marburg 8 SAS<sup>TM</sup>-Programmierer. Zu den Aufgaben der SAS<sup>TM</sup>-Programmierer gehören neben der Erstellung von SAS<sup>TM</sup>-Programmen zur Auswertung und Integration klinischer Daten auch die Erstellung von SAS<sup>TM</sup>-Standardauswertungsprogrammen auf der Grundlage von CDISC (Clinical Data Interchange Standards Consortium). Des Weiteren werden standortübergreifend kontinuierlich neue Tools entwickelt, die zur Arbeitserleichterung dienen, und eine gemeinsame Programmbibliothek (für SAS<sup>TM</sup>-Macros, SAS<sup>TM</sup>-Standardprogramme und Tools) gepflegt.

## 2 Validierungsmodelle

Gemäß der Accovion Standard Operating Procedure (SOP) zur Validierung von SAS<sup>TM</sup>-Programmen wird für jedes zu erstellende SAS<sup>TM</sup>-Programm das adäquate Validie-

rungsmodell in Absprache mit dem Statistiker bei Accovion sowie dem Kunden festgelegt. Die Validierungs-SOP unterscheidet 5 verschiedene Validierungsmodelle A bis E mit aufsteigendem Validierungsaufwand. Der Entwickler des SAS™-Programmes wird im Folgenden auch Erstvalidierer genannt, der unabhängige SAS™-Programmierer, der das SAS™-Programm überprüft, Zweitvalidierer.

Als Basis der Validierung stehen dem Programmierer entweder ein Spezifikationsdokument zur Erstellung eines SAS™-Datensatzes oder ein Spezifikationsdokument zur Erstellung von Tabellen/Listen/Grafiken („Table shells“) zur Verfügung. Außerdem dient der statistische Analyseplan (SAP) eines Projekts als Grundlage für die Programmierung und Validierung. Hierin ist vom Statistiker festgelegt, wie die Daten ausgewertet werden sollen.

Das Spezifikationsdokument zur Erstellung eines SAS™-Datensatzes beschreibt alle Variablen, die der Datensatz enthalten soll, deren Formate, Eigenschaften und Herleitung bzw. Berechnung. Dieses Spezifikationsdokument wird üblicherweise als Word™- oder Excel™-File erstellt.

Variable Name	Label	Role	Type / Length	Format or Controlled Terminology	Source	Derivation Rule
STUDYID	Study Identifier	Identifier	Char \$20		Derived	= "RNN263/2001"
DOMAIN	Domain Abbreviation	Identifier	Char \$2		Derived	= 'DM'
USUBJID	Unique Subject Identifier	Identifier	Char \$40		Derived	If UNIQSUBJ is available for subject, then = TRIM(UNIQSUBJ.PSTUDYID)    '/'    TRIM(UNIQSUBJ.PSITEID)    '/'    TRIM(UNIQSUBJ.PSUBJID) Otherwise, = TRIM(DEMO.STUDYID)    '/'    PUT(DEMO.SITEID,Z4.0)    '/'    PUT(DEMO.SUBJID,Z7.0)
CSUBJID	Clintrial Unique Subject ID	Identifier	Char \$9		= DEMO.CSUBJID	
SITEID	Study Site Identifier	Identifier	Char \$4		Derived from DEMO.SITEID	PUT(DEMO.SITEID,Z4.0)
INVNAME	Investigator Name	Identifier	Char \$200		Derived from INVNAME.INVFIRST, INVNAME.INVLAST	INVNAME=COMPBL(INVNAME.INVLAST    ','    INVNAME.INVFIRST)
SUBJID	Subject Identifier for the Study	Identifier	Char \$7		Derived from DEMO.SUBJID	PUT(DEMO.SUBJID,Z7.0)

**Abbildung 1:** Auszug aus dem Spezifikationsdokument für Datensätze

Bei „Table shells“ handelt es sich um Vorlagen, die Layout und Inhalt der zu erstellenden Tabellen/Listen/Grafiken beschreiben. Diese sind üblicherweise als Word™- oder pdf-File abgelegt.

**Demographic data (ITT population)**

Characteristic	Statistic	Placebo	XXX
Total subjects treated			
Sex			
Male	N (%)		
Female	N (%)		
Age			
	Mean		
	SD		
	Median		
	Min		
	Max		
< 65 years	N (%)		
>= 65 years	N (%)		
BMI (kg/m <sup>2</sup> )	N		
	Mean		
	SD		
	Median		
	Min		
	Max		
Obese (BMI > 28 kg/m <sup>2</sup> )	N(%)		

**Abbildung 2:** Auszug aus den „Table shells“**2.1 Modell A**

Der Ersteller des SAS<sup>TM</sup>-Programmes (=Erstvalidierer) überprüft sein Programm durch den Review von Programm, Log-File und dem (eventuell erstellten) Output. Die Überprüfung findet anhand der Vorgaben im Spezifikationsdokument und dem SAP sowie anhand von Plausibilitätsprüfungen des erzeugten Outputs statt. Als Hilfe dient ihm außerdem die Validierungs-Checkliste, die alle Punkte beinhaltet, auf die bei der Validierung zu achten sind. Diese Checkliste kann im Bedarfsfall jederzeit erweitert werden.

Die Anwendung von Modell A ist die Minimalvalidierung für jedes erstellte SAS<sup>TM</sup>-Programm und ist in jedem der im folgenden beschriebenen Modell impliziert.

**2.2 Modell B**

Bei diesem Validierungsmodell wird nach der Erstvalidierung (wie in Modell A beschrieben) ein unabhängiger SAS<sup>TM</sup>-Programmierer bestimmt, der den Output des

SAS<sup>TM</sup>-Programmes (SAS<sup>TM</sup>-Datei, -Tabelle, -Liste oder Grafik) auf Richtigkeit überprüft. Als Hilfe dient ihm der SAP und die Validierungs-Checkliste.

## 2.3 Modell C

Bei Modell C wird nach der Erstvalidierung (wie in Modell A beschrieben) ein unabhängiger SAS<sup>TM</sup>-Programmierer bestimmt, der zusätzlich zu Modell B neben dem generierten Output das SAS<sup>TM</sup>-Programm durch Review von Programm und Log-File überprüft. Dabei gleicht er ab, ob alle Vorgaben des Spezifikationsdokumentes berücksichtigt wurden. Außerdem wird auch für die Validierung nach Modell C der SAP und die Validierungs-Checkliste benutzt.

## 2.4 Modell D

Bei der Validierung nach Modell D wird nach der Erstvalidierung (Modell A) ein unabhängiger SAS<sup>TM</sup>-Programmierer bestimmt, der anhand des Spezifikationsdokumentes ein eigenes SAS<sup>TM</sup>-Programm erstellt. Die Ergebnisse werden mit Vergleichsprozeduren analysiert, die Diskrepanzen diskutiert und beseitigt.

## 2.5 Modell E

Dieses Validierungsmodell wird üblicherweise bei SAS<sup>TM</sup>-Macros benutzt, die für den projektübergreifenden Einsatz vorgesehen sind. Hierzu werden vorab Testszenarios definiert. Mittels Testdaten werden dann die erwarteten mit den tatsächlichen Ergebnissen verglichen.

# 3 Prozess der Validierung

Der Erstvalidierer erstellt das SAS<sup>TM</sup>-Programm auf dem SAS-Server (Unix-Rechner) in einer Entwicklungsumgebung (./pg\_dev/..). Ist die Erstvalidierung erfolgt, wird das SAS<sup>TM</sup>-Programm mittels eines Tools (Unix-Befehl) in ein anderes Unix-Verzeichnis verschoben, in die sogenannte Validierungsumgebung (./pg\_val/..). Hier ist das SAS<sup>TM</sup>-Programm nun schreibgeschützt abgelegt und für die Zweitvalidierung freigegeben.

Nachdem die unabhängige Zweitvalidierung erfolgt ist und das SAS<sup>TM</sup>-Programm für valide befunden wurde, wird es mittels eines anderen Unix-Befehls in das Unix-Verzeichnis verschoben (./pg/..), in dem alle erfolgreich validierten SAS<sup>TM</sup>-Programme abgelegt sind. Gleichzeitig wird eine Kopie des SAS<sup>TM</sup>-Programmes im Verzeichnis ./pg\_dev/./history abgespeichert. Dadurch bleiben alle Programmversionen erhalten.

Falls keine Zweitvalidierung erfolgen soll (Modell A) verschiebt der SAS<sup>TM</sup>-Programmierer nach erfolgreicher Erstvalidierung das SAS<sup>TM</sup>-Programm in das Unix-Verzeichnis, in dem alle finalen SAS<sup>TM</sup>-Programme abgelegt werden (./pg/..).

Sollte ein SAS<sup>TM</sup>-Programm nach erfolgreicher Zweitvalidierung noch einmal editiert werden, muss es mittels eines Tools (Unix-Befehl) wieder aus der endgültigen Unix-Umgebung (./pg/..) in die Entwicklungsumgebung (./pg\_dev/..) verschoben werden. Dann erfolgt die Änderung des SAS<sup>TM</sup>-Programmes (und deren Erstvalidierung), an-

schließlich die Zweitvalidierung der Änderung. Danach wird das SAS™-Programm wieder über die Validierungsumgebung in das endgültige Unix-Verzeichnis verschoben (wie oben beschrieben). Dieser Prozess stellt eine lückenlose Dokumentation der Änderungshistorie sicher.

## 4 Dokumentation der Validierung

Für die Dokumentation der Validierung stehen Templates als Excel™- oder Word™-Files zur Verfügung, die in den nachfolgenden Kapiteln beschrieben werden.

### 4.1 Validierungs-Trackingsheet (Excel™)

Dieses Excel™-sheet wird pro Projekt bzw. Studie angelegt. Hier ist für jeden zu erstellenden Datensatz bzw. für jede zu erstellende Tabelle/Liste/Grafik ein Eintrag zu machen.

Folgende Informationen werden ausgefüllt:

- SAS™-Programmname
- Name des Outputs
- Name des Erstvalidierers
- Datum der vollständigen Erstvalidierung
- Validierungsmodell (A-E)
- Name des Zweitvalidierers (bei Modell B-E)
- Datum der vollständigen Zweitvalidierung (bei Modell B-E)
- eventuell: Name des Statistikers, der den Output zusätzlich prüft, und das Datum seiner Überprüfung



Study Code: study/project

Derived Data Set or Table/Listing/Graph	Program Name	Output Name	Program Developer	Date First Validation Completed (dd-mmm-yyyy)	Validation Model (A,B,C,D,E)	Independent Validator (For Model D Document Name of Validation Program)	Date Validation Completed (dd-mmm-yyyy)	Output Review Statistician	Date Statistical Review Completed (dd-mmm-yyyy)
Adverse event	ae.sas	ads.ae	XX YY	11-Apr-04 13-May-04	C	XX1 YY1	12-Apr-04 13-May-04		
Demographics	demog.sas	ads.dm			C				
Demographics - Safety population	1dm01t.sas	1dm0001t.lst	YY	25-Jun-05	C	XX1	30-Jun-05	ZZ	15-Jul-05
Demographics - ITT population	1dm01t.sas	1dm0002t.lst	YY	25-Jun-05	B	ZZ	15-Jul-05	ZZ	15-Jul-05
Demographics - PP population	1dm01t.sas	1dm0003t.lst	YY	25-Jun-05	B	ZZ	15-Jul-05	ZZ	15-Jul-05

Abbildung 3: Auszug aus dem „Validierungs-Trackingsheet“

Nach Abschluss des Projekts bestätigen alle in die Programmentwicklung und Validierung beteiligten Programmierer die Eintragungen mit ihrer Unterschrift auf dem ausgedruckten Validierungs-Trackingsheet.

## 4.2 Validierungs-Checklisten (Word™)

Hier stehen verschiedene Checklisten zur Verfügung. Diese fassen übersichtlich zusammen, worauf bei der Validierung zu achten ist und welche Aspekte geprüft werden sollen.

Für den Review von SAS™-Programmcode sind dies zum Beispiel:

- Wurde die Accovion Guideline „Good Programming Practice“ befolgt? Diese legt u.a. den Standardkopf für das SAS™-Programm fest; sie gibt vor, wie ein SAS™-Programm strukturiert werden soll, worauf bei der Programmierung zu achten ist, etc.
- Entspricht der Programmcode den Vorgaben im SAP (Statistical Analysis Plan) und dem Spezifikationsdokument?
- Ist die Logik des Programms plausibel, werden die Variablen korrekt selektiert?
- Ist die Anzahl der Records nach jedem Data-step plausibel?

Bei der Validierung eines neu erstellten Datensatzes soll u.a. geprüft werden:

- Werden die Daten adäquat und komplett abgelegt?
- Sind die neu erstellten Variablen korrekt berechnet?
- Ist der Umgang mit fehlenden Werten korrekt umgesetzt?
- Entspricht die Reihenfolge der Variablen im Datensatz der im Spezifikations-Dokument vorgegeben Reihenfolge?
- Sind die Labels für Datensatz und Variablen korrekt definiert, sind die Formate korrekt zugewiesen?

Bei der Validierung von Tabellen/Listen/Grafiken sollen folgende Punkte überprüft werden:

- Entspricht der Output den Vorgaben im SAP (Statistical Analysis Plan) und dem Spezifikationsdokument?
- Sind Überschriften und Fußnoten korrekt? Stimmen die Spaltenüberschriften?
- Sind Variablen und Labels korrekt dargestellt und nicht abgeschnitten?
- Sind fehlende Daten richtig dargestellt?
- Sind die Seitenumbrüche korrekt?
- Ist die Grafik auf der Seite korrekt platziert?
- Sind Grafik-Achsen und Legenden richtig definiert und beschriftet?
- Stimmt die Anzahl der Patienten in den Behandlungsgruppen?
- Sind Häufigkeiten und Prozentangaben korrekt?
- und vieles mehr

Zur Überprüfung des korrekten Ablaufs eines SAS™-Programmes wird u.a. auf folgende Punkte geachtet:

- Wurde der Output mit der korrekten Version der Daten erzeugt?
- Ist das Log-File fehlerfrei?
- Warnungen im Log-File werden überprüft.
- Notes im Log-File werden überprüft („merge statements with repeats of by values“, „conversion of data types“, „generation of missing values“, „values outside axis range“, „division by zero“).
- Stehen die erstellten Outputs in den entsprechenden Verzeichnissen?

### 4.3 „Validation-Findings“-Dokument (Word™)

Im Validation-Findings-Dokument informiert der Zweitvalidierer den Erstvalidierer (=Programmersteller) über die notwendigen Änderungen im SAS™-Programm. Hier können aber auch Fragen oder Probleme dokumentiert werden, die dann mit dem zuständigen Statistiker diskutiert werden.

<b>Program name: ae.sas</b>		
<b>Variable</b>	<b>Comment</b>	<b>Resolution</b>
<b>SEXCD</b>	<i>Variable is called SEXC in the specification document. (program line 54) Format for SEX is not assigned. ddmmyyyy/program validator</i>	<i>Specification revised ddmmyyyy/program developer</i>
<b>CSUBJID</b>	<i>Variable has length of 12 instead of 9. ddmmyyyy/program validator</i>	<i>Program revised ddmmyyyy/program developer</i>
<b>definition of PERIOD</b>	<i>PERIOD is set to missing, open issue in SAP, has to be discussed with statistician ddmmyyyy/program validator</i>	<i>No changes necessary, PERIOD can be left missing ddmmyyyy/statistician</i>

**Abbildung 4:** Auszug aus dem „Validation-Findings“-Dokument

Zur Dokumentation werden außerdem die von den unabhängigen SAS™-Programmierern erstellten Programme bzw. Hilfslisten (proc print, proc freq, proc means ...) abgelegt.

## 4.4 Dokumentation der Re-Validierung

Ist ein SAS™-Programm bereits erfolgreich erst- und zweitvalidiert, muss aber danach noch einmal geändert werden, findet eine Re-Validierung statt.

Diese wird wie folgt dokumentiert:

- Der Erstvalidierer macht die notwendigen Änderungen im SAS™-Programm. Im Programmkopf ergänzt er eine Zeile unter dem Punkt „Modification History“. Hier dokumentiert er das Datum, seinen Namen und die Art der Änderung, die gemacht wurde.
- Im Validierungs-Trackingsheet wird vom Erstvalidierer zu dem geänderten SAS™-Programm ein Record mit dem aktuellen Datum ergänzt (siehe 4.1, Adverse event Datensatz).

## 5 Zusammenfassung

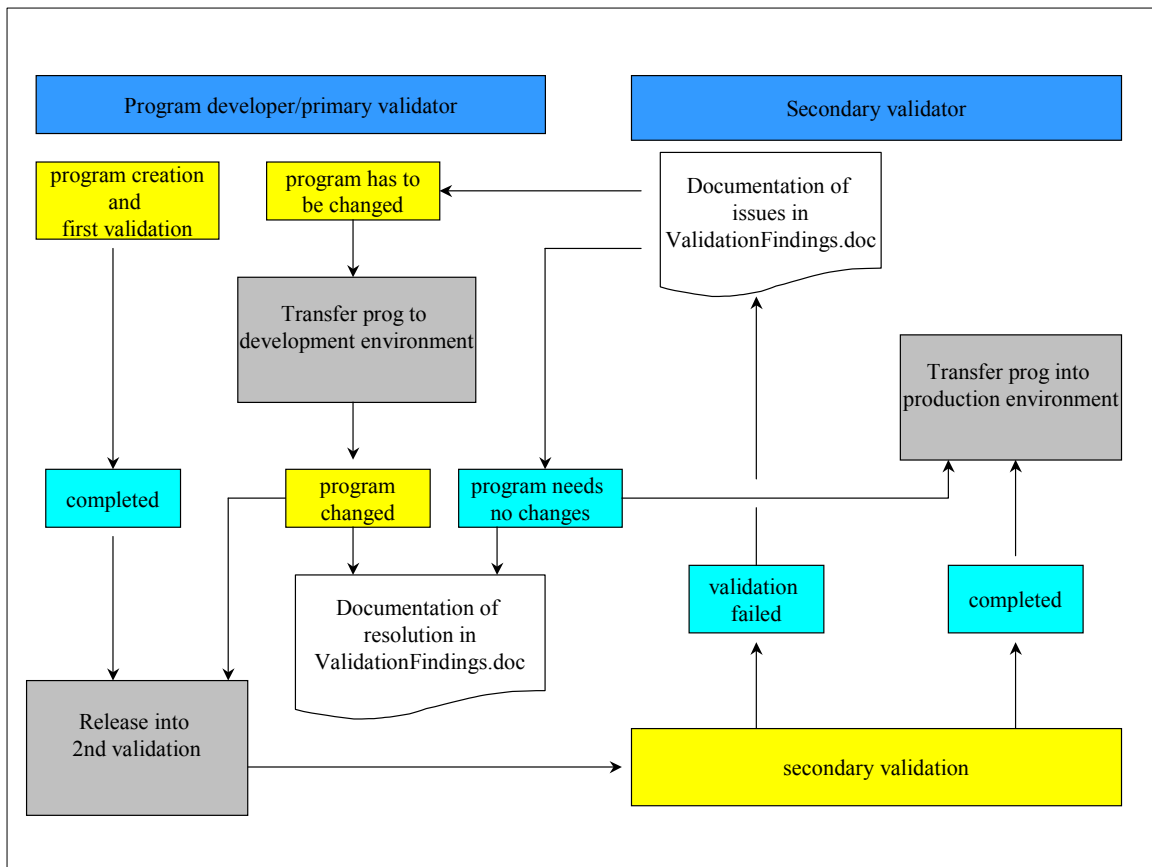


Abbildung 5: Ablauf des Validierungsprozesses

Der Aufwand der Validierung von SAS™-Programmen hängt davon ab, welches Modell zur Validierung gewählt wird. Unabhängig vom Validierungsmodell ist der Ablauf der Validierung gleich. Für den korrekten Ablauf der Validierung ist, neben der sorgfältigen Validierung selbst, eine disziplinierte und systematische Arbeitsweise des SAS™-Programmierers notwendig, zum einen bei dem Ausfüllen der Dokumente, zum anderen



bei dem Transfer der validierten SAS<sup>TM</sup>-Programme in die verschiedenen Unix-Verzeichnisse. Ungeachtet des Aufwandes überwiegen die Vorteile. Der standardisierte Validierungsablauf hilft dem SAS<sup>TM</sup>-Programmierer, bei der Validierung an alles zu denken. Die SAS<sup>TM</sup>-Programmierer können schnell von einem Projekt in ein anderes wechseln. Die Einarbeitung in neue Projekte verringert sich, weil der Validierungsprozess immer der gleiche bleibt. Anhand des Validierungs-Trackingsheets ist für jeden ersichtlich, wie der Status der Validierung ist, wie viele SAS<sup>TM</sup>-Programme noch erst- bzw. zweitvalidiert werden müssen. Das Validierungs-Trackingsheet dient auch als Grundlage für den Projektleiter, der den Fortgang des Projekts kontrolliert, Ressourcen plant und Timelines überwacht. Der standardisierte Validierungsprozess ermöglicht den flexiblen Einsatz der SAS<sup>TM</sup>-Programmierer in den unterschiedlichsten Projekten und das kurzfristige Einspringen in zeitkritische Projekte. Außerdem garantiert die Einhaltung des Validierungsprozesses eine hohe Qualität der Programmierung.