SAS-AMO, Stored Processes und SAS Foundation – Nutzung der SAS-BI Landschaft in der Pflanzenschutzmittelforschung

Andreas Büchse
BASF SE
Agrarzentrum
67117 Limburgerhof
andreas.buechse@basf.com

Matthias Klein
BASF SE
Agrarzentrum
67117 Limburgerhof
matthias.klein@basf.com

Grischa Pfister
iCASUS GmbH
Vangerowstr. 2
69115 Heidelberg
g.pfister@icasus.de

Zusammenfassung

In der Pflanzenschutzmittelforschung der BASF stellen Applikationen, die SAS Technologie verwenden, seit vielen Jahren einen Kern-Baustein innerhalb der statistischen Datenanalyse dar. Der SAS Enterprise BI Server integriert sich in eine Service Oriented Architecture (SOA) und passt optimal in unsere IT Strategie.
Statistische Methoden, Modelle und Workflows können mit der SAS Stored Process Technologie des SAS BI Servers bereitgestellt und sowohl in SAS als auch in 3rd party Applikationen verwendet werden. Mit dieser Architektur sind kurze Entwicklungszyklen von Statistik-Software in der Forschung möglich. Redundante Entwicklung von Statistik-Software wird vermieden und die Verwendung von standardisierten Datenanalysen in allen Applikationen ermöglicht. Das seit SAS 9 zur Verfügung stehende SAS-Add-In für MS Office Applikationen (SAS-AMO) bietet sowohl Möglichkeiten zur flexiblen Datenanalyse mit dem Methodenspektrum von SAS/Stat und SAS/Graph als auch zur Verwendung der genannten Stored Processes.
Mit dieser Kombination kann die zentrale, serverbasierte Steuerung der Datenanalyse und die Bereitstellung von Workflows und Modellen durch Experten in ein flexibles Nutzungskonzept eingebunden werden. Hierbei kann jeder Nutzer eine seinem Kenntnisstand und Aufgabengebiet entsprechende Rolle einnehmen und zwischen automatisierter oder flexibler Anwendung wählen. Neben den ohnehin genutzten Datenbanken kann MS EXCEL eine Rolle als universelles Frontend einnehmen und sowohl zum Upload der Daten als auch zur Ergebnispräsentation genutzt werden.
Der Beitrag möchte die Nutzung des SAS-AMO und der Stored Processes für die Datenanalyse in einem großen Forschungsunternehmen darstellen. Als Beispiel hierfür dient die Auswertung von Dosis-Wirkungskurven in der Wirkstoffsuchforschung.

Schlüsselwörter: SAS Enterprise BI Server, Add-In for Microsoft Office, Stored Process, Dosis-Wirkungskurve, Pflanzenschutzmittelforschung, Service Oriented Architecture

1 Dosis-Wirkungskurven als ein Beispiel für die statistische Datenanalyse in der Pflanzenschutzmittelforschung

Zwischen der Dosis einer chemischen Substanz und ihrer Wirkung auf biologische Systeme besteht in der Regel ein monotoner Zusammenhang. Bereits Paracelsus (1493-1541) formulierte: *„[...] All Ding' sind Gift und nichts ohn' Gift; allein die Dosis macht, das ein Ding kein Gift ist. [...]“*. In anderen Worten, je höher die Dosis, desto höher die Wirkung. Bei der Entwicklung neuer Pflanzenschutzmittel spielt diese Beziehung eine überragende Rolle. Ausschließlich Wirkstoffe, die bei geringer Dosis eine biologische Aktivität für den Zielorganismus (Phytopathogene Pilze, Unkräuter, Insekten) aufweisen, sind wirtschaftlich interessant. Um toxische Effekte für Nicht-Zielorganismen auszuschließen, sollte die wirksame Dosis für Mensch und Tier möglichst mehrere Potenzen höher liegen.
Wenn Wirkung und Dosis graphisch dargestellt werden, so zeigt sich im Allgemeinen eine nichtlineare Beziehung. Hierbei ist es üblich, die Dosis auf der Abzisse in logarithmischer Skalierung abzutragen. Dieses bewirkt in der Regel einen sigmoiden Verlauf der Funktion. In Abb. 1-3 und Tab. 1 sind typische Beispiele wiedergegeben.

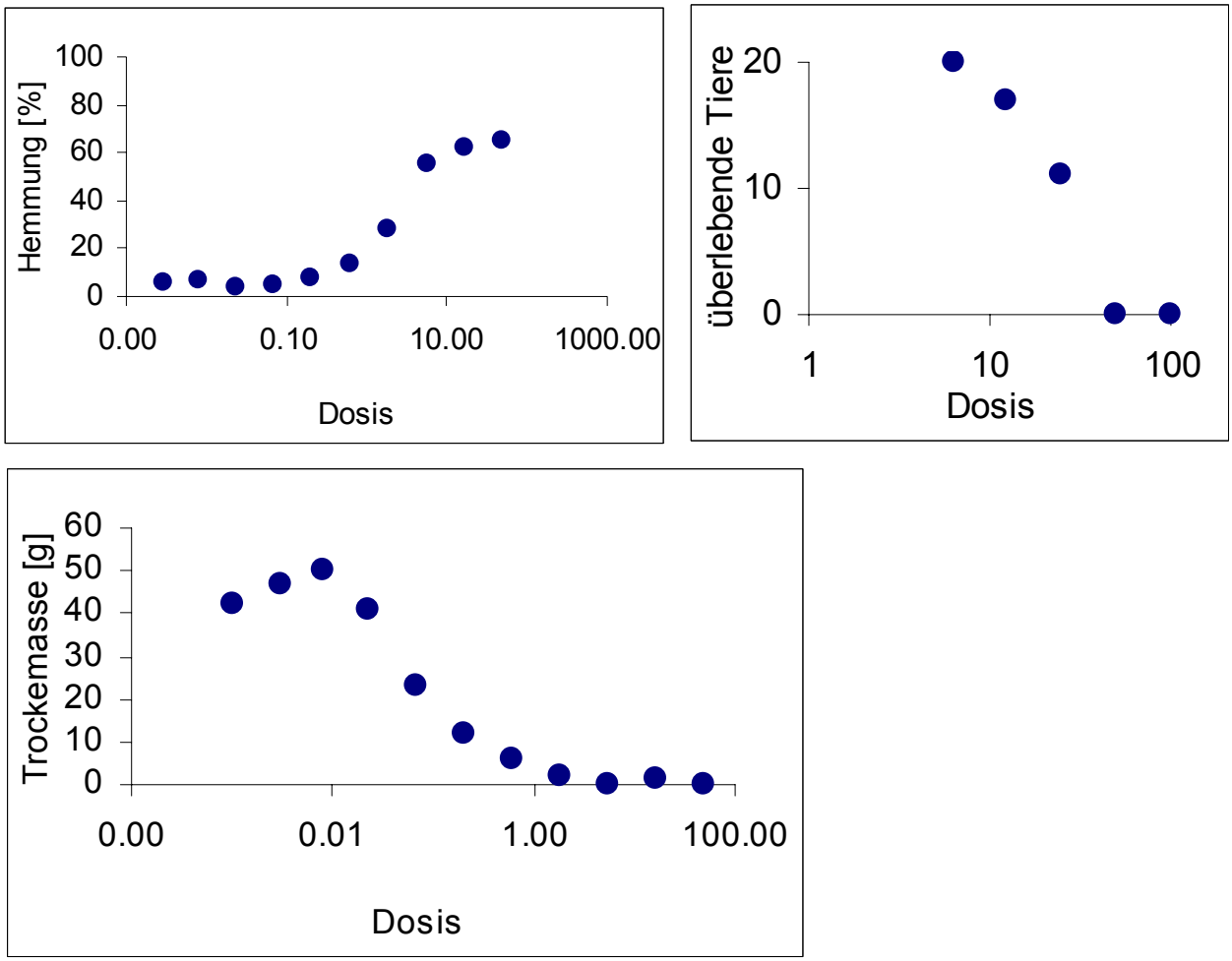

Abbildung 1-3: Beispiele für Dosis-Wirkungsbeziehungen im Pflanzenschutz

Tabelle 1: Beispieldaten Enzymhemmung

Dosis	Hemmung [%]
0.003	5.5
0.008	6.9
0.023	3.8
0.069	5.2
0.206	8.0
0.62	13.8
1.85	27.8
5.56	54.9
16.7	62.3
50.0	65.0

Struktur und Herkunft der Daten sind innerhalb der Forschung sehr heterogen. Die Ordinate ist im Allgemeinen intervallskaliert (Gewicht, %). Die Daten sind durch Schätzung, Zählung oder Messung entstanden. Je nach Forschungsgegenstand gibt es ein Minimum oder Maximum, oder die Skala ist unbegrenzt. Die Wirkung nimmt grundsätzlich mit zunehmender Dosis zu, die Funktion ist monoton steigend. Wenn auf der Ordinate allerdings ein Wachstumsparameter oder eine Überlebensrate abgetragen wird, dann nimmt dieser Wert mit zunehmender Dosis in der Regel ab, d.h. die Funktion ist monoton fallend. Daneben können Sonderfälle auftreten, bei denen die Funktion nicht monoton ist. In Abb. 3 ist ein bei Herbiziden beobachtetes Phänomen dargestellt, welches als *Hormesis* bezeichnet wird. Eine geringe Dosis wirkt zunächst aktivierend, erst bei weiterer Erhöhung der Dosis wirkt die Substanz toxisch.

Für die Modellierung der gezeigten Dosis-Wirkungskurven sind verschiedene Modelle entwickelt worden. Sehr verbreitetet sind das Probit- und das log-logistische bzw. Logit-Modell ([2], [1]).

$$p = \Phi\left(\frac{\ln(z)-\mu}{\sigma}\right),\quad p = \Psi\left(\frac{\ln(z)-\mu}{\sigma}\right)$$

Mit: p = Mortalitätsrate, Φ = Dichte der Normalverteilung, $\Psi(x) = 1 / (1+\exp(-x))$, z = Dosis, μ = Parameter für Wendepunkt, $\exp(\mu)$ ist die ED_{50}-Dosis, σ = Parameter für Steigung

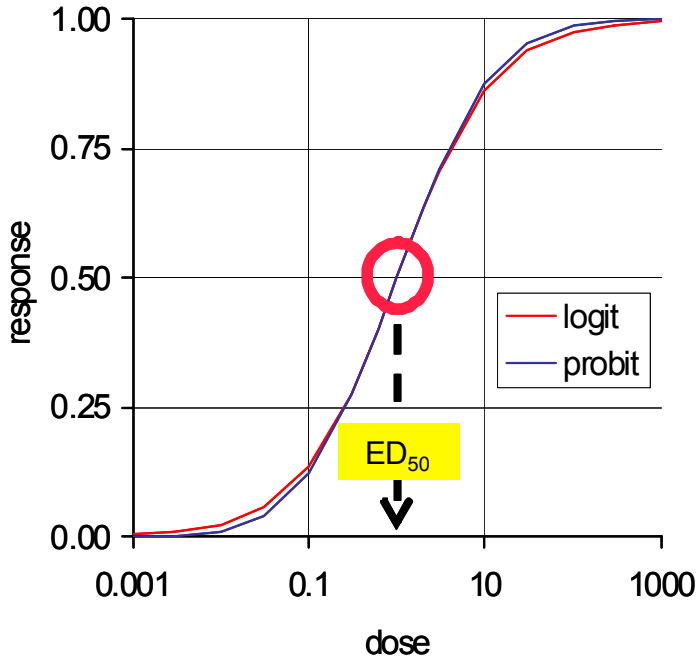

Abbildung 4: Verlauf der durch Probit- und Logit-Modell beschriebenen Funktion mit Wendepunkt bei einer Dosis von $ED_{50} = 1$

Der Wendepunkt der durch das Probit- bzw. Logit-Modell beschriebenen Funktion stellt genau die Dosis dar, die erforderlich ist, um eine 50%ige Wirkung zu erzielen. Diese Dosis wird je nach Forschungsgebiet als ED_{50}, IC_{50}, LD_{50} oder LC_{50} bezeichnet. Nachteil dieser beiden klassischen Modelle ist die Beschränkung auf Anteile als abhängige Variable mit Fixierung des Minimums und Maximums auf 0 bzw. 100%. Daher wurden von verschiedenen Autoren alternative Parametrisierungen des logistischen Modells vorgeschlagen, die hinsichtlich Minimum und Maximum flexibel sind und damit keine Vorgabe des Wertebereichs 0-100 erfordern ([7], [6]). Eine zusätzliche Erweiterung [5] erlaubt die Schätzung einer Dosis für einen frei wählbaren Wirkungsgrad. So beschreibt z.B. die ED_{90} die Dosis, die für 90% Wirkung notwendig ist. Zusätzlich wird auch eine hormetische Wirkung geringer Dosen erlaubt.

$$p = \delta + \frac{\alpha - \delta + \gamma z}{1 + \omega \exp\left[\beta \ln\left(z / ED_k\right)\right]}, \quad \omega = \frac{K}{100 - K} + \left(\frac{100}{100 - K}\right)\frac{\gamma ED_k}{\alpha - \delta}$$

im Falle von $\gamma = 0$ *(keine Hormesis):* $\omega = \frac{K}{100 - K}$

p = Mortalitätsrate bzw. Wirkung, α *= Maximale Wirkung,* δ *= Minimale Wirkung, z = Dosis,* ED_k *= Dosis für k% Wirkung,* $(\alpha - \delta)*k/100$, β *= Steigung am Wendepunkt, slope =* $-(\beta/4)$, *K = Anteilswert, z.B. K = 10 für* ED_{10}, γ *= Parameter für Hormesis*

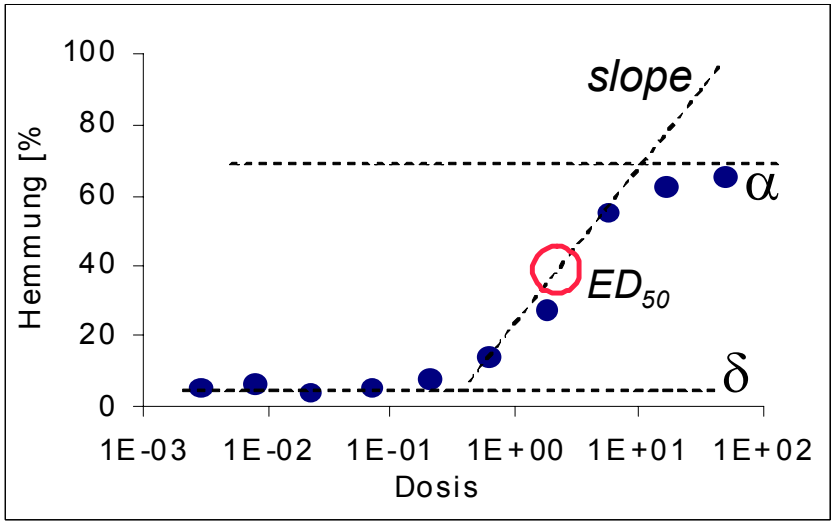

Abbildung 5: Beispiel für das 4- bzw. 5-parametrische log-logistische Modell Schabenberger [5]

2 Modellierung mit SAS PROC NLIN

Die unter 1 dargestellten nichtlinearen Modelle können mit der SAS Software an experimentelle Daten angepasst werden und die Parameter wie ED_{50}, Steigung, Minimum und Maximum geschätzt werden. Hierfür stehen u.a. die Prozeduren PROBIT, NLIN und NLMIXED des SAS/STAT Paketes zur Verfügung. Die Eingabe der Daten kann in einfachster Weise mit einem Data-Step erfolgen. Die Modellwahl erfolgt durch Syntax-Änderung oder durch Makrovariablen. So kann z.B. die Wahl des Dosis-Parameters *k* im Modell nach Schabenberger et al. [5] mit einer Makrovariablen erfolgen.

```
data test;
input description$ dose response;
cards;
Enzymtest       0.003      5.5
Enzymtest       0.008      6.9
Enzymtest       0.023      3.8
Enzymtest       0.069      5.2
Enzymtest       0.206      8.0
Enzymtest       0.620      13.8
Enzymtest       1.850      27.8
Enzymtest       5.560      54.9
Enzymtest       16.70      62.3
Enzymtest       50.00      65.0
;
run;
%let k = 50;
Proc nlin data=test method=marquardt noitprint;
```

```
      parameters
        alpha= 60 delta = 0  slope  = 1 EDk = 0.1 1 10;
        beta = -4*slope;
        omega = (&k/(100-&k));
        term = 1 + omega * exp(beta*log(dose/EDk));
      bounds alpha>delta;
      model response = delta + ((alpha - delta ) / term ) ;
      /*   model response = 0     + (( 1    - 0 )     / term ) ; */
run; quit;
```

Output

Parameter	Estimate	Approx Std Error	Approx. 95% Conf. Limits	
alpha	65.5407	1.5321	61.7917	69.2897
delta	5.7899	0.7875	3.8631	7.7167
slope	0.4021	0.0430	0.2969	0.5072
EDk	2.3935	0.1826	1.9466	2.8404

Diese Lösung sichert größtmögliche Flexibilität, erfordert vom Anwender jedoch sehr viel Know-how bezüglich der Syntax der verwendeten Prozeduren. Für die vielfältigen Einsatzgebiete der Modelle in der heterogenen Forschungslandschaft eines großen Pflanzenschutzunternehmens bedeutet dies eine separate und spezifische Programmierung des SAS-Codes, häufig auf Laborebene. Unter verstärkter Nutzung von Makrovariablen lässt sich dieses Problem ansatzweise vermindern.

```
      /* Modellwahl */
%let model_choice = probit;
      /* ED50, ED10, ED90 */
%let k = 50;
      /* Fixieren von Min und oder Max */
%let alpha = 100; %let delta = 0;
%let control_dose = 0; /* Behandlung der Kontrolle */
      /* Optionen für grafische Darstellung */
%let auto_scaling = no; %let y_axis_min = 0; %let y_axis_max = 20;
%let y_axis_step = 5; %let unit_x_axis = [µmol/l];
%let unit_y_axis = [%]; %let logbase = 10;
      /* Optionen für Ergebnisausgabe */
%let test_or_final = test;
%let output_style = basic;
```

Nach wie vor muss der Anwender aber direkt im SAS-Editor Eingaben vornehmen, wenn er die Makrovariablen aktiv verändern möchte. Die verstärkte Steuerung des Codes über Makrovariablen stellt aber einen ersten Schritt dar, hin zu einer zentralen Ablage des Codes auf einem zentralen Server.

3 Publikation der SAS-Makros als „Stored Process“ in einer SAS-BI-Umgebung

Die SAS-BI-Technik bietet das Einbinden der SAS-Makros in einen Stored Process an. Dieses ist ein SAS-Programm, das auf einem Server gespeichert ist und von anderen Anwendungen aufgerufen werden kann. Aufrufmöglichkeiten sind z.B.: SAS-Enterprise Guide, SAS-MS-Office-Integration (SAS-AMO), Web. Es ist eine Berechtigungsverwaltung und Parametrisierung möglich. Grundsätzlich kann jedes SAS-Programm zu einem Stored Process werden. Es wird bei Erstellung lediglich um Metadaten erweitert, die den Stored Process beschreiben. Die Ausführung erfolgt i. d. R. über einen Stored Process Server. Für viele Anwender ist die Anbindung an Microsoft Excel über das SAS-AMO besonders interessant [3].

Statt für jedes Modell und jede Umgebung einen separaten Code zu programmieren, wird der Code bei einem Stored Process zentral auf einem SAS Enterprise BI Server bereitgestellt und mittels Makrovariablen steuerbar gemacht. Das angenehme für SAS unerfahrene Anwender ist, dass die SAS-Stored-Process-Technologie eine grafische Eingabemöglichkeit für die als Parameter bezeichneten Makrovariablen bietet - die direkte Berührung mit SAS-Code entfällt. Für den SAS-Programmierer bietet sich die Möglichkeit, aus bereits vorhandenem SAS-Makrocode mit wenigen Mausklicks ein professionell anmutendes Anwendungsprogramm mit grafischer Nutzeroberfläche zu gestalten, das sich zudem perfekt in die MS Office Welt der Nutzer integriert. Als Tool zur Erstellung der Stored Processes und Publikation auf dem SAS Server haben wir mit dem SAS Enterprise Guide gute Erfahrungen gemacht.

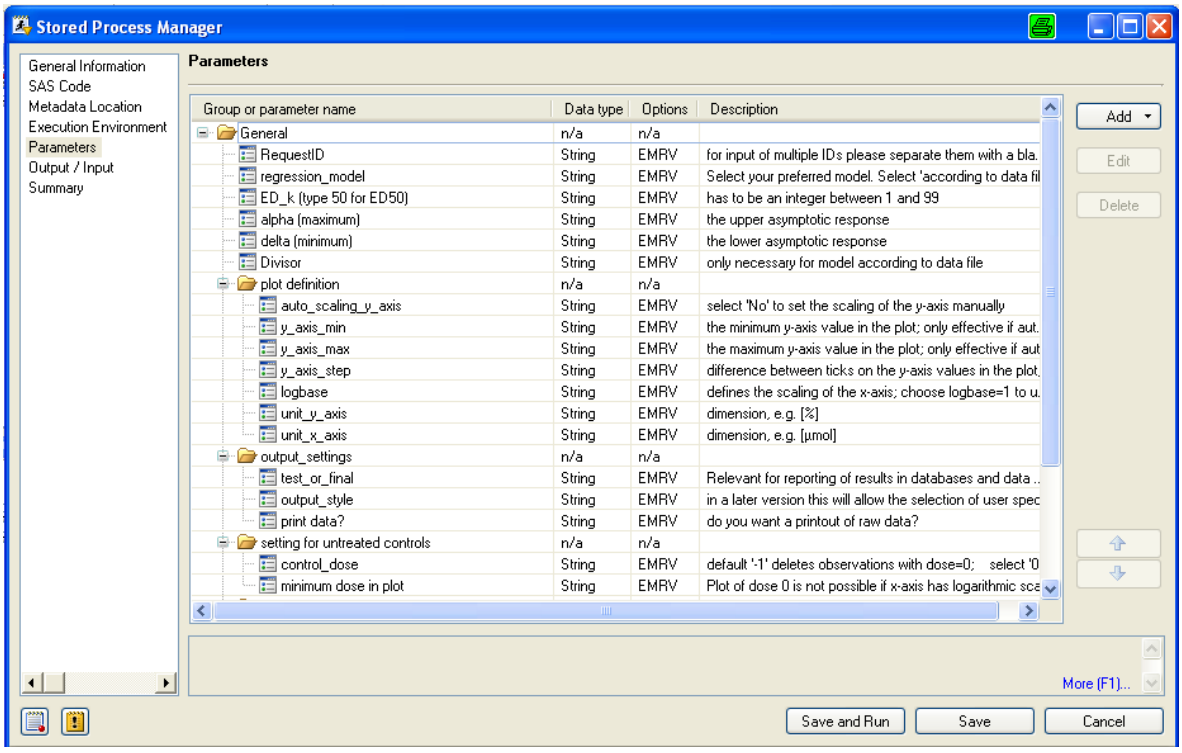

Abbildung 6: Definieren der Makro-Variablen als Parameter im SAS Enterprise Guide

4 Einbindung der Stored Process Technologie im Sinne einer Service Oriented Architecture (SOA)

Die auf dem zentralen SAS-Server bereit gestellten Stored Processes können von verschiedenen Nutzern und Systemen über eine Web-Service-Schnittstelle, das SAS-AMO oder auch 3rd-Party-Applikationen aufgerufen werden. Zur Bereitstellung der Daten wird eine Datenbank genutzt, die die Daten temporär vorhält. Der Stored Process holt sich die Daten dann bei Aufruf des Services ab. Um zwischen verschiedenen Datensätzen wählen zu können, wird eine laufende Nummer (RequestID), z.B. beim Datenupload vergeben, auf die im SAS-Makro über einen Parameter referenziert wird. Dieser Work- und Datenflow im Rahmen eines „Statistik Service“ ist in Abb. 7 schematisch dargestellt.
Diese Architektur mit einem standardisierten Web-Service ermöglicht eine Vielzahl von Quellsystemen als Datenlieferant und verschiedenen Anwendern als Service-Consumer und passt damit perfekt in die von uns angestrebte „Service Oriented Architecture“.

5 Nutzung des SAS-AMO

Für Nutzer ohne Programmiererfahrung und mit geringen Statistik-Kenntnissen ergeben sich durch die Bereitstellung der SAS-MS-Office-Integration (SAS-AMO) interessante Möglichkeiten für eine statistische *ad hoc* Analyse eigener Daten mittels grafischer Benutzeroberfläche. Im Rahmen des oben dargestellten Statistik-Service ist nun besonders interessant, dass ein Anwender mit SAS-AMO Installation auch direkt aus z.B. Excel über die Schaltfläche „Berichte“ auf die zentral abgelegten Stored Processes zugreifen kann. Die Ergebnisse werden dann wiederum direkt in Excel ausgegeben.

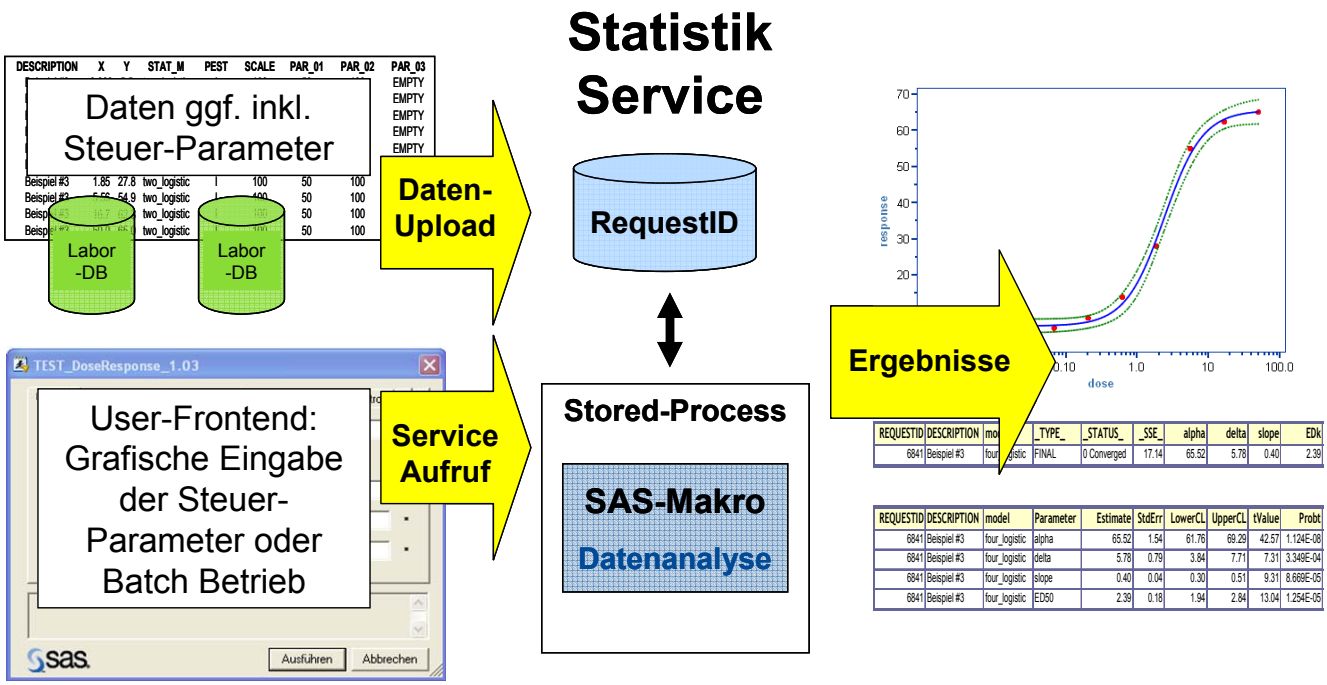

Abbildung 7: Workflow und Datenflow des Statistik Service

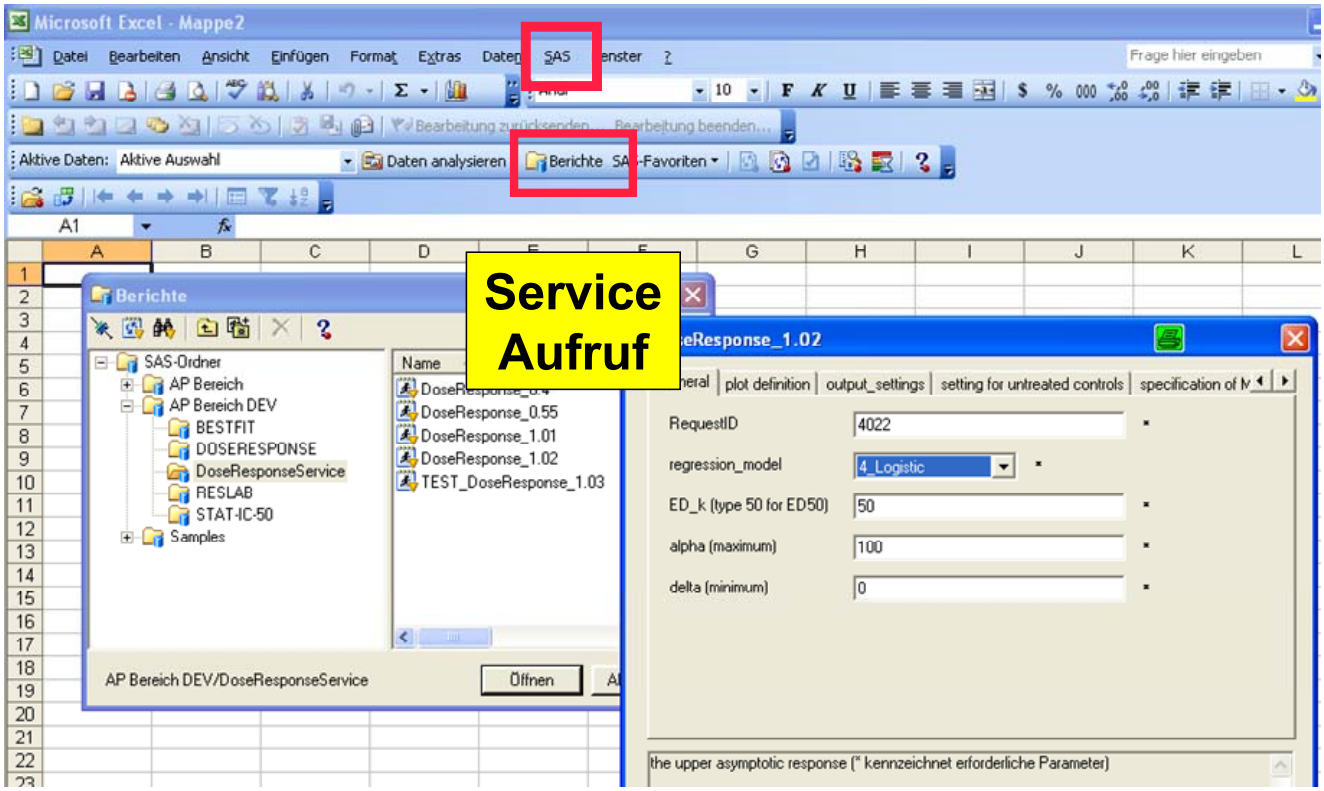

Abbildung 8: Aufruf des Stored Process „DoseResponse“ direkt aus Excel heraus

Mit der bei einer Standardinstallation des SAS-AMO zur Verfügung gestellten Konfiguration ist es allerdings nicht möglich, Daten die sich innerhalb eines geöffneten Excel-Arbeitsblattes befinden, auf den SAS-Server hochzuladen und dann mittels eines selbst erstellten Stored Processes zu verarbeiten. Der Grund hierfür ist, dass der SAS-AMO-Workspace-Server und der SAS-Stored-Process-Server verschiedene SAS-Sitzungen darstellen und in unterschiedlichen temporären Work-Directories arbeiten. So hat eine zur Auswertung mit dem SAS-AMO auf den Server hochgeladene Excel-Datei üblicherweise den systeminternen SAS-Dateinamen „WORK._EXCEL_“. Eine Anbindung des Stored Process im Data-Step an eben diese WORK._EXCEL_ führt allerdings ins Leere. Der Stored Process findet die Datei nicht. Um dieses Problem zu umgehen, müssen eigene Lösungen programmiert werden. So nennt z.B. der Redscope Forumseintrag zum Thema unter http://www.redscope.org/node/592 folgende Lösungswege:

„Einfach und unbequem: Die Excel-Datei auf dem Server speichern und anschließend im Stored Process einlesen, diesem muss der Speicherort der Datei bekannt gemacht werden, zum Beispiel über einen Parameter.“

„Aufwändig und komfortabel: Ein Add-In für SAS Enterprise Guide schreiben (das dann auch im Office Addin verfügbar wird), mit dem man Stored Processes direkt in der SAS-Workspace-Sitzung ausführen kann. Dieses kann dann wie eine in SEG eingebaute Anwendungsroutine aufgerufen werden und work._excel_ wäre dann verfügbar.“

6 Custom Add-In Erweiterung zum SAS-AMO

Wir haben uns für die aufwändige aber im Sinne der User komfortablere Lösung entschieden, die Entwicklung eines Custom Add-In als Erweiterung zum SAS-AMO. Hierfür wurde ein Visual Studio 2003 .NET-Template verwendet, das SAS im Internet zur Verfügung stellt [4].

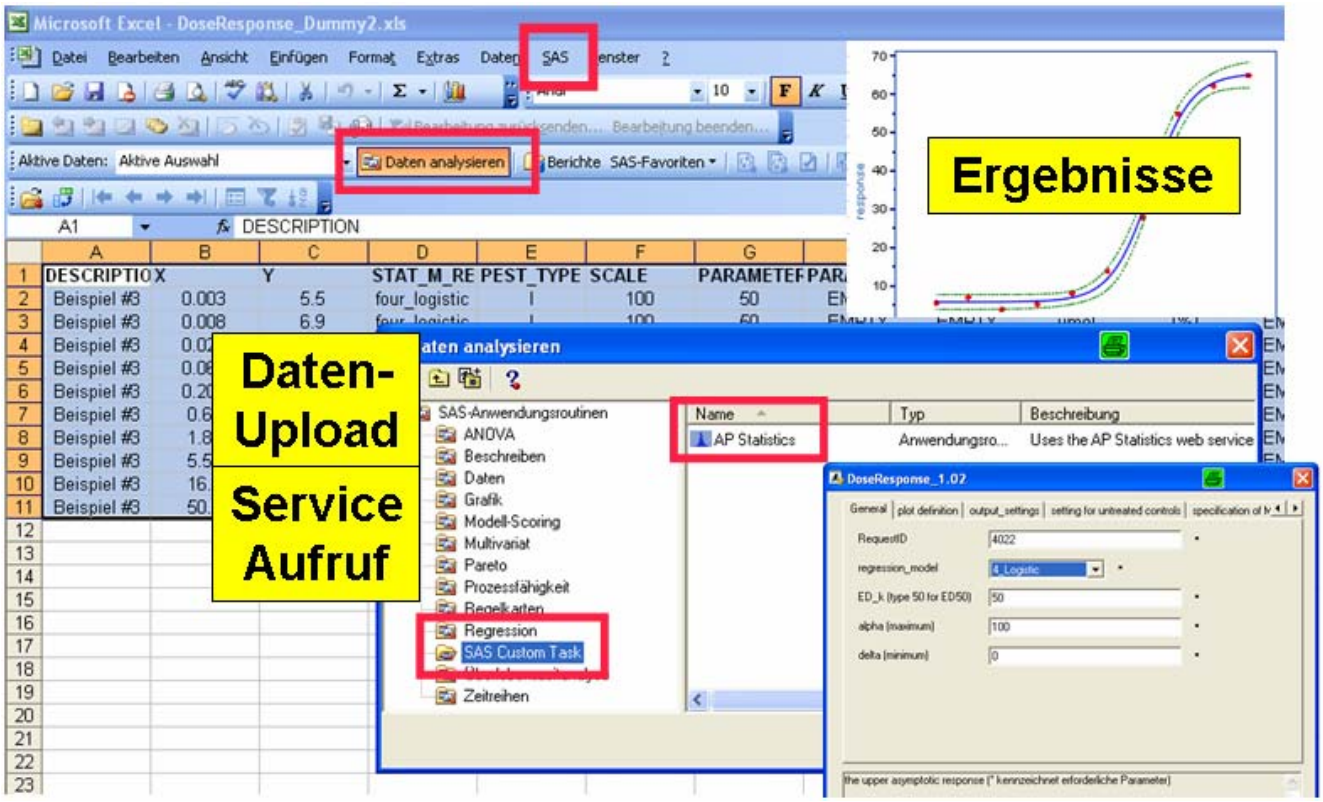

Abbildung 9: Aufruf des Stored Process „DoseResponse“ über das Custom Add-In

Durch das Custom Add-In ist es nun möglich, Daten die sich innerhalb eines geöffneten Excel-Arbeitsblattes befinden, auf den SAS-Server hochzuladen und dann mittels eines selbst erstellten Stored Processes zu verarbeiten. Der Aufruf dieser Funktion geschieht über den SAS Menüpunkt "Daten analysieren" und dann durch die Auswahl unseres Custom-Add-In (AP Statistics). Die Ergebnisse werden direkt in Excel angezeigt.

7 Diskussion

Durch die gewählte Systemarchitektur kann den häufig heterogenen Anwenderwünschen entgegen gekommen werden und eine gewisse Flexibilisierung zugelassen werden. Nutzer können entsprechend ihrer Rolle aus den verschiedenen Services und Schnittstellen für Input und Output wählen. Modellwahl und Steuerung sind interaktiv möglich. Anwender ohne jegliche SAS-Kenntnis können damit die Power des SAS-Systems nutzen. Hierdurch steigen allgemein die Möglichkeiten und auch die Motivation zu einer tiefer gehenden statistischen Datenanalyse. Gleichzeitig können Datenma-

nagement und statistische Datenanalyse zentral administriert werden. Ein Wildwuchs von Methoden wird verhindert. Jedem Nutzer kann entsprechend seiner Rolle und seinem Know-how die Auswahl an Services und die Entscheidungsbefugnis zugewiesen werden, die gewünscht ist. Ebenso lässt sich die dargestellte Architektur für eine automatische Datenanalyse in der Routine einsetzen. So können z.B. Optionen und Modellparameter bereits zusammen mit den Daten übergeben werden.

Um die Leistungsfähigkeit des gesamten Systems annähernd ausschöpfen zu können, ist eine gute und enge Zusammenarbeit von Statistikern, Datenbank-Programmierern, SAS-Programmierern und Anwendern notwendig.

8 Notwendige SAS-Komponenten

Für die Erstellung und Bereitstellung der Stored Processes:

- SAS Enterprise Guide oder SAS Foundation und SAS Management Console
- SAS-Enterprise-BI-Server

Für die Nutzung (Clients)

- Web-Browser oder SAS-MS-Office-Integration (AMO) oder AMO mit SAS-Custom-Task

Literatur

[1] Berkson (1944): Application of the logistic function to bioassay. J. Amer. Stat. Ass. 41, 357-365.

[2] Bliss (1934): The method of probits. Science 79, 38-39.

[3] SAS (2007): SAS Add-In 2.1 for Microsoft Office: Getting Started with Data Analysis. Cary, NC: SAS http://support.sas.com/documentation/onlinedoc/91pdf/sasdoc_913/ms_addin_9701.pdf

[4] SAS (2009): Creating Custom Add-In Tasks for SAS Enterprise Guide. Visual Studio .NET 2003 template files for creating add-in tasks. http://support.sas.com/documentation/onlinedoc/guide/customtasks/index.htm

[5] Schabenberger, Tharp, Kells & Penner (1999): Statistical Tests for Hormesis and Effective Dosages in Herbicide Dose Response. Agronomy Journal 91, 713–721.

[6] Seefeldt, Jensen & Fuerst (1995): Log-logistic analysis of herbicide dose-response relationships. Weed Technol. 9: 218–227.

[7] Streibig (1988): Herbicide bioassay. Weed Res. 28: 479–484.