

Java Erweiterungen in der SAS Software

Frank Rosner
SAS Institute

Heute verfügbare Web- Anwendungen

- Existierende oder neue Anwendungen? *Web Publishing*
- Statische oder dynamische Daten ?
- Welche Art von Verarbeitung ? *Dispatcher*
- Training/Weiterbildung der Mitarbeiter ? *htmSQL*
- Wie interaktiv ist die Anwendung ?
- Wie oft und durch wen wird sie benutzt ? *Dynamic HTML*
- Welche Browser ?
- Wie groß sind die Datendateien ?



Überblick

- Java: Pro und Cons
- Java Clients für SAS Server
 - “thin client graphics”
 - Metaspaces Explorer
- webAF
- webEIS

Teil 1

Java:

Pro und Cons

Java

- Eine **objektorientierte Programmiersprache**
- ausgeführt durch die **Java Virtual Machine (JVM)**
- **Portabel**, läuft auf vielen Betriebssystemen
- Applets werden durch den Webbrowser ausgeführt und dabei durch eine HTML Seite “angezogen”

Java Applets

- Werden dynamisch heruntergeladen
- Ausführung passiert auf der Client-Seite (Browser)
- haben eine sichere Umgebung
 - *“Untrusted applets”* werden durch den Security Manager überwacht
 - *“Trusted Applets”* haben konfigurierbare Einschränkungen

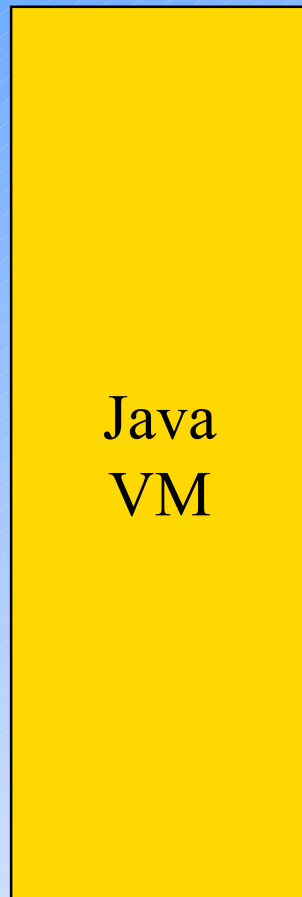
Sicherheit bei “Untrusted Applets”

- Netzwerk-Zugriff *nur auf den Webserver* (wo das Applet herkam)
- auf lokale Dateien des Clients kann *nicht* zugegriffen werden
- auf andere Anwendungen auf dem Client-Rechner kann nicht zugegriffen werden

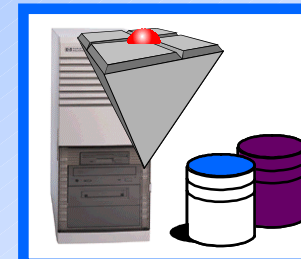
Verteilte Anwendungen mit Applets



Client



Entfernter
Methoden-
Aufruf



Anwdg.server

Herunterladen der
Java-Klassen
der Anwendung



Webserver

Java DataBase Connectivity (JDBC)

- Ist in Java-Version 1.1 eine Standard
- braucht herstellerspezifische Treiber oder eine JDBC/ODBC-Bridge
- gibt Datenzugriff mittels SQL
 - *Anfragen*
 - *neue Tabellen erstellen*
 - *Updates bestehender Tabellen*

Cons: Java-Performance

- Performance *ist* ein Problem
- Was tun ?
 - JIT-Compiler schaffen Linderung
 - JAR-Files verkürzen die Zeit der Herunterladens der Klassen
 - JDK 1.2 - JFC
 - *zusätzliche Oberflächen-Komponenten sind schon beim Endbenutzer*
 - *=> Applets werden kleiner*

Cons: Java Stabilität

- Die Haupt-Browser verbesserten ihre Java -Unterstützung nur langsam
- Es gab und gibt **Unterschiede** in der Implementierung der VMs
- Das ***Java Plug-in*** gibt eine stabile Java-Umgebung für Netscape und Internet Explorer

Tendenz: Java Plug-in

- Gibt konsistente JDK 1.1 Unterstützung
- **Sicherheitsbegrenzungen** auf die Serververbindungen sind **einstellbar** - das ist insbesondere für Intranet-Applets nützlich
- Erhältlich von Sun/JavaSoft

Typen der Java-Applets

- **“leichtgewichtige” Applets**
 - bekommen alle Parameter von der HTML-Seite, haben keine Extra-Verbindung mit einem Server
- **“Connection-Applets”**
 - unterhalten eine Verbindung mit einem (normalerweise entfernten) Server

Teil 2

Java Clients

für SAS Server

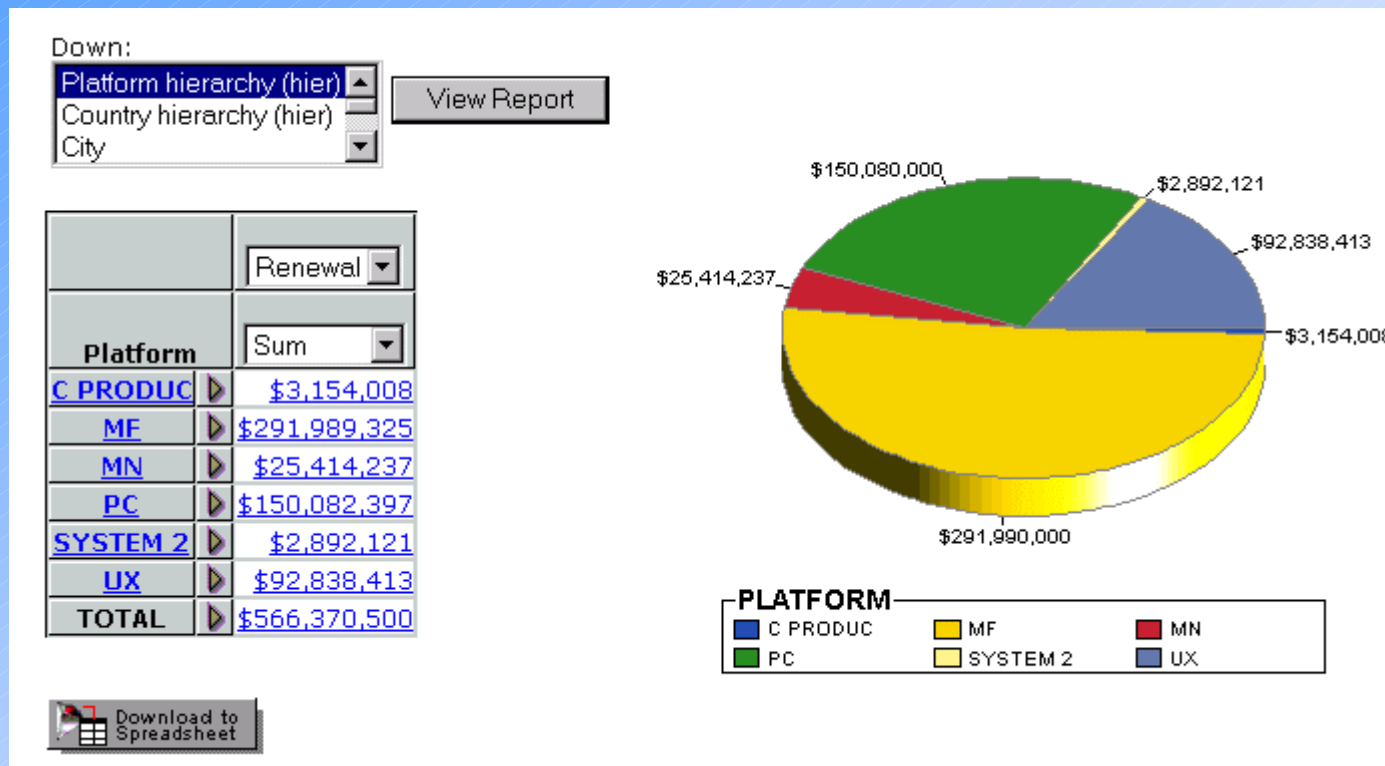
Java-Komponenten von SAS Institute

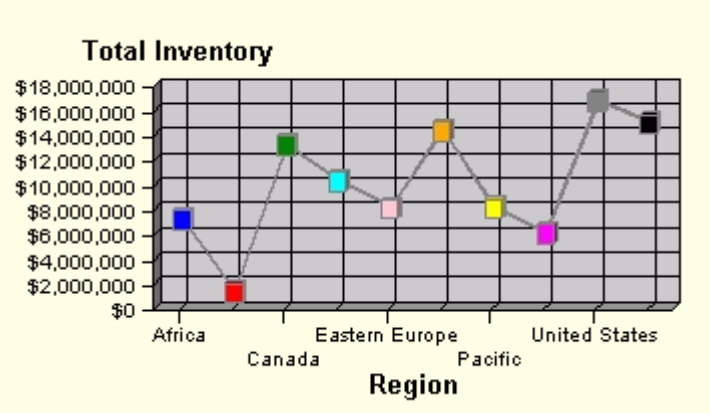
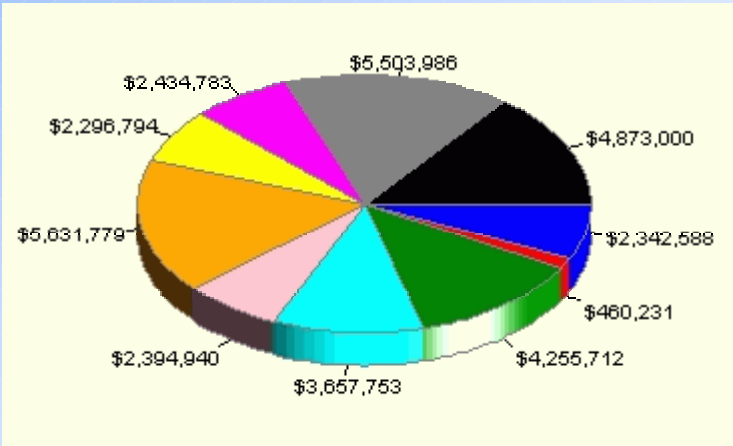
- Leichtgewichtige Applets
 - ***“thin client graphics”-Applet***
- Connection-Applets:
 - *Applets, die den SAS/SHARE Treiber für JDBC benutzen*
 - *Applets, die den SAS/Connect Treiber für Java benutzen*
 - ***MetaSpace Explorer***

„thin client graphics“

- **Ziel:**
 - leichtgewichtiges Applet
 - interaktive, gutaussehende Graphiken für das Web
- **Wie:** *Paket aus 2 Teilen*
 - GraphApplet HTML Generator (ds2graf)
 - die Applet-Klassen (GraphApplet.jar)

Einsatz des „thin client graphics“ applet im MDDDB report viewer





SAS als Server: Typen

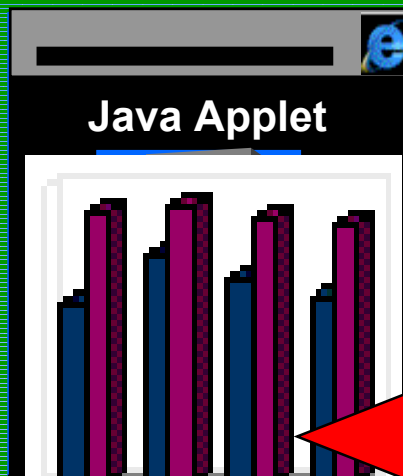
- Datenserver: Zugriff über JDBC
 - *SAS/SHARE Server*
 - *Scalable Performance Data Server (SPDS)*
- Applikations- (“Compute) Server
 - ***SAS/CONNECT Server***
 - *Application Dispatcher (CGI-Zugriff von Java aus möglich)*

SAS/CONNECT Client und Server: Features

- SAS-Statements und Prozeduren submitieren
- SQL-Interface
- (remote) SCL-Objekte (in neuer Version)
- pro Benutzer eine eigene SAS-Sesssion
- verschiedenster Outputmöglichkeiten

SAS/Connect Treiber für Java

Browser

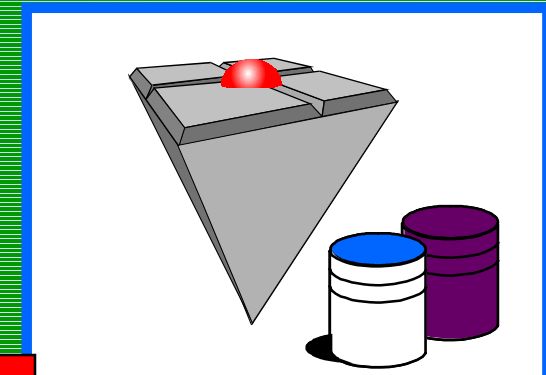


Connecttreiber

Webserver



Klassenbibliotheken



SAS/CONNECT

"Start auf Anforderung"

Herunterfahren



Wann sollte man den SAS/Connect Treiber benutzen ?

- Wenn Rechen- und Datendienste erforderlich sind
- wenn pro Benutzer mehrere Anfragen kommen, die untereinander verbunden sind.

SAS/Connect Applet: MetaSpace Explorer

- Benutzt Metadaten, die vom SAS/Warehouse Administrator 1.3 exportiert wurden
- Aufgabe: zu den Daten und Berichten navigieren können
- für normale Benutzer

MetaSpace Explorer

- JDBC für das Lesen
 - der Metadaten (exportiert von SAS/WA 1.3)
 - der Daten, die der Benutzer ausgewählt hat
- SAS/Connect Treiber: um die Katalogeinträge zu extrahieren
- ROCF: für das MDDDB Datenmodell

Wo wird die Graphik erzeugt ?

- Server-seitig: am besten bei **großen Datenmengen** oder komplexen Rechenoperationen
- Client-seitige Erzeugung: bei (heruntergeladenen) kleineren Datenmengen, aus denen dann die Graphik erzeugt wird

WIE wird graphischer Output erzeugt ?

- **Server-seitige Graphiken:**
 - GRSEG, IMAGE , das zu GIF umgewandelt wird
 - VRML-Output von proc ggraf
- **Client-seitige Graphiken:**
 - VRML browser
 - “thin client graphics”

Teil 3

WebAAF

AppDevStudio

- webAF und webEIS sind Teil eines neuen SAS-Produkts für Entwickler: AppDevStudio
- AppDevStudio beinhaltet diese beiden Java-Komponenten und zusätzlich eine Win32-SAS-Installation mit den von Entwicklern benötigten Produkten (AF, EIS, ...)

Was ist ein intelligenter Client ?

- Einfacher Zugriff auf den SAS-Server
- Transparenter Zugriff auf SAS/AF[®] Objekte
- Transparenter Zugriff auf Nicht-SAS/AF SAS[®] - Objekte
- Zugriff auf Daten (Tabellen, MDDDBs)
- Zugriff auf Rechenleistung durch Procedure-Submittierung

Ziele von webAF

- Entwicklung von intelligenten Clients einfach machen
- Komplexität verteilter Anwendungen verbergen
- Bestehende Kunden-Investitionen in das SAS System sichern

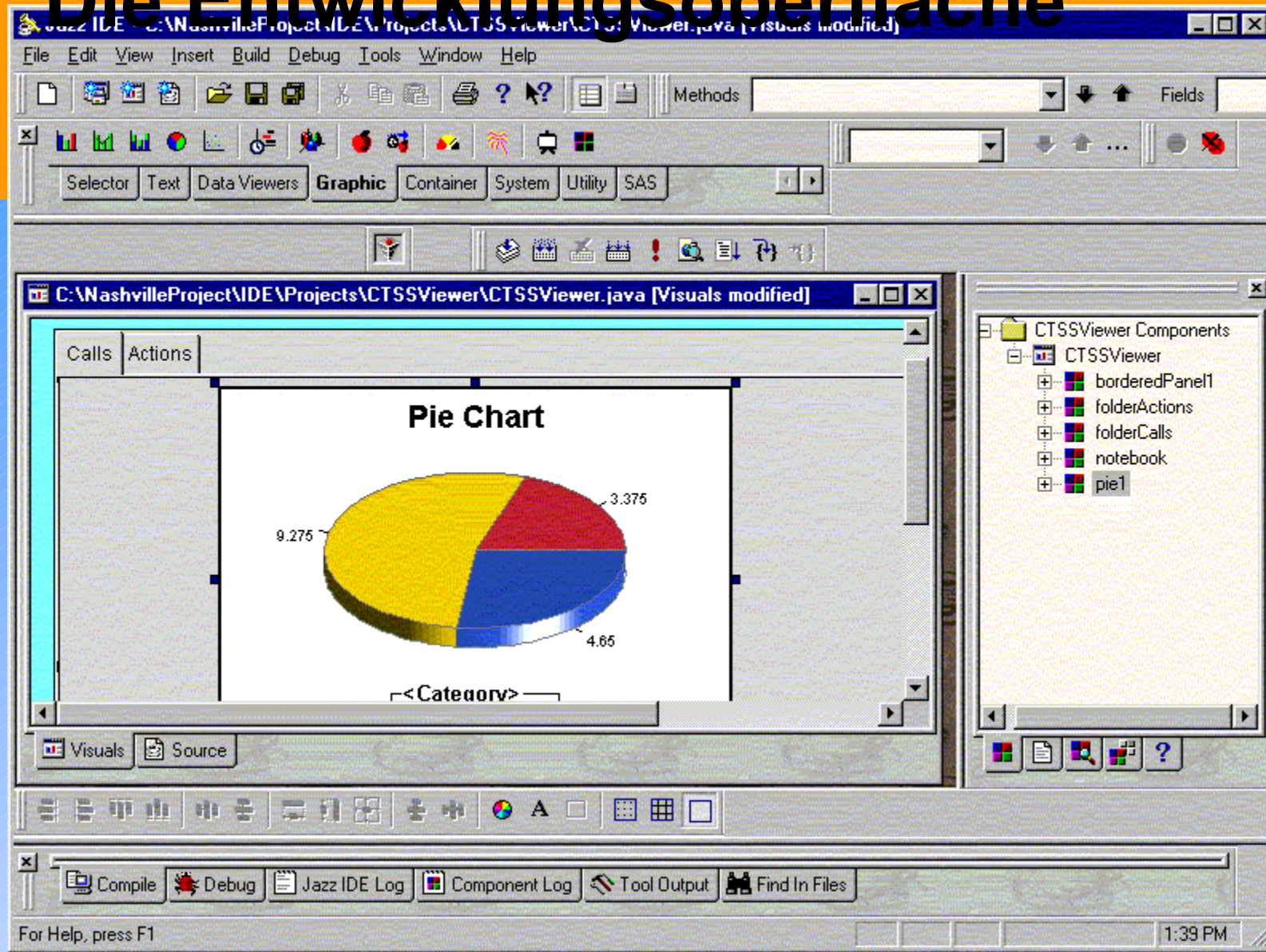
SAS Investitionen schützen

- Nicht jeder braucht in Java geschult werden.
- Serverseitige Geschäftslogik in SAS durch geschulte SAS-Entwickler
- Java-Programmierer greifen auf diese Modelle zu, sie schreiben das GUI dazu.

webAF Grundaussagen

- Visuelle Java-Entwicklungsumgebung für die leichte Nutzung von SAS Server-Sessions
 - V6 Server
 - V7 Server
- ist Win32-Anwendung
- Generiert “100% reinen” Javacode

Die Entwicklungsoberfläche



webAF Grundaussagen

- Zugriff auf SAS/AF[®] Objekte
- Datenzugriff (Tabellen, MDDDB)
- Prozeduren-Submits unterstützt

webAF Grundaussagen

- unterstützt JDK 1.1.7
- ist “bean compliant”
- fußt auf Model-View -Konzepten für erweiterte Wiederverwendbarkeit der Komponenten
- Benutzt “visual linking” der Komponenten - damit wird der Programmieraufwand reduziert

webAF Grundaussagen

- Über 100 Komponenten werden ausgeliefert
- Für die wichtigsten Dinge in der SAS-Server-Session sind serverseitige Modelle verfügbar
- kann Beans anderer Hersteller integrieren

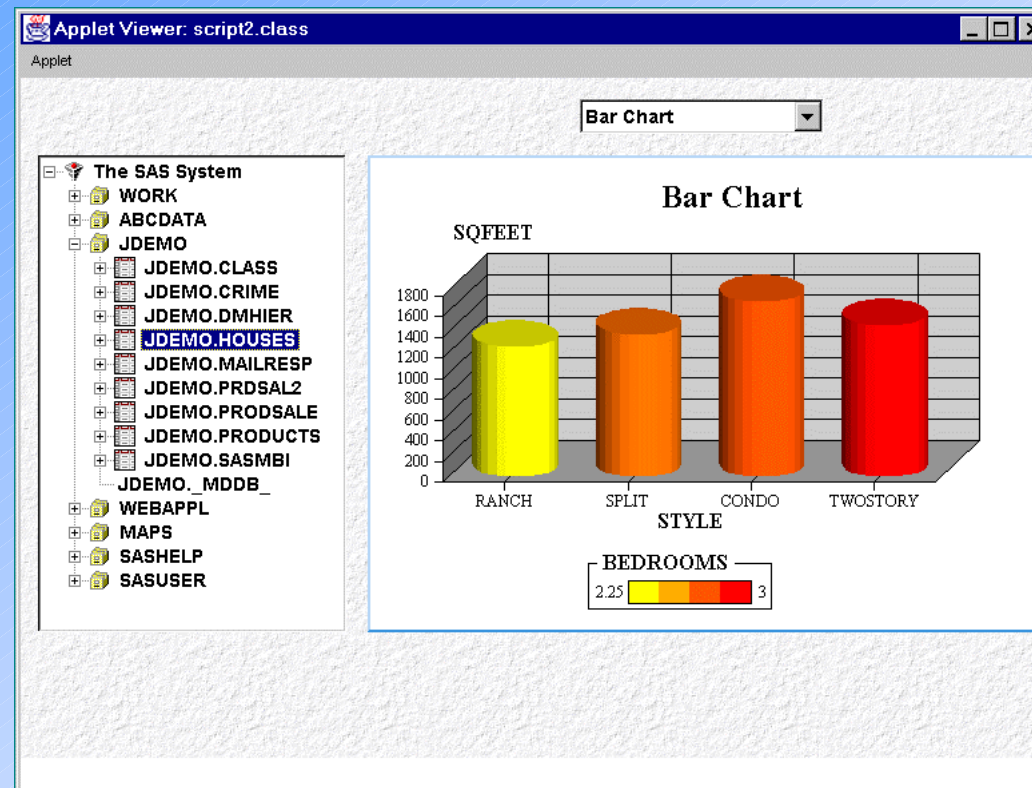
webAF Grundaussagen

- eingebauter Debugger
- sprachsensitiver Editor
- Wizards für sich wiederholende Aufgaben
- andere Goodies

Anforderungen an die (serverseitige) SAS-Software

- SAS 6.12 (TS060)
 - SAS/IntrNet[®]
 - SAS/Connect[®]
 - SAS/AF[®] (für die Entwickler auf der Serverseite)
 - [SAS/SHARE[®]]

Beispielschirm (1)



Beispielschirm (2)

Applet Viewer: script3b.class

Applet

PRODTYPE		FURNITURE		OFFICE	
ANLSYS		Actual Sales	Predicted Sales	Actual Sales	Predicted Sales
COUNTRY	REGION	SUM	SUM	SUM	SUM
CANADA		\$97,864.00	\$89,941.00	\$149,126.00	\$143,078.00
GERMANY	EAST	\$49,751.00	\$47,998.00	\$74,796.00	\$69,581.00
	WEST	\$51,443.00	\$45,315.00	\$70,008.00	\$68,660.00
U.S.A.		\$91,567.00	\$95,064.00	\$145,782.00	\$146,658.00

Geographic Product Line Time

Geographic Product Line Time

Actual Sales
Sales Lag
Predicted Sales
Sales Ratio

NMISS
N
SUM
MIN
MAX
USS
RANGE
AVC

Display Table

Traffic Lighting

Teil 4

webEIS

The image features the text 'webEIS' in a large, bold, sans-serif font. Each letter is filled with a different color from a rainbow spectrum: 'w' is pink, 'e' is orange, 'b' is yellow, 'E' is green, 'I' is blue, and 'S' is purple. The letters have a slight 3D effect with soft shadows cast to the right and slightly forward. The background is a light blue gradient with a fine, repeating pattern of small, light-colored lines.

webEIS

“was ist es?”

- 100% “Pure Java” Anwendung
- Applet-Generator
- “visual building tool”
- “thin client” OLAP Lösung für das Web
- entwickelt sich (noch)
- alles zusammen !!!

webEIS

“so...”

- Ermöglicht den Anwendern in der gesamten Organisation, auf **multidimensionale Geschäftsdaten** zuzugreifen und sie zu analysieren
- gibt den Fachabteilungen die Möglichkeit, Daten auf zentralisierten Servern zu pflegen
- gestattet es den Usern, EIS-artige Webdokumente zu erzeugen, die **interaktive, drillfähige Graphiken und Tabellen** enthalten - und zwar **ohne jegliches Coding** !

webEIS

“mehr...”

- Dokumente können **mehrere Sektionen** haben, jede kann potentiell auf verschiedene Datenbanken zugreifen
- webEIS kann MDDDB-Daten anzeigen, auch von **HOLAP-Umgebungen**, die auf Servern liegen
- webEIS gibt den Usern EIS - Operationen, darunter **Drilldown, Expand, Collapse, Traffic lighting** und mehr

webEIS

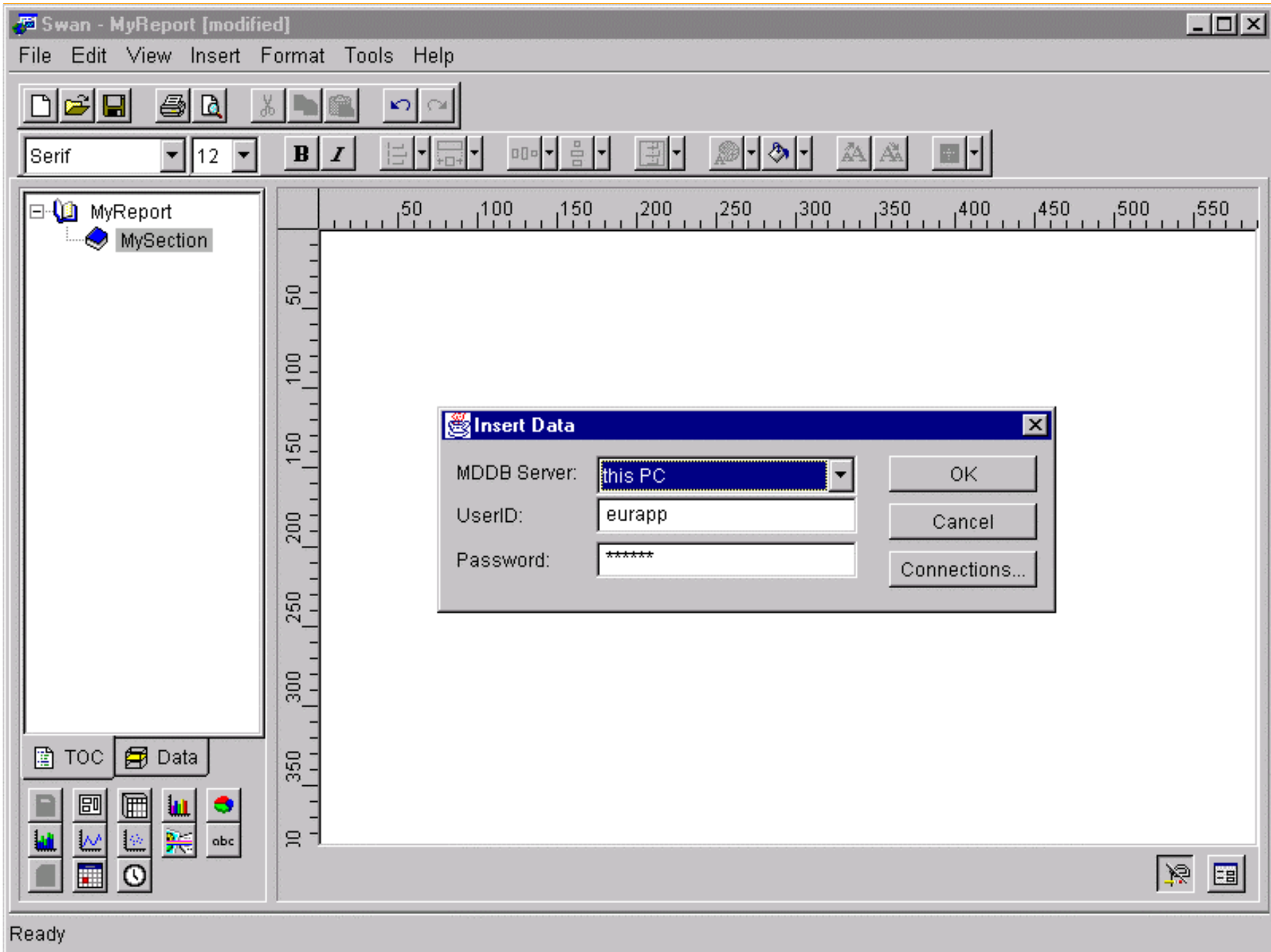
“Was ist nötig?”

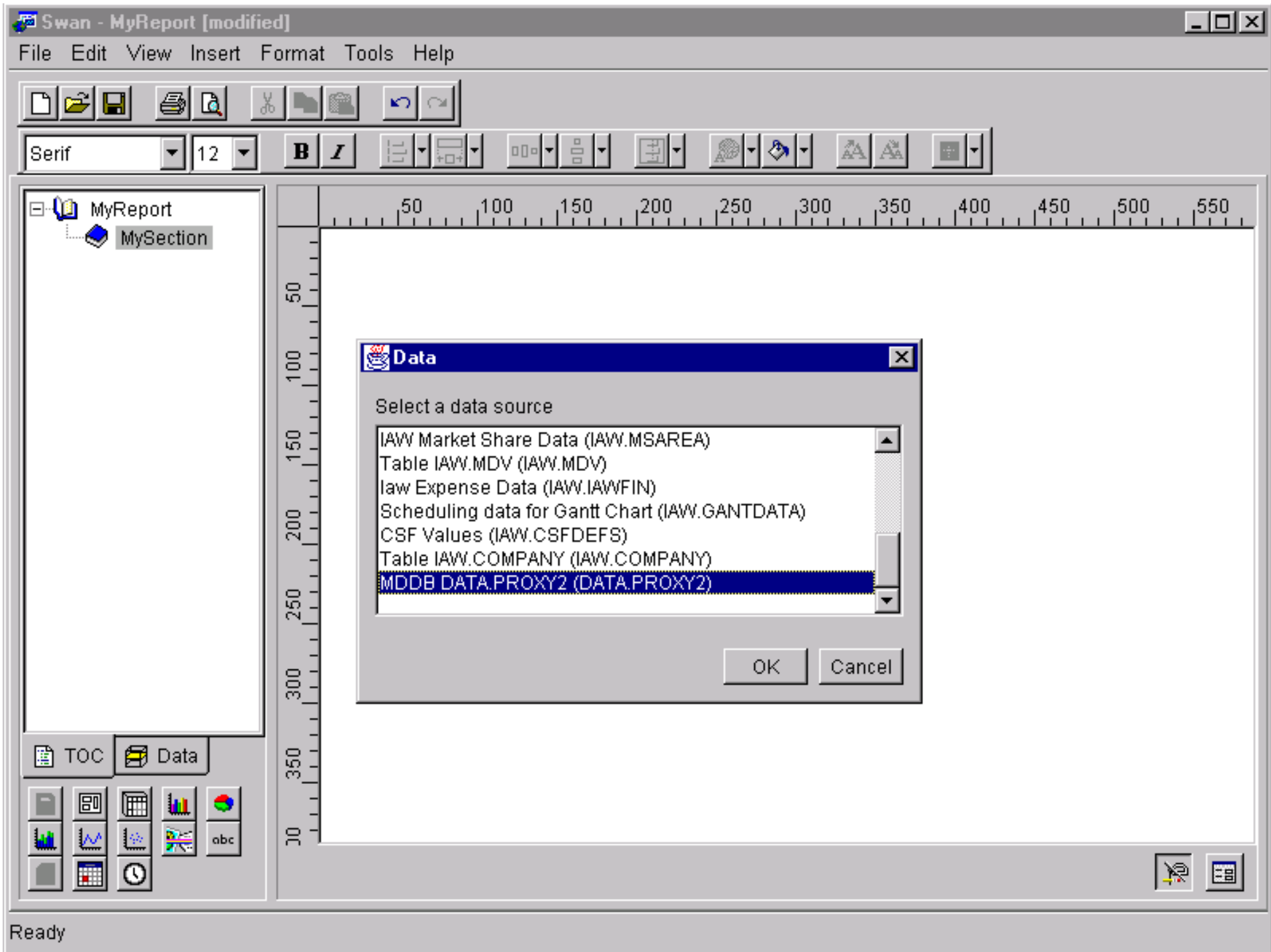
- Server:
 - SAS 6.12 maintenance (TS060)
 - SAS/EIS
 - MDDDB Server
 - SAS/Connect
 - SAS/IntrNet

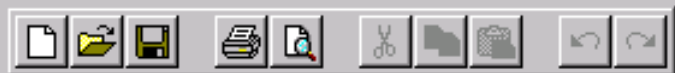
webEIS

“Was ist nötig?”

- Client:
 - IE4, Netscape 4 (beide mit JavaSoft Java plug-in, bzw. “Activator”), oder HotJava Browser für den Applet-Fall
 - JavaSoft JDK 1.1 für Applikationen
 - Windows NT oder Windows 95 (auf anderen Hosts wird noch getestet)





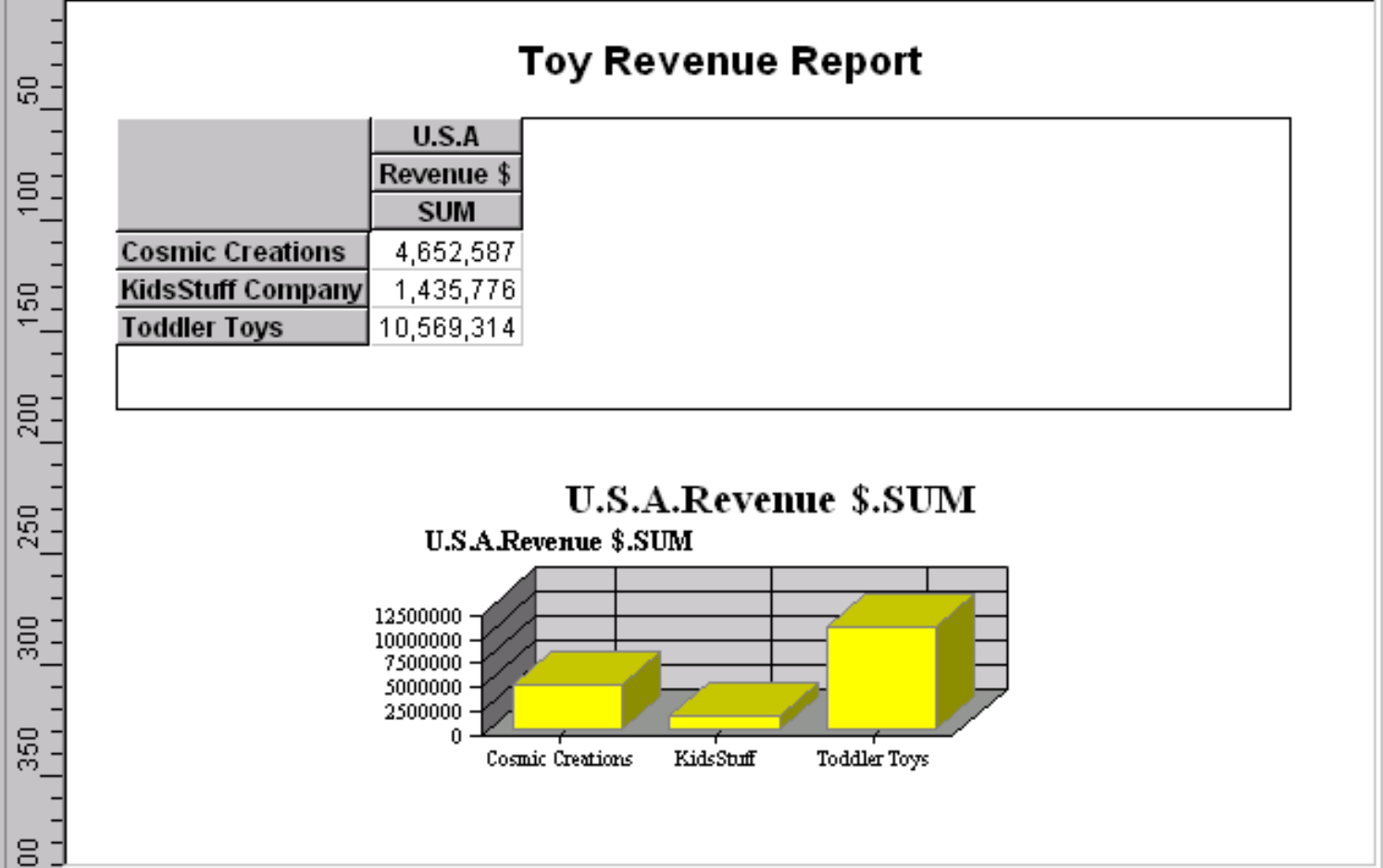
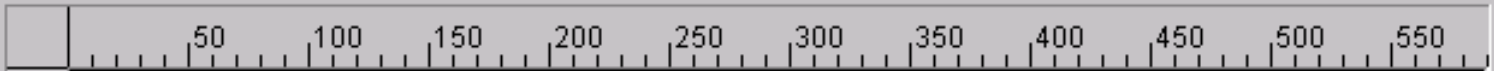


MySection

- ▶ Subsetters
- ▼ Rows/Down
 - Product...
- ▼ Columns/Across
 - Geography...
 - Revenue \$
- ▼ Statistics
 - SUM

Apply

TOC Data



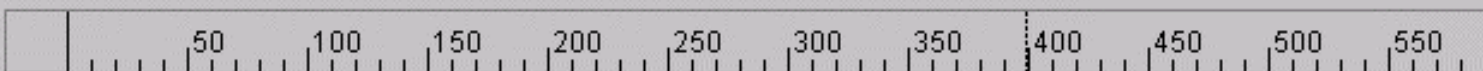
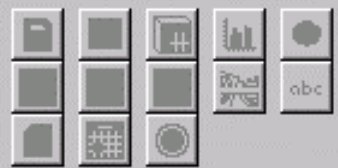


MySection

- ▶ Subsetters
- ▼ Rows/Down
 - Product...
- ▼ Columns/Across
 - Geography...
 - Revenue \$
- ▼ Statistics
 - SUM

Apply

TOC Data

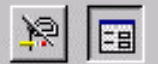


Toy Revenue Report

		West	Central	South	East
		Revenue \$	Revenue \$	Revenue \$	Revenue \$
		SUM	SUM	SUM	SUM
Wholesome Toys	Joyful Noise	109,022	105,632	75,450	72,990
	Vulcan	26,741	26,928	17,119	17,357
	ToyaSaurus	116,484	117,584	78,896	77,658
Violent Games		389,669	384,072	265,464	259,206
Sports Products		721,682	715,408	476,333	488,819

West.Revenue \$.SUM

Column	Sports Products
West.Revenue \$.SUM	721682



Ausblick

- Java und SAS (als Server) passen gut zusammen
- Es gibt mehrere Lösungen vom SAS Institute



Java-Initiativen von SAS Institute

- SAS/IntrNet-Komponenten
- *MetaSpace Explorer-Applet*
- *“thin client graphics”-Applet*
- *AppDevStudio*
 - *webAF (ehemals: Jazz IDE)*
 - *webEIS (ehemals: Swan Document Builder)*