

# Innovate2010

Die IBM Rational Software Konferenz



Udo Partsch – SoforTe GmbH

Implementierung von

Rational Developer for Systems z (RDz)

**- Erfahrungen -**

Intelligentes Business braucht intelligente Software und Systeme.



## Agenda

- **Vorstellung SoforTe**
- **Überblick Rational Developer for System z (RDz)**
- **Implementierung von RDz**
  - Anforderungen
  - Tool-Integration
  - Beispiele praxiserprobter Lösungen

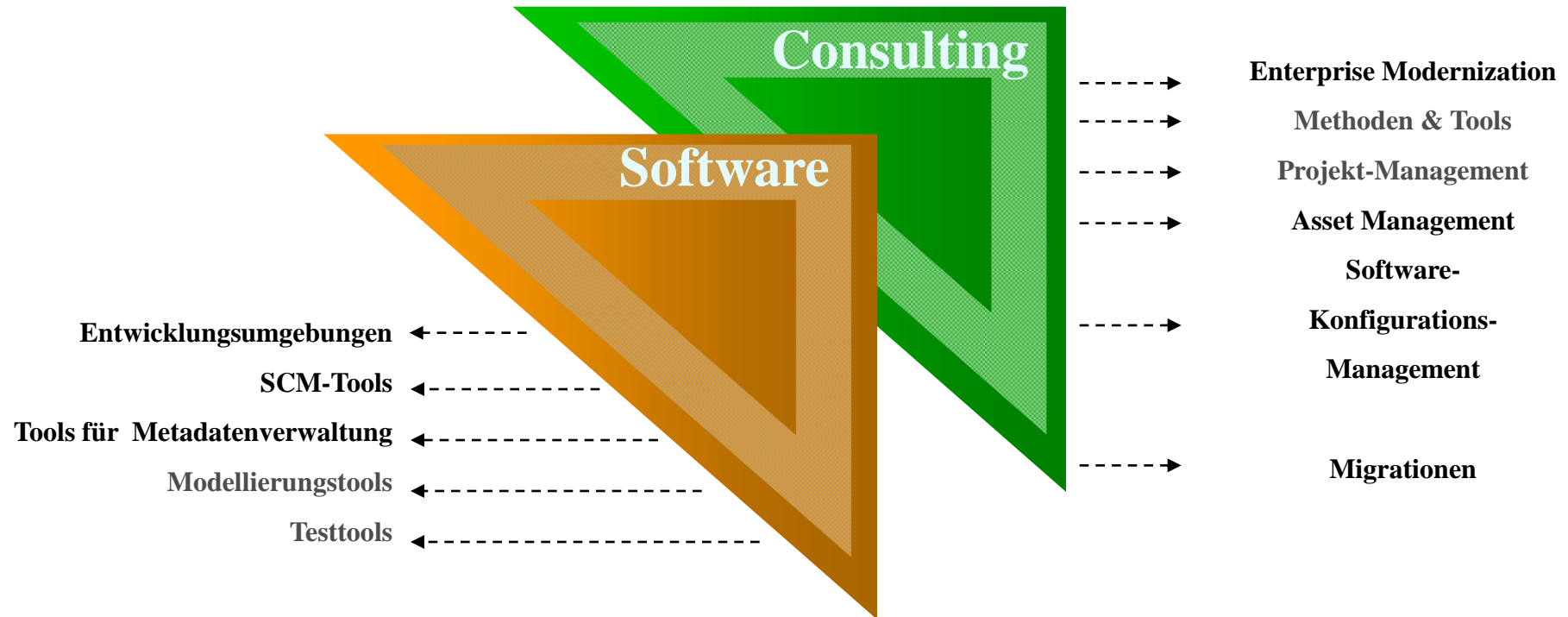


# SoforTe – Solutions for Teams

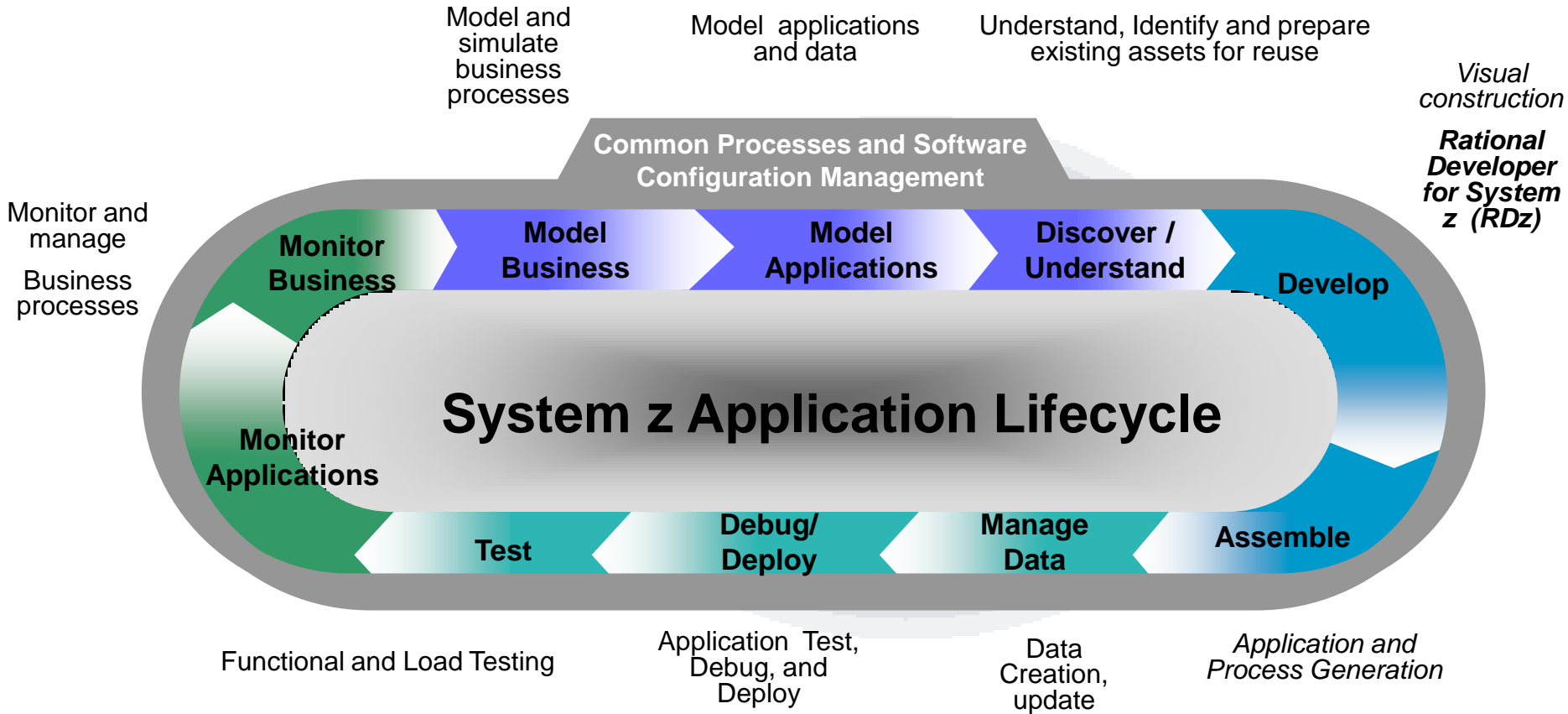
## Ihr Spezialist

für

## Softwareentwicklungsumgebungen



# Software Entwicklungsprozess

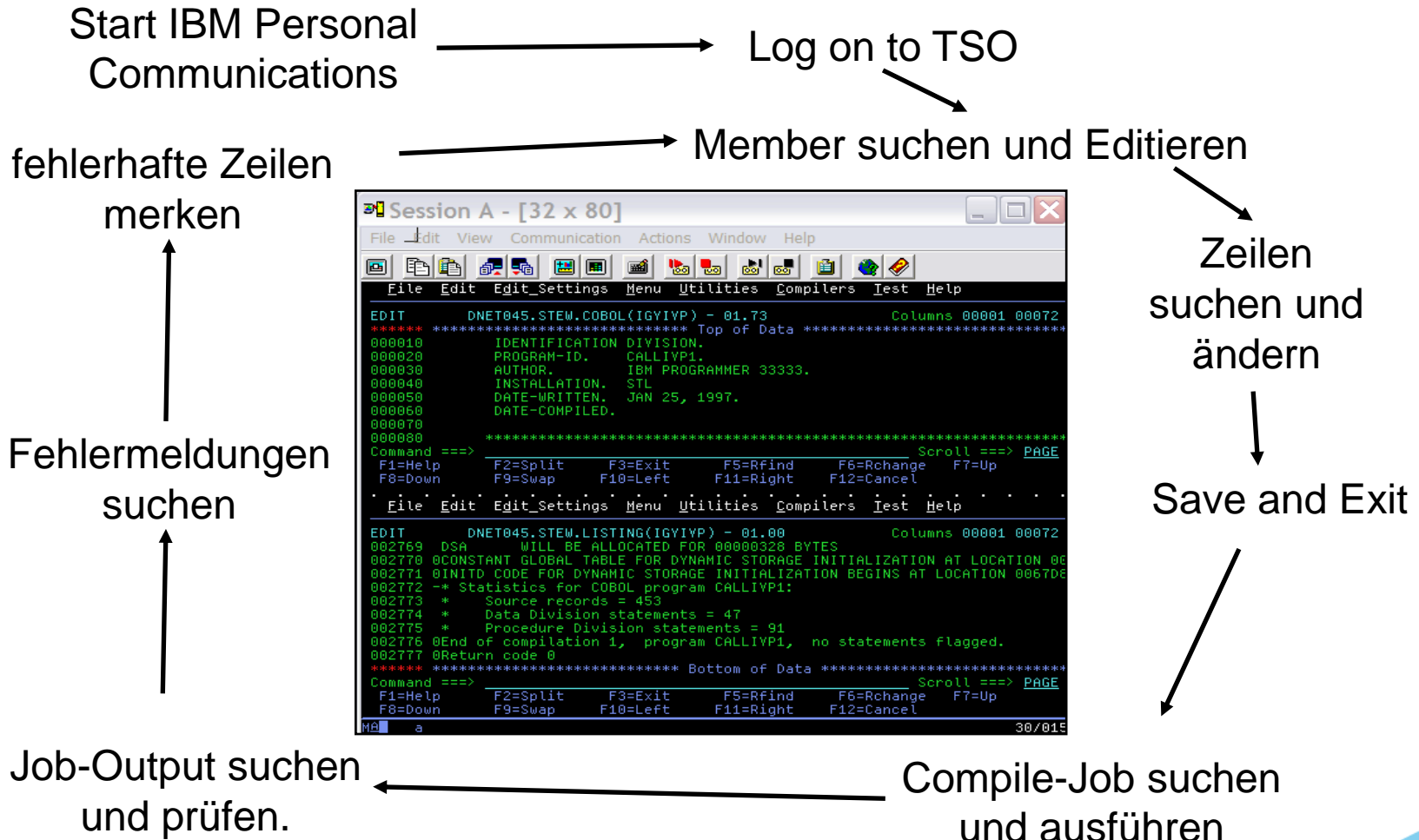


## Was ist RDz?

- **Eclipse basierte Anwendung**
  - Look & Feel wie unter Eclipse
    - Drag & Drop
    - Kontext-sensitive Menüs (rechter Mausklick)
- **Zielgruppe**
  - Mainframe Entwickler
- **Unterstützt die Entwicklung von z/OS-Anwendungen**
  - COBOL, PLI, HLASM, C/C++
  - TSO/Batch, CICS, IMS, DB2, USS, WAS
  - Sprach-sensible Editoren
  - Grafisches Debuggen (lokal und remote)
  - Grafische Editoren für CICS- und IMS-Masken
  - DB2 Stored Procedures – COBOL, PLI, Java, SQL
  - Web Services für CICS
  - Job-Monitor
  - z/OS-Datei-Explorer
  - .....
- **Mainframe-Entwicklung ohne Nutzung von ISPF**



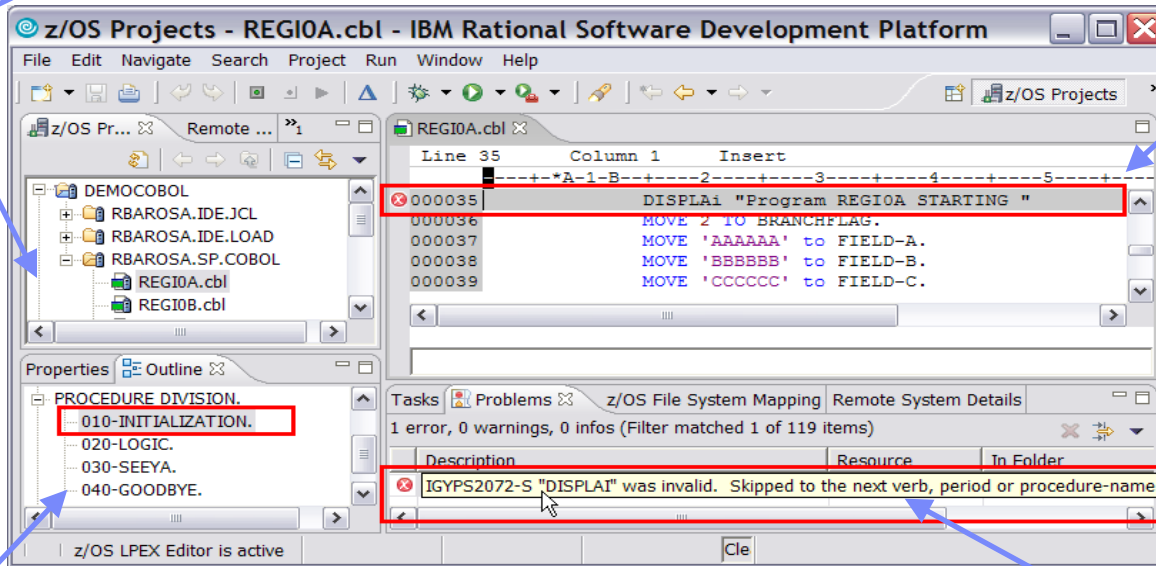
# Traditionelle Anwendungsentwicklung



# Rational Developer for System z

Syntax check

Edit Source

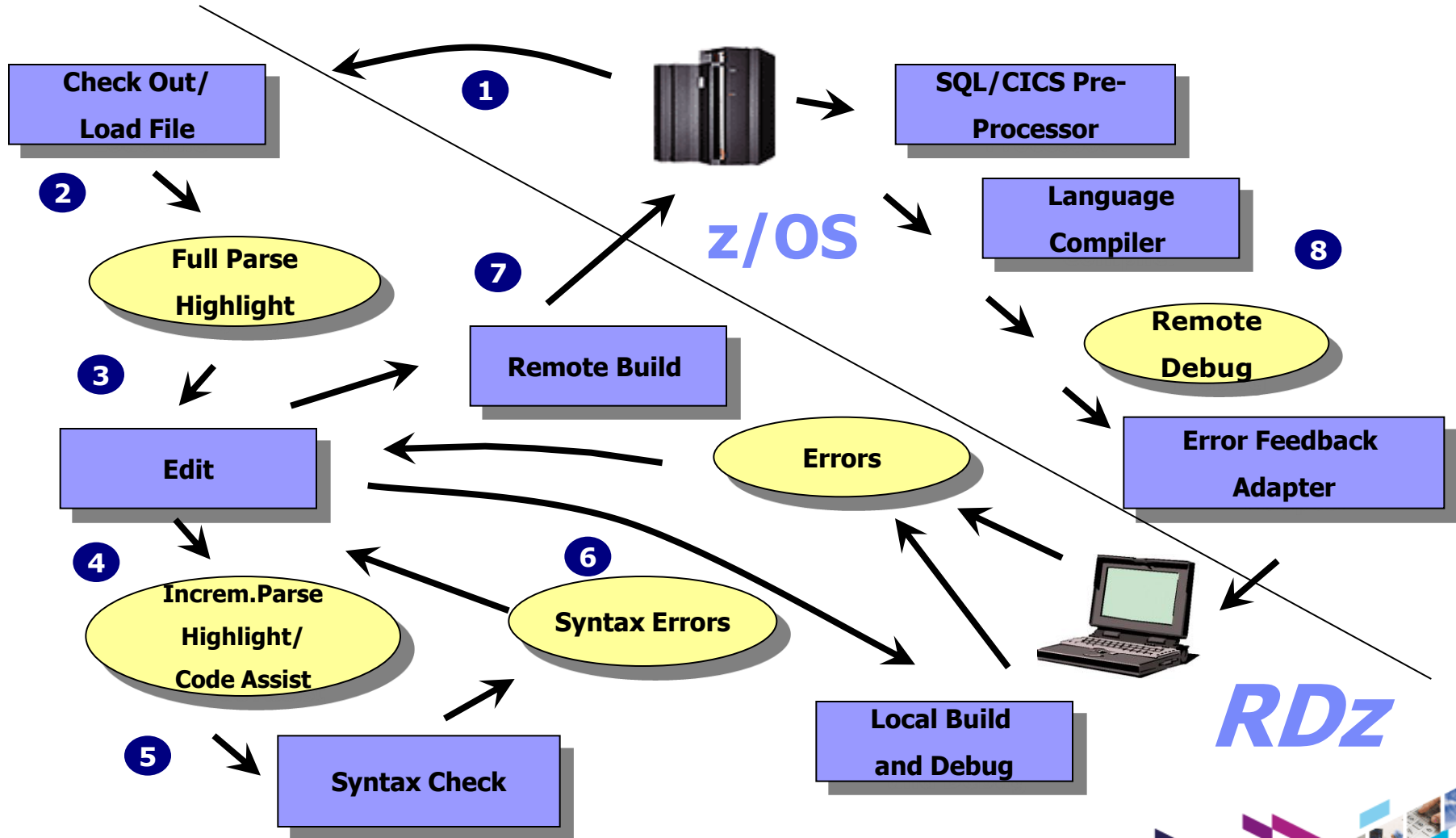


Outline View

Error Feedback



# RDz Entwicklungsprozess - allgemein



## Vorteile durch den Einsatz von RDz (im Vergleich zu ISPF)

- **Höhere Produktivität**
  - Funktionen sind direkt verfügbar (Kontextmenü)
  - Automatisierte Prozesse (Syntaxcheck, Error Feedback)
- **Einheitliche Entwicklungsplattform (Look and Feel)**
  - basierend auf Eclipse (Open Source)
  - Integration aller Entwicklungstools unter Eclipse
  - JAVA-Entwickler und traditionelle Entwickler (COBOL, ...) kommen sich näher
- **Vollständige Unterstützung moderner Anwendungsarchitekturen**
- **Hohe Akzeptanz von Eclipse bei Entwicklern**
  - Erhebliche Vorteile bei der Ausbildung neuer Mitarbeiter
- **Hohe Akzeptanz von Eclipse bei Toolherstellern**
  - Großes Toolangebot (Standardlösungen)
  - Flexible Einbindung individueller Lösungen
- **Weiterentwicklung von ISPF und ISPF-Tools stagniert**
- **CPU-Einsparung**



## Erforderliche Maßnahmen vor Einführung von RDz

### – Analyse des Ist-Zustands

- Welche vorhandenen Tools werden von den Entwicklern regelmäßig verwendet und müssen in RDz integriert werden?

### – Projekt aufsetzen zur Integration von Tools

- Integration des vorhandenen Software Konfigurationsmanagements:

- Source-Verwaltung
- Build-Prozess
- Freigabe-Prozess

- Integration zusätzlicher Pre-Compiler und Generatoren

- Integration von „Problem Determination Tools“ (Debugger, Abbruchanalyse)

- Integration wichtiger Eigenentwicklungen

### – Schulungskonzept

### – Roll-Out Plan



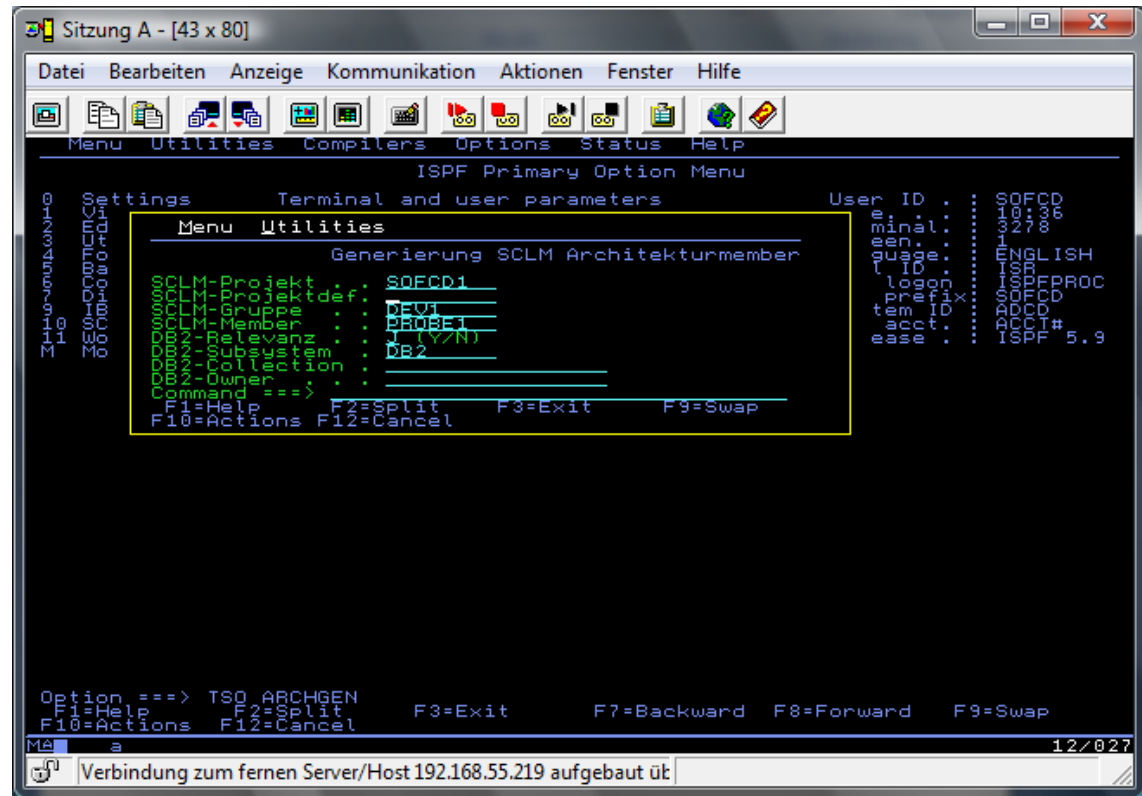
## Alternativen zur Integration von (Host-)Tools unter RDz

- **RDz Menü Manager**
  - Für einfache Tool-Integration ohne Programmierung
- **RDz Plug-In Entwicklung**
  - Sehr flexible Möglichkeiten zur Integration von Tools durch Plug-In Programmierung
- **HATS**
  - Zur Integration sehr komplexe Host-Tools durch GUI-fizierung
- **RDz CARMA Interface**
  - Zur Integration eines Host-SCM-Tools
- **SoforTe TAURUS**
  - Zur Integration von Host-Tools und Plug-Ins in eine konfigurierbare Benutzeroberfläche



# ISPF Beispieldialog

- Beispiel: Generierung von Architekturdefinitionen
- Wie kann dieses ISPF Tool (Eigenentwicklung) unter RDz integriert werden?



# ISPF Beispieldialog integriert mit RDz Menü Manager

## – RDZ Menü Manager

- grafisches Interface zur Erweiterung von RDz Kontextmenüs, Definition des Tool-Aufrufs und einfacher Dialoge

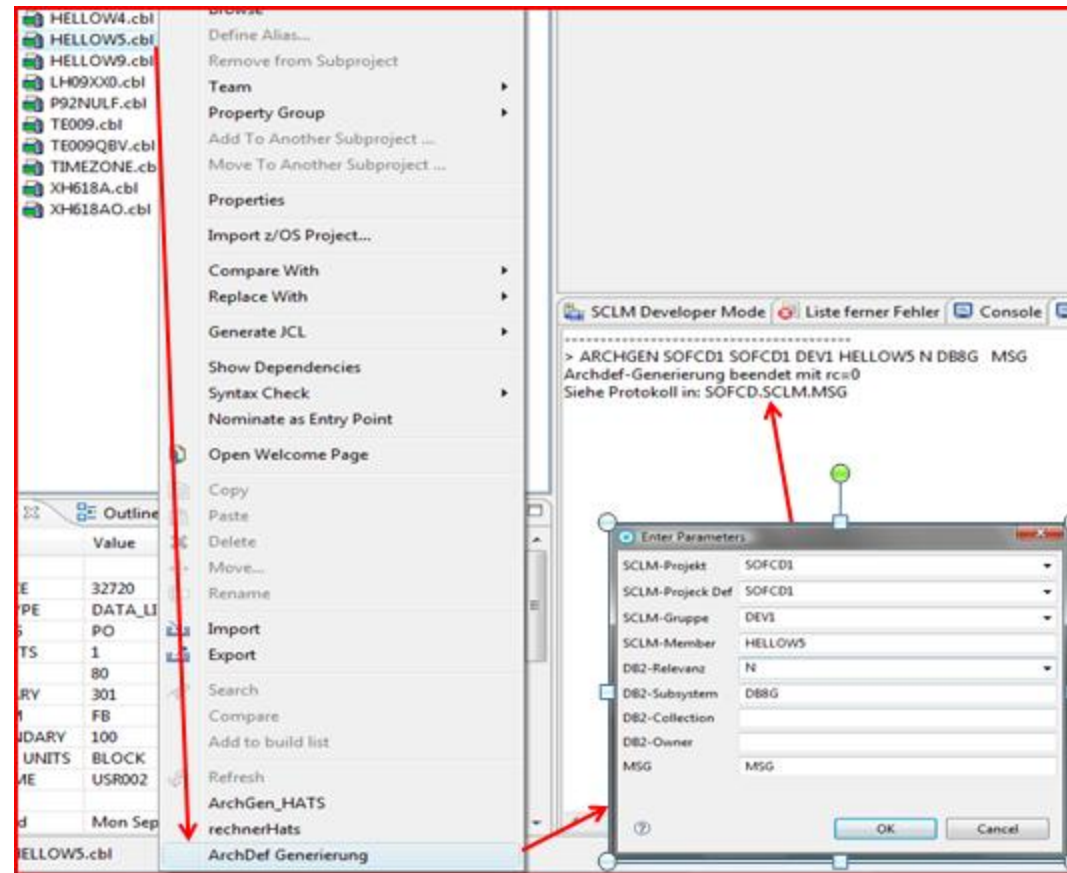
## – Vorteile:

- Sehr einfach zu erstellen
- Keine Java Programmierkenntnisse erforderlich

## – Nachteile:

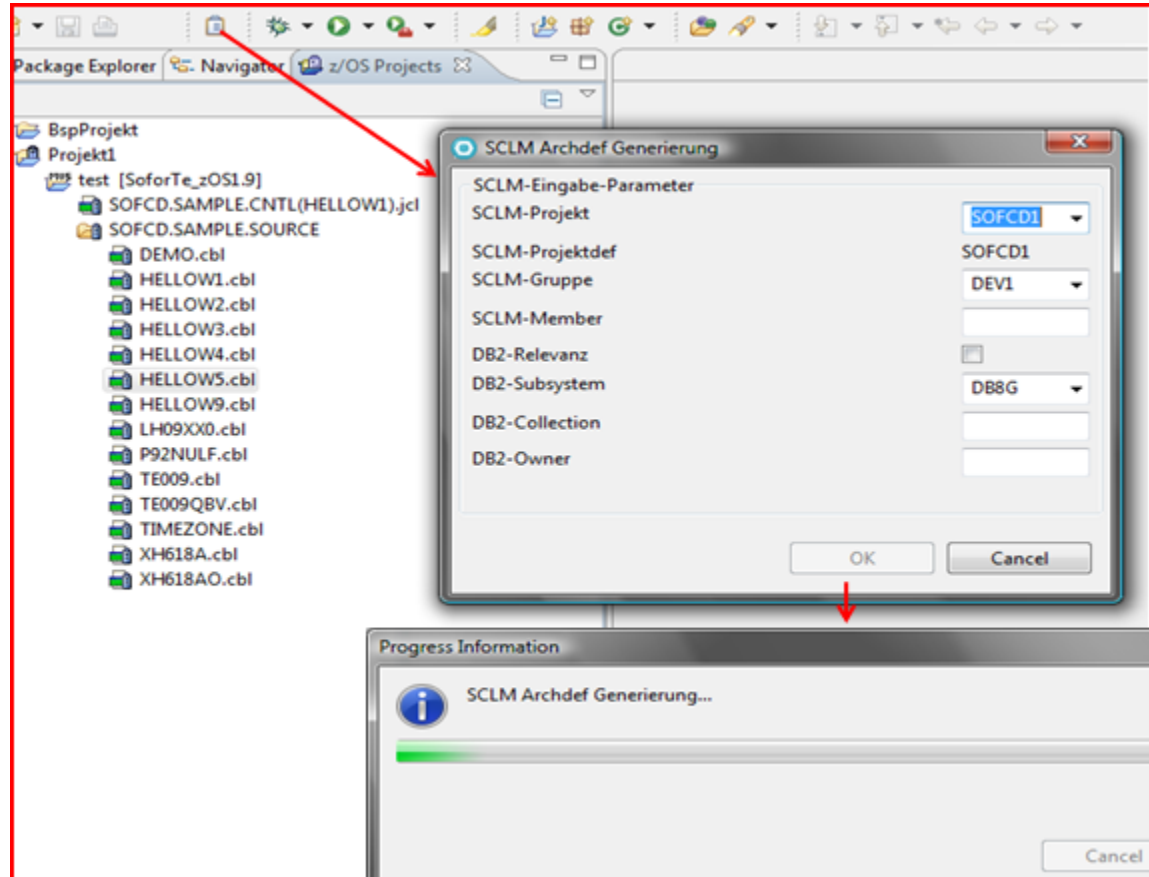
- Nicht alle RDz Kontext-Menüs können erweitert werden
- Eingeschränkter Funktionsumfang
- Nicht geeignet zur Integration komplexer SCM-Tools wie
  - SCLM, ChangeMan, Endeavor

## – Aufwand: 1 Std.



## ISPF Beispieldialog integriert mit Plug-In Entwicklung

- **RDz Plug-In Entwicklung**
  - Tool-Integration durch Eclipse Programmierung und Verwendung von RDz APIs
- **Vorteile:**
  - Höchste Flexibilität
  - Gestaltung einer individuellen Benutzeroberfläche
- **Nachteile:**
  - Sehr gute Java Plug-in Programmierkenntnisse kombiniert mit ISPF-Skill erforderlich
  - Sehr hoher Ausbildungsaufwand
  - Sehr hoher Programmieraufwand
  - Lange Projektlaufzeit
- **Aufwand: 3 Personentage**



# ISPF Beispieldialog integriert mit HATS

## - HATS

- Guifizierung einer beliebigen 3270-Anwendungen (IMS, CICS, ISPF,...)
- Generierung einer Eclipse RCP (Rich Client Plattform) Anwendung zur Integration in RDz

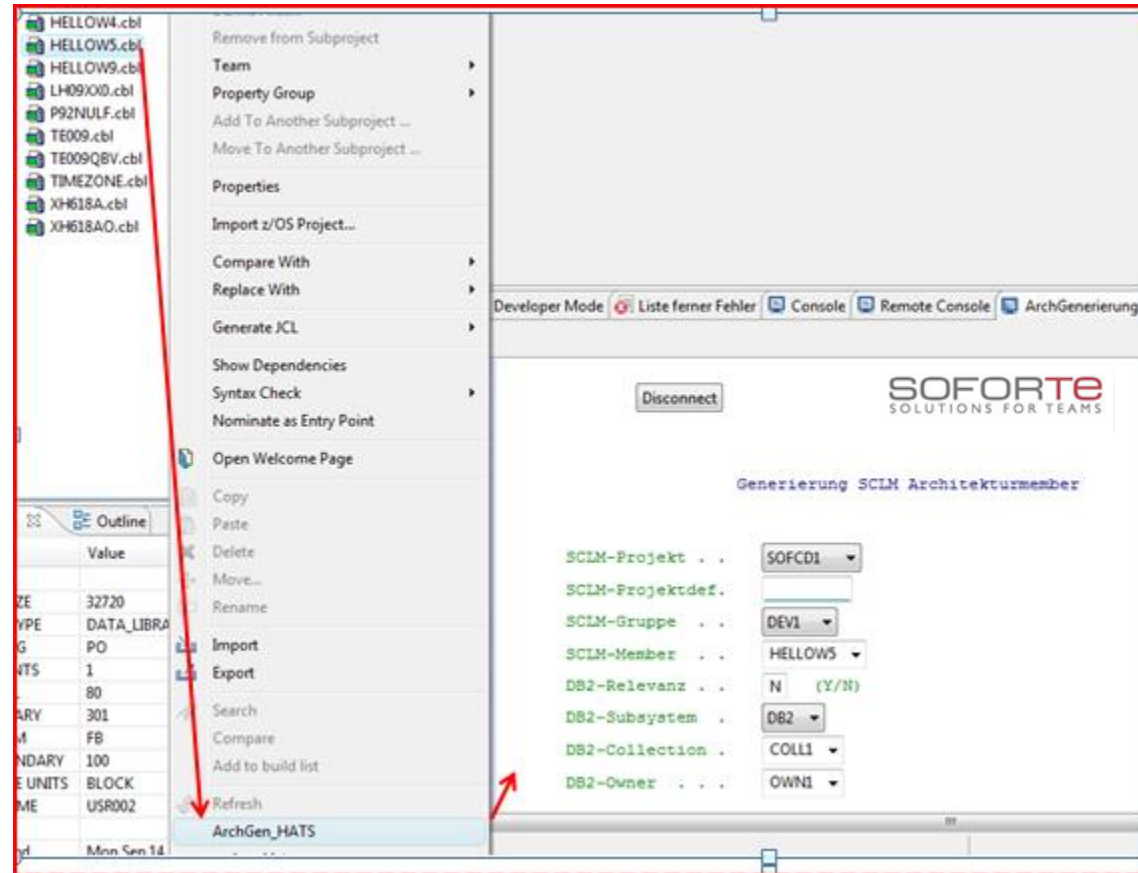
## - Vorteile:

- Keine Anpassung der ISPF Anwendung erforderlich
- Eclipse Look and Feel erreichen ohne Programmierung
- Kann verwendet werden, wenn die Source des ISPF-Tools nicht existiert oder nicht wartbar ist.

## - Nachteile:

- Mehrere HATS Anwendungen können nicht gleichzeitig aktiv sein (Anwendung beginnt mit TSO-Logon)
- Benutzer darf nicht gleichzeitig unter TSO angemeldet sein
- Höherer Wartungsaufwand

## - Aufwand: 6 Std.



# ISPF Beispieldialog integriert über RDz CARMA

## – RDz CARMA Interface

- Generische Schnittstelle zur Anbindung eines z/OS SCM-Systems an RDz

## – Vorteile:

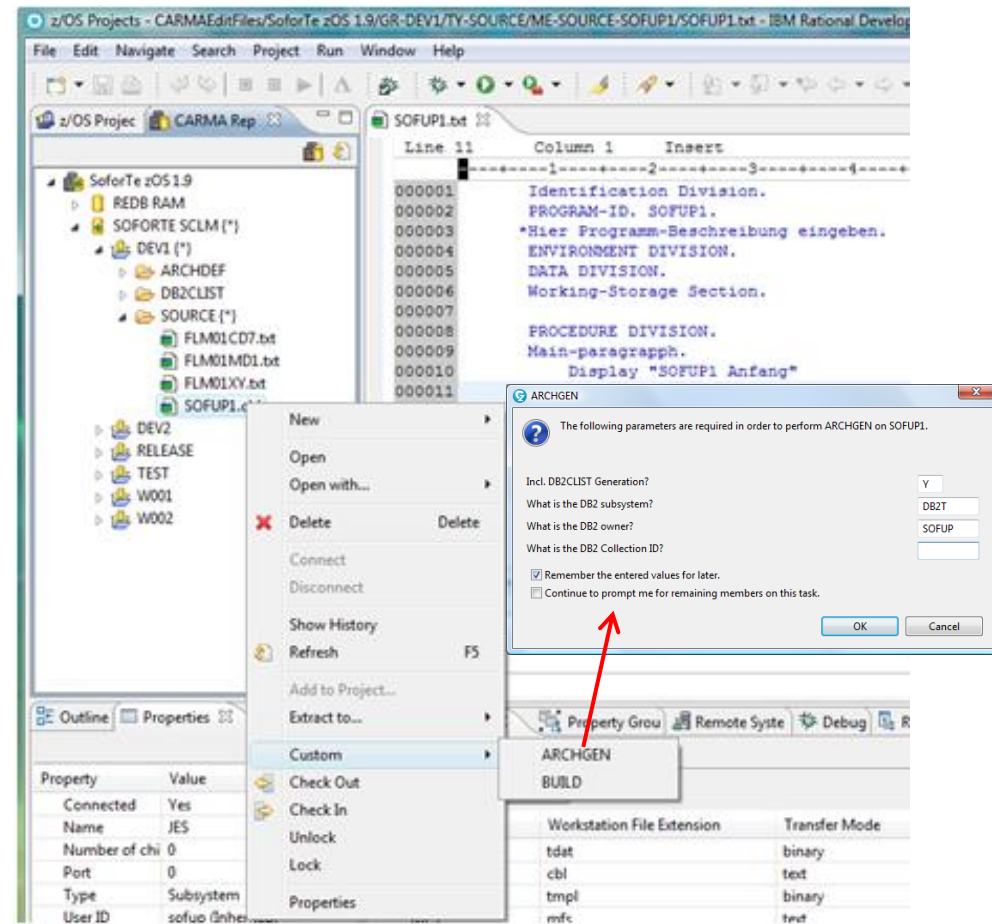
- Vorhandenes Wissen (REXX, ISPF, ...) kann genutzt werden
- Bei Verwendung des generischen Clients:
  - Keine Plug-in Entwicklung und kein Java Wissen erforderlich

## – Nachteile:

- Nur zur Einbindung von Host SCM-Systemen gedacht
- Einige RDz Funktionen sind nicht verfügbar, wenn die CARMA View verwendet wird
  - Kein Error Feedback
  - Kein local Syntaxcheck
- Hoher Aufwand, wenn der CARMA Client nicht verwendet werden kann und stattdessen mit dem CARMA-API Plug-Ins entwickelt werden müssen

## – Aufwand:

- 4 Std. zur Integration des Tools in eine auf CARMA basierende SCM-Lösung



# ISPF Beispieldialog integriert über SoforTe TAURUS

– **SoforTe TAURUS for RDz**

- Flexible Konfiguration eines Benutzer-Interfaces unter RDz
- Flexible Konfiguration der Tool-Integration
- Modellierung des Benutzer-Interfaces und der Tool-Einbindung

– **Vorteile:**

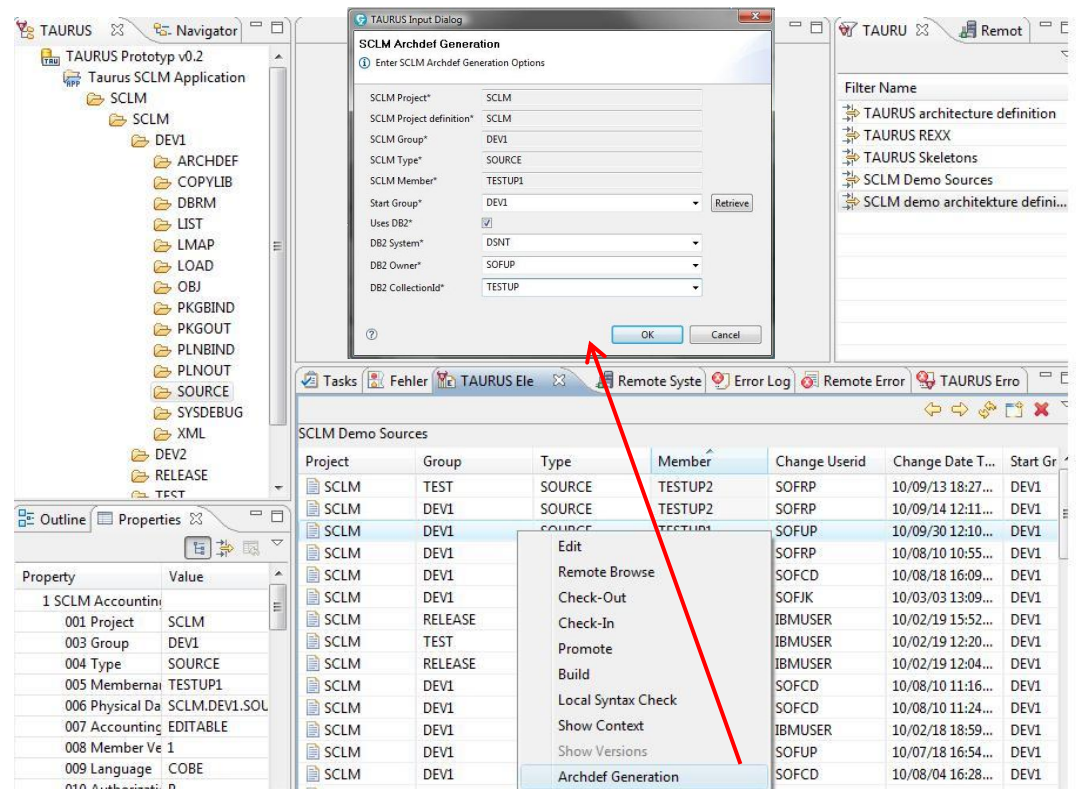
- Einfach zu erstellen
- Vorhandenes Wissen (REXX, ISPF, ...) wird genutzt
- Keine Plug-in Entwicklung und kein Java Wissen erforderlich
- Drastische Reduzierung des RDz Implementierungsaufwands

– **Nachteile:**

- Kostenpflichtiges Add-On zu RDz

– **Aufwand:**

- 30 Min. zur Integration des Tools in eine konfigurierte TAURUS Anwendung



## Zusammenfassung RDz Tool Integration

- **Menü Manager**
  - Für einfache Tools
  - Eingeschränkte Funktionalität
- **RDz Plug-In Entwicklung**
  - Sehr flexible Integration
  - Hoher Entwicklungsaufwand, hoher Wartungsaufwand
  - Erfordert Entwickler mit sehr gutem Plug-In und Host-Skill
- **HATS**
  - Für komplexe Host-Tools
    - bei schlecht wartbarer Source
    - wenn keine Source verfügbar
  - GUIifizierung der 3270-Anwendung für Eclipse
- **CARMA**
  - Integration eines Host-SCM-Tools
  - Host Konfiguration
  - Host Programmierung im Vordergrund
  - i.d.R. zusätzlicher Plug-In Entwicklungsaufwand erforderlich
- **TAURUS**
  - Flexible Integration von Host-Tools und Plug-Ins in RDz
  - Vollständige Konfiguration der Toolintegration und des Benutzer Interfaces
  - Minimierung Eigenentwicklung und Wartungsaufwand



## Erfahrungsaustausch zu RDz und Toolintegration

### – Guide Share Europe (GSE)

#### – GSE Enterprise Modernization Konferenz

- Europäische Konferenz
- RDz ist derzeit Schwerpunktthema
- Erfahrungsaustausch der Anwender (ca. 100 Teilnehmer)
- Diskussionen mit den IBM Labors
- 2 Tagungen pro Jahr a 3 Tage
- Infos:

<http://www.gse.org/ABOUTGSE/GSEWorkingGroups/tabid/115/Default.aspx>

#### – GSE SCM Konferenz

- Deutsche Arbeitsgruppe zum Thema Software Konfigurationsmanagement (SCM)
- Integration von SCM-Tools in RDz und/oder RTC werden intensiv diskutiert
- Erfahrungsaustausch der Anwender (ca. 30 Teilnehmer)
- Diskussionen mit den IBM Labors
- 2 Tagungen pro Jahr a 3 Tage
- Infos: <http://www.scm-guide.de/>

