

**VP2000 -  
Gigaflops im 7·500-  
Bereich**

## **Ist:**

VP-Familie nur mit MSP/MVS zu betreiben

## **Ziel:**

VP-Familie mit 7·5xx/BS2000 betreibbar

## **System VSP:**

- MSP-Erweiterung bzgl. Vektor-Unit
- Keine interaktive Komponente
- MSP-Kopplung via JES/MAS, JES/E oder NJE
- Programmentwicklung im MSP (FEP)

## Wege zum BS2000-Anschluss:

- BS2000 auf VP-Hardware
- BS2000 als FEP

## Implementierung der 2. Variante mit zwei stark getrennten Themenbereichen:

- Heterogene BS2000-VSP-Kopplung (RHRK)
- Portierung des FORTRAN77-Systemes (SIEMENS)

## Randbedingungen:

- BS2000 verhält sich MSP-analog
- Neue BS2000-Komponenten FORTRAN77, Filetransfer, Jobtransfer
- Einfache Userschnittstelle, SDF
- Siemens-Produkt
- Auf andere Sprachen erweiterbar
- Software nicht privilegiert
- Compile im BS2000-PLAM-Format
- Möglichst wenig VSP-Software
- Einfache Administrationschnittstelle
- Zugangskontrolle im BS2000
- Wartbarkeit
- Manuale

## Benutzerinterface:

- Leistungen auf Kommandoebene
- Für erfahrene Anwender offener VSP-Zugang

## Kommandos:

- COMPILE-FORTRAN-SOURCE
- LINK-VECTOR-PROGRAM
- {ENTER|CALL}-VECTOR-PROGRAM
- ASSIGN-VECTOR-SYSDTA
- ASSIGN-VECTOR-SYSLST
- MODIFY-VECTOR-PROTECTION
- OPTIMIZE-VECTOR-FORTRAN-SOURCE
- SHOW-VECTOR-JOB-STATUS
- CANCEL-VECTOR-JOB

## ... Fortsetzung Kommandos:

- {ENTER|CALL}-VECTOR-JOB
- SET-VECTOR-FILE-LINK
- CREATE-VECTOR-FILE
- CONVERT-VECTOR-FILE
- DELETE-VECTOR-FILE
- LIST-VECTOR-CATALOG
- PRINT-VECTOR-FILE
- VERIFY-VECTOR-FILE
- TRANSFER-VECTOR-FILE

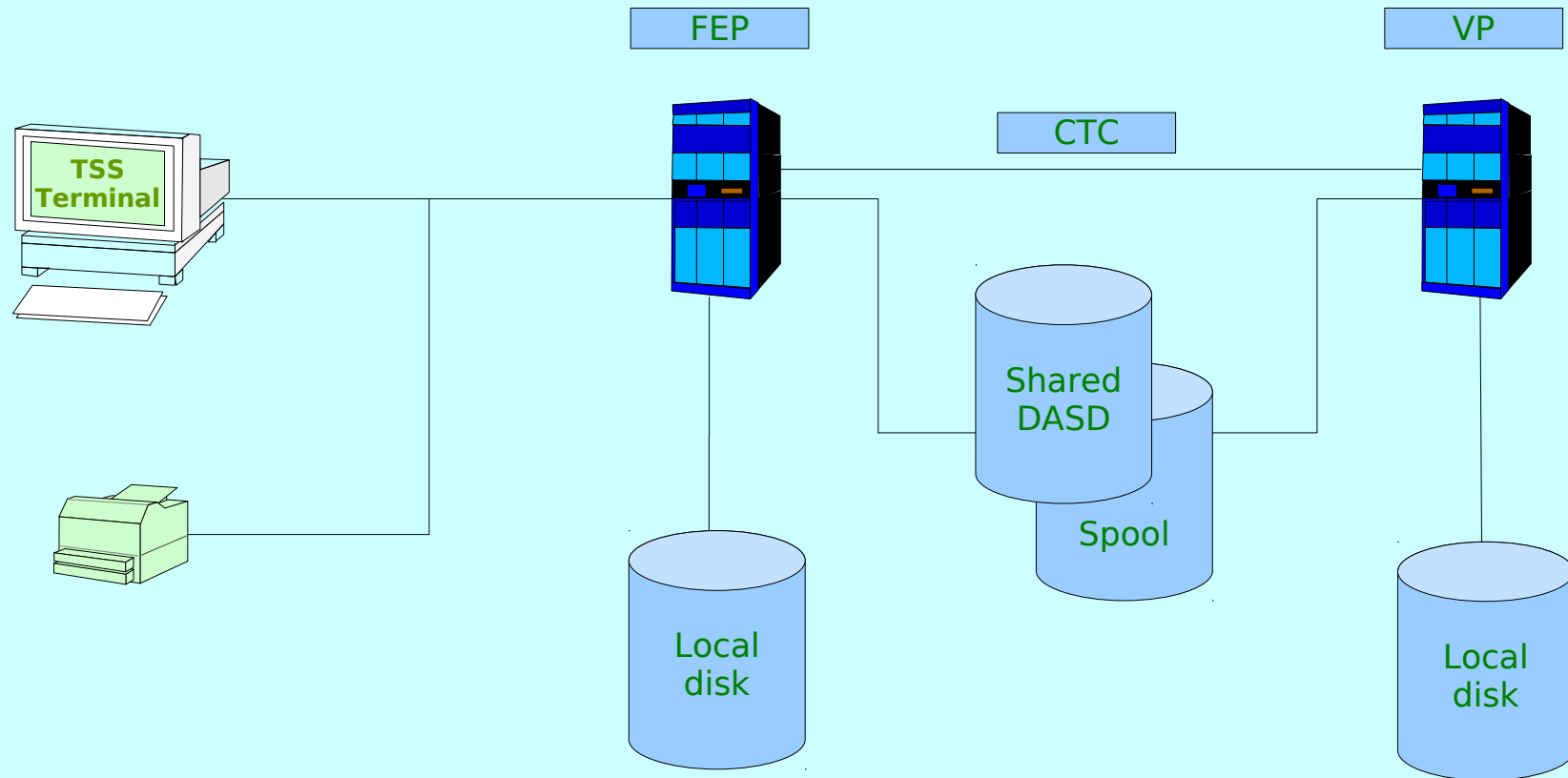
# **Verbund-Komponente**



## Möglichkeiten:

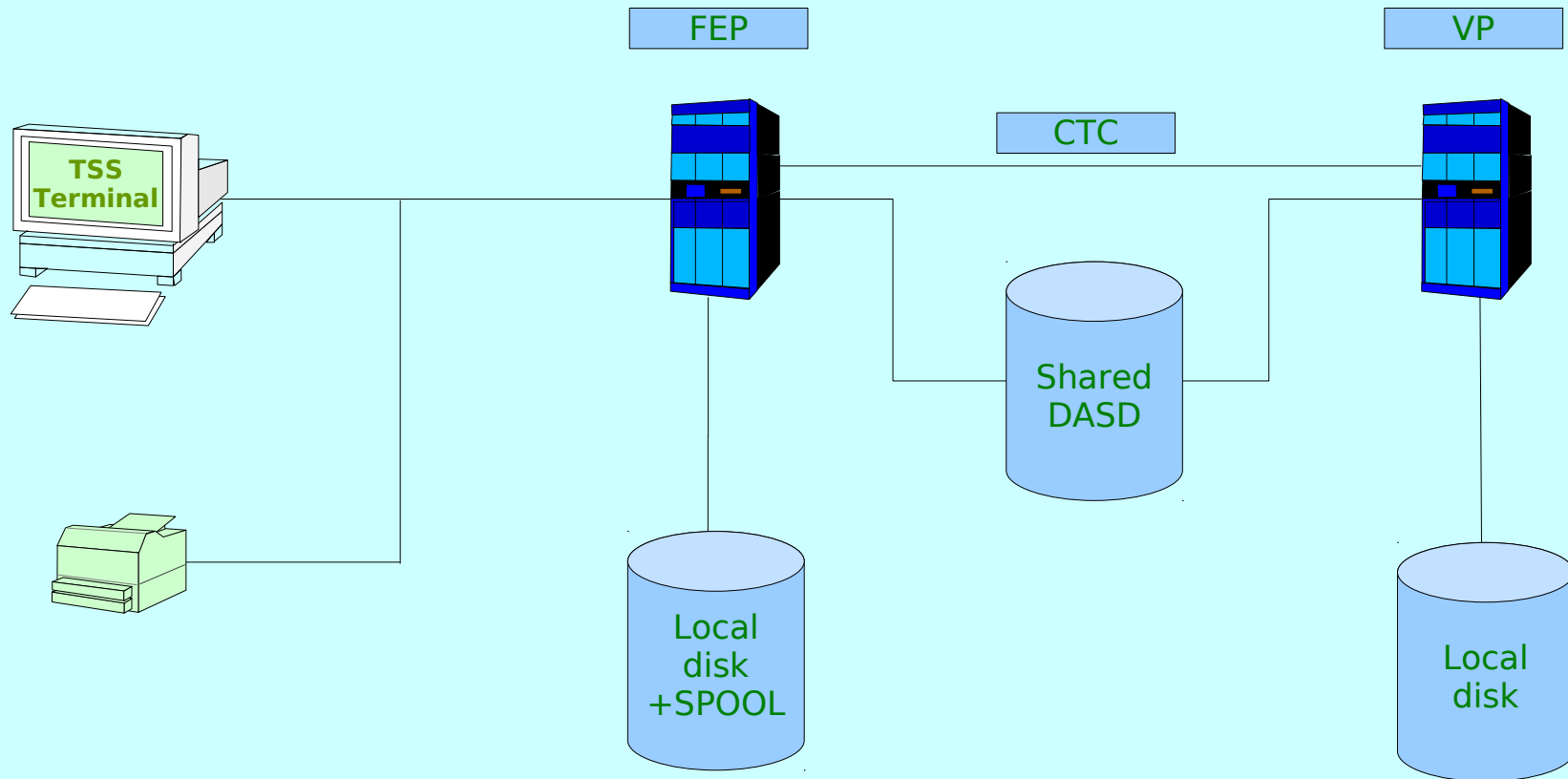
- NJE über CTC
- JES/MAS
- NJE

## NJE über CTC

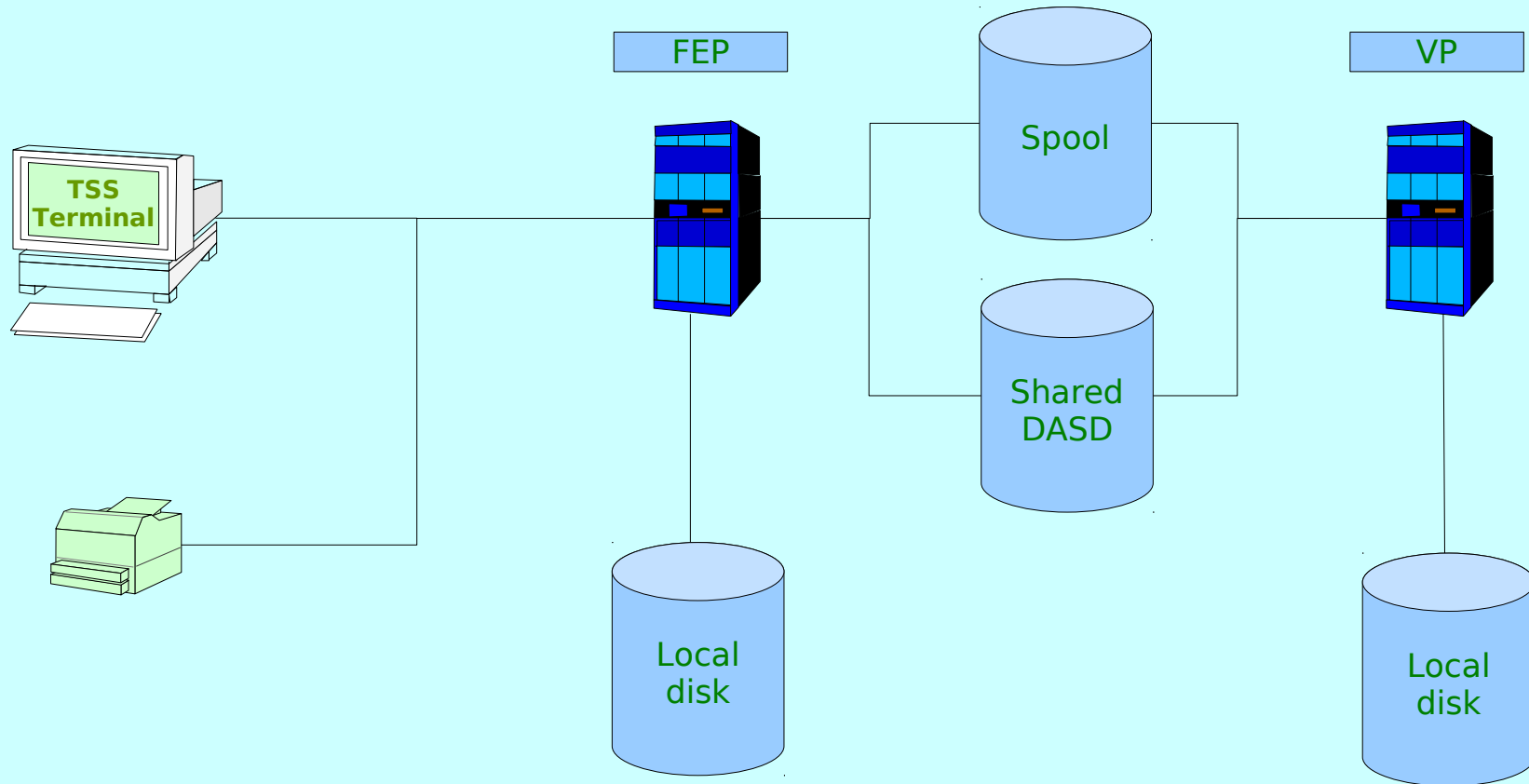


# VP2000 – Gigaflops im 7·500-Bereich

## NJE



## JES/MAS



# **Entscheidung für JES/MAS**

## Komponenten von JES/MAS:

- Auftragseingabe
- Konvertierung Steuersprache
- Auftragsplanung
- Ausführungskontrolle
- Ausgabeplanung
- Ergebnisausgabe
- Auftragslöschung

**Kommunikation über JOB-Queue  
(HSP und Checkpointfile auf Shared DASD)**

## Im BS2000 Realisierung von:

- Checkpoint-Prozessor
- Internal Reader
- Output Writer
- Kommando-Prozessor (JES-Kommandos)
- Internal Reader und Output Writer als parallele Subtasks über DSSM

**Konstruktion gestattet  
Zusammenarbeit  
mehrerer BS2000,  
mehrerer MSP (oder MVS)  
und mehrerer VSPs über  
einen Shared Spool**



# **File-Zugriff**

## Mögliche Implementierungen:

- MSP-Zugriffsmethoden im BS2000
- Direkte Dateikopie
- JES/MAS-Verwendung

## Realisierung über File-Copy (COPY2000)

- Anpassung an CKD
- Nur Transfer kompletter Files
- Voraussetzung: Fileattribute abbildbar
- Abprüfungen im VSP-Standardjob
- Vom BS2000 aus keine Längenänderungen
- Einbettung in VSAM-Katalog-Struktur
- Zieldateien komplett überschrieben
- Korrekturversuche nur auf niederster Ebene

# **Verbund-Basis**

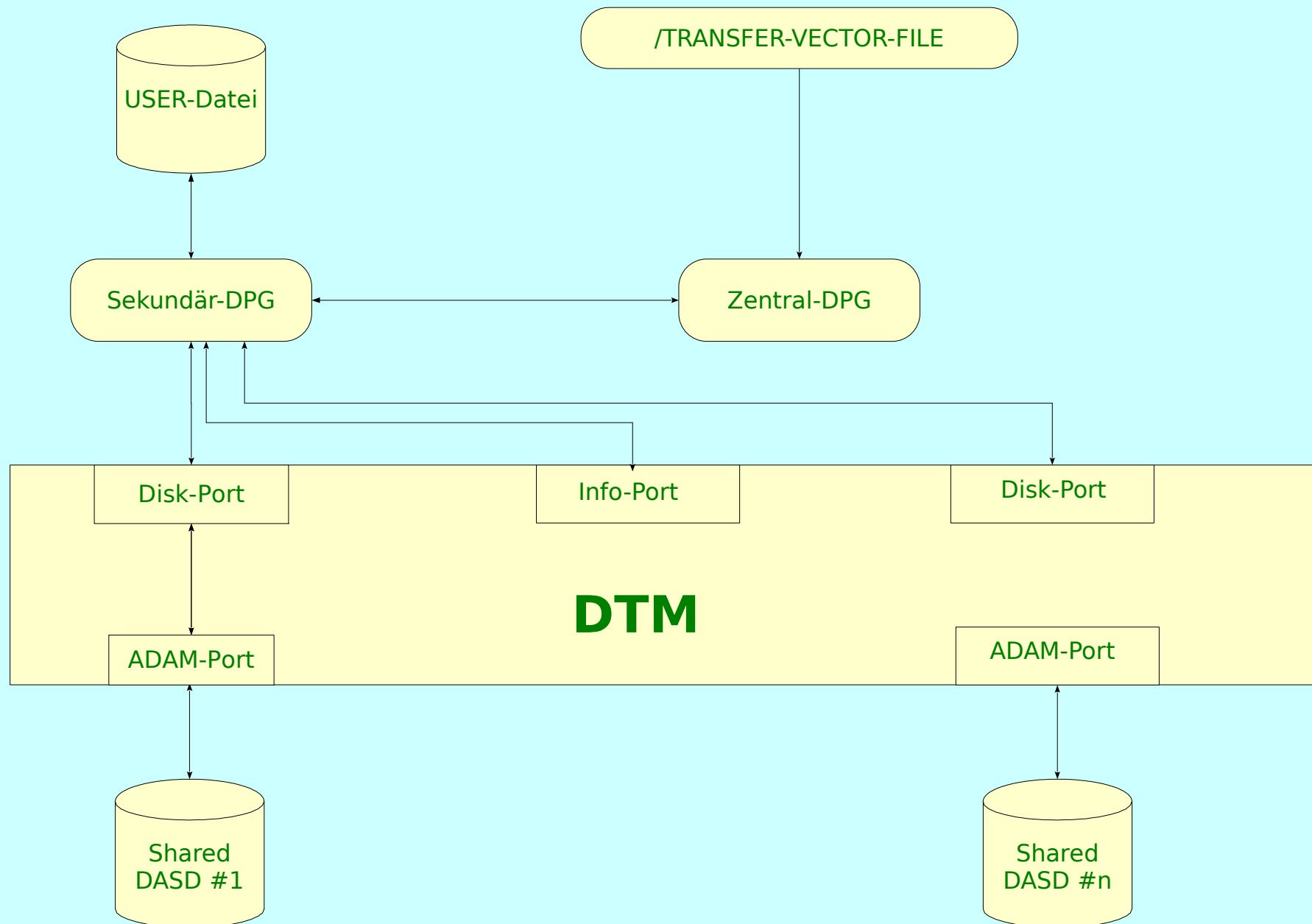
- Kommunikation über Platte (Typ: 3380/3880) mit **ADAM** (DTM)
- Darüber die Funktionen JES und FC zentral und User-spezifisch (mehrfach)
- Kommunikation über **FITC**)

## Konstruktionsziele:

- Saubere Aufgaben- und Schichtentrennung
- Datenschutz und -Sicherheit
- Bearbeitung von BS2000-Files aus Bändern/Wechselplatten
- Reduzierung der Kontakte USER  $\leftrightarrow$  FC/JES

**Datenfluss DPG ↔ DTM**

# VP2000 – Gigaflops im 7·500-Bereich





# **FORTRAN77-SYSTEM**

## MSP-Komponenten von FORTRAN77:

- FORTRAN77-Compiler mit RTS
- FORTRAN77/VP-Compiler mit RTS
- Interactive Vectorizer (VECTUNE)
- FORTUNE (dynamische Programmlaufanalyse)
- TESTFORT77 (interaktive Fehlersuche)
- DOCK/FORT77 (interaktive Fehlersuche im Fullscreen-Mode)
- SSLII-Bibliothek
- SSLII/VP-Bibliothek

# Vektorisierung ohne Quelländerung!

## BS2000-Anwender:

- Prüfe, ob FORTRAN77-Standard (bei Portierung vorhandener Programme), ggffls. Teile umschreiben
- Übersetzung mit BS2000-Variante FORTRAN77-PLAM-Bibliotheken
- Binden mit RTS und SSLII-Bibliotheken über TSOSLNK
- Test mit AID/IDA oder portiertem TESTFORT77, Optimierung mit FORTUNE
- Übersetzung mit FORTRAN77/VP
- Binden mit RTS und SSLII/VP (TSOSLNK)
- Umwandeln im BS2000 in einen VSP-Lademodul

## Technik der Compiler-Migration - Compiler

- MSP-Sprache = SPL/100, stark unterschiedlich zu SPL4
- SPL/100-Compiler übersetzt i.A. nach Assembler
- Portierung SPL/100 nicht möglich
- Änderung SPL4 zu umfangreich
- Technik soll die Portierung verschiedener Versionen automatisieren

## Implementierungssprachen für:

- Test/Tuning-Tools: SPL/100
- Laufzeitsystem: Assembler
- SSLII: FORTRAN77
- SSLII/VP: Fortan77

## Mehrere Möglichkeiten:

- SPL/100-Quellen im BS2000: nicht möglich
- SPL/100-Quellen nach Assembler-Quellen ist möglich, letztere sind aber anzupassen (System-Makros auf BS2000-Makros **im BS2000** abbilden), Quell-Änderungen bei Zugriff von FORTRAN77 auf Systemtabellen
- Portierung des Compiler-Objectcodes. Dabei **im MSP** System-Makros auf BS2000-Makros abbilden

# **Wahl Variante Nr. 2**



## Aufwand:

- Compiler enthält ca. 250.000 Quellzeilen
- Systemabhängiger Teil verwendet ca. 50 Makros
- Prompter-Umstellung TSS-BS2000-Dialog
- Maincontroller und Compilercontroller
- FORTRAN77 und FORTRAN77/VP ähnlich
- Im RTS ca. 60 Makros
- VSAM und High Speed I/O nicht portabel

## Vorgehensweise:

- Modellcompiler
- Vektorleistung im BS2000
- Programmentwicklung im BS2000
- Tuning im BS2000