

# **Brugerhåndbog**

## **Del VI**

### **Konvertering af modeller i PCIM**

*September 1999*

## **Indhold:**

|                                       |   |
|---------------------------------------|---|
| Introduktion                          | 1 |
| Installation af PCIM-udviklingspakken | 2 |
| Konvertering                          | 3 |
| Modellen                              | 4 |
| Data                                  | 5 |

# Konvertering af modeller i PCIM version 7.04

## 1. Introduktion

I det følgende forklares kort hvordan udviklingspakken til PCIM anvendes. Papiret er opdelt i 4 mindre afsnit. Det andet afsnit beskriver hvordan konverteringsmodulet installeres og sættes op på harddisken. Det tredje afsnit er en brugervejledning til konverteringsmodulet.

Det fjerde afsnit giver teknisk beskrivelse af opstillingen af modelligninger til brug for konverteringsmodulet. Det femte og sidste afsnit beskriver kort, hvordan man kan oprette en databank til den nye model.

For at køre FORTRAN oversætteren og PCIM kræves mindst en 386-maskine med tilhørende co-processor (80387) og mindst 4 MB RAM.

Udvikling af modeller til PCIM kræver en LAHEY EM/32 (version 5.0 eller højere) FORTRAN oversætter. Desuden kræves Spindrift Utility og Spindrift Windows utilities (Oversætter og utilities kan fx. købes hos Ravenholm Computing i Lyngby). Oversætteren og tilhørende utilities installeres typisk i direktoriet F77L3 med en række tilhørende underdirektorer.

Skanner-programmet, som er en del af udviklingspakken, kan kun afvikles under et 32-bit styresystem.

## 2. Installation af PCIM-udviklingspakken

Hvis ADAM, maj 1998, og FORTRAN oversætteren mv. er installeret med standardopsætningen, så kan konverteringsmodulet installeres automatisk med den medfølgende install.bat fil. Hvis harddisken hedder C, og diskette-drevet hedder A, så behøver brugeren blot sætte disketten i diskette-drevet og taste:

```
A:\INSTALL A: C:
```

Herved kopieres konverteringsmodulet ind i direktoriet C:\ADAM\KONVERT. Det kan være nødvendigt at tilpasse opsætningen af konverteringsmodulet, hvis det fx. er nødvendigt at angive en søgesti til memory-drivere el.lign. Dette gøres i filen KONVERT.BAT. Filen kan redigeres i en DOS-editor. Find linien:

```
PATH c:\dos;c:\f77l3;c:\f77l3\bin;
```

I denne linie kan alle de direktorer, som indgår i den almindelige søge-sti, tilføjes.

Hvis konverteringsmodulet ikke skal installeres i en standardopsætning af ADAM, maj 1998, og FORTRAN oversætteren, så anbefales det at anvende følgende retningslinier:

- Udviklingspakken bør installeres i et separat direktorie. Den kunne fx. placeres i et direktorie, der hedder <PATH>\KONVERT.
- Filen KONVERT.BAT skal tilpasses den lokale opsætning. Indholdet af bat-filen beskrives i afsnittet om konvertering nedenfor.

Det anbefales at afprøve konverteringsmodulet på ADAMs formelfil. Kopier MAJ98.FRM over i et nyt direktorie og skift over til dette direktorie. Konverter nu modellen med følgende kommando:

```
<PATH>\KONVERT maj98
```

Hvor <PATH> i standardopsætningen er C:\ADAM\KONVERT. Konverteringen tager nogle minutter. Til sidst dannes blandt andre en exe-fil, MAJ98.EXE, som bør være identisk med den af modelgruppen leverede fil af samme navn.

### 3. Konvertering

Konverteringen består af to dele. Først skannes formelfilen, <model>.frm, og TSP-formlerne omsættes til FORTRAN. Dernæst oversættes FORTRAN programmet og linkes til en eksekverbar fil, <model>.exe.

Konvertering kan kun foregå fra det direktorie, hvor modellens formelfil, ligger.

Selve konverteringen af modellen sker ved kaldet:

```
<PATH>\KONVERT <model>
```

Kommandoen KONVERT kalder en BAT-fil (der ligger i udviklingspakken). BAT-filen foretager automatisk de nødvendige trin i konverteringen. De centrale dele af KONVERT.BAT ser således ud:

```
SKANNER6 %1 c:\adam\konvert\fcim6000 F77L3
-Sc:\f77l3\lib\spin3,c:\f77l3\lib\graph3
SET 386link=-stack 90000 -pack -stub runb -maxd 6000

CALL IBCOM

PAUSE Der må ikke være fejlmeddelelser i det følgende
TYPE ibcom.err | more
```

Hvis brugeren ikke har valgt standardopsætningen, så skal han, når udviklingspakken er installeret, udskifte følgende stier:

```
c:\adam\konvert med det direktorie, hvor udviklingspakken en placeret.
c:\f77l3\lib      med det direktorie, som F77L3 har til sine biblioteker.
```

De enkelte kommandolinier i BAT-filen har følgende betydning:

```
SKANNER7 <MODEL> <PCIM-direktorie> -S<andre direktorier>
    Konvertering af modellens TSP-formler til FORTRAN. PCIM-direk-
    toriet er udviklingspakkens direktorie.
SET 386link=-stack 90000 -pack -stub runb -maxd 6000
    Angiver nogle optioner til linkerens.
    Programmet skanner7 danner FORTRAN programmer, og en BAT-
    fil: IBCOM.BAT
IBCOM
    Kalder FORTRAN-oversætteren og linkerens.
    Så skulle den nye version være klar. Hvis der er fejl i oversættelsen,
    så vil fejludskrifterne kunne findes vha:
TYPE IBCOM.ERR | more
    Denne udskrift må ikke vise nogen fejl.
```

Hvis der er fejl, vil det typisk være syntaksfejl i en formel. Fejllisten kan desværre ikke direkte bruges til finde formelen, der er fejl i. Det fejlbehæftede udtryk, som fremgår af IBCOM.ERR vil typisk se således ud:

$$b(332) = a(23)+23.2 \text{ osv}$$

Det fejlbehæftede udtryk skal derefter findes i FORTRAN-kildeteksterne. Disse ligger i filer, der hedder IB<tal>.FOR. Den relevante fil's navn fremgår af filen IBCOM.ERR.

Ved søgning gennem FORTRAN-kildeteksten, kan det fejlbehæftede udtryk findes. Når det fejlbehæftede er lokaliseret, så kan relationens endogene variabel aflæses lige oven over. Derefter kan man finde den relevante formel i modellens formler. Dernæst kan fejlen findes og rettes.

Når man er overbevist om, at der ikke er fejl, kan alle midlertidige filer slettes. det sker med kommandoen:

```
DELETE IB*.*
```

Når konverteringen er vel overstået, så vil direktoriet rumme en række PCIM-filer; disse er:

|             |                                                           |
|-------------|-----------------------------------------------------------|
| <MODEL>.EXE | PCIMs exe-fil, som er den konverterede model              |
| <MODEL>.FRM | Den originale formel-fil, som er input til konverteringen |
| <MODEL>.FRN | Den nye formel-fil, som produktet af konverteringen       |
| <MODEL>.NRM | En beskrivelse af processen fra "frm" til "frn"           |
| <MODEL>.VAR | En variabelliste                                          |
| F77L3.EER   | Fejlmeddelelser                                           |

#### 4. Modellen

Modellens formler skal ligge i en fil med navnet <MODEL>.frm . Hver model bør have sit eget direktorie. Dette er ikke noget krav, men det er en god skik, der letter vedligeholdelsen.

En model ser typisk sådan ud:

```
<De simultane formler>

AFTER $

<Eftermodel, der løses efter hver periode>

AFTER2 $

<Eftermodel, der løses efter alle perioder er løst>
```

Den sidste eftermodel bør kun anvendes, hvis det er helt nødvendigt, dvs hvis systemet ikke virker ellers.

PCIM accepterer de TSP-formler, som også NASS (engang i tidernes morgen) accepterede. *Formlerne* ser normalt sådan ud:

```
FRML <Formelnavn> <Variabel>           = UDTRYK $
FRML <Formelnavn> DIF(<Variabel>)        = UDTRYK $
FRML <Formelnavn> LOG(<Variabel>)         = UDTRYK $
FRML <Formelnavn> DLOG(<Variabel>)        = UDTRYK $
```

*Formelnavnet* kan ikke vælges frit. Første tegn i formelnavnet kan have betydning. Følgende tegn i første position har betydning for konverteringen af ligningen:

##### 1. position.

- \_ Formelnavnet vil blive brugt til at styre skanningen. De efterfølgende positioner i formelnavnet kan bruges til fx. at pålægge relationen justeringsled og eksogeniseringskonstruktion.
- Y Venstreside variablen opfattes som eksogen. Derfor skal fx bagvendte JLED beregnes i en Y formel. Forskellen er, at bl.a. venstreside variablen aldrig vil blive initialiseret til foregående periodes værdi.
- Z Den pågældende relation kan dæmpes ved løsningen.
- F Denne relation kan ændres dynamisk i kørslen vha fix kommandoen.

Det anbefales at lade formelnavnet starte med understreg, <\_>. Hvis formelnavnet starter med <\_> vil navnet, som sagt, blive anvendt til at styre skanningen. De enkelte positioner efter <\_> har forskellig effekt.

## 2. position er fri.

Her anbefales det at bruge følgende tegn:

- I For identitetsligninger
- D For definitionsligninger
- S For stokastiske relationer
- G For øvrige relationer

## 3 . og 4. position styrer indsættelse af J-led.

- J\_ Giver indsættelse af et almindeligt JLED.  
Dvs.  $\text{endogen} = \text{relation} + J < \text{endogen} >$   
og i eftermodellen  $J < \text{endogen} > = \text{endogen} - \text{relation}$
- JD Giver præcis den samme struktur.  
Men JLED'et bliver kaldt  $JD < \text{endogen} >$ .  
Denne type kan anvendes til JLED i ændringsrelationer.
- JR Giver et relativt JLED.  
Dvs.  $\text{endogen} = \text{relation} \cdot (1 + JR < \text{endogen} >)$   
og i eftermodellen:  $JR < \text{endogen} > = \text{endogen} / (\text{relation}) - 1$
- \_ Giver intet JLED

## 5. position styrer eksogenisering.

- D Betyder eksogenisering.  
Dvs.  $\text{Endogen} = \text{relation}$  bliver til:  
 $\text{endogen} = (\text{relation}) \cdot (1 - D < \text{endogen} >) + Z < \text{endogen} > \cdot D < \text{endogen} >$   
og i eftermodellen:  $Z < \text{endogen} > = \text{endogen}$
- \_ Betyder ingen eksogenisering.

## 6. position styrer om formelen skal kunne FIX'es.

- F Muliggør fixning. Det svarer til et F i første position, hvis der ikke er et \_ i formelnavnet.
- \_ udelukker fixningsmuligheden.

## 7. karakter styrer dæmpningen.

- Z Muliggør dæmpning. Det svarer til et Z i første position, hvis der ikke er et \_ i formelnavnet.
- \_ udelukker dæmpning.

Formelnavnenes længde kan være indtil 7 karakterer. Men de kan være kortere. Man behøver ikke at angive alle 7 karakterer, hvis man kun har brug for de første. Eksempelvis gælder følgende:

|                  |                    |            |
|------------------|--------------------|------------|
| <u>I</u> _____   | kan forkortes til  | <u>I</u>   |
| <u>SJ</u> _____  | kan forkortes til  | <u>SJ</u>  |
| <u>SJD</u> _____ | kan forkortes til  | <u>SJD</u> |
| <u>SJR</u> __Z   | kan ikke forkortes |            |

Hvis ikke formelnavnet starter med <\_>, så kan det strengt taget vælges frit med en længde på indtil 8 karakter. Vælges første tegnet Y, Z eller F, så har det som beskrevet ovenfor betydning for konverteringen af ligningen. Udover disse muligheder anbefales det, at vælge det første tegn blandt følgende muligheder:

- I For identiteter
- D For definitionsligninger
- S For stokastiske relationer
- G For øvrige relationer

For alle relationer, bortset fra dem, hvor første tegn er <\_>, anbefales det, at formelnavnet konstrueres således:

Formelnavn: <første tegn><endogen variabel>

Dette anbefales af hensyn til PCIMs muligheder for at dekomponere formelen.

*Variabelnavne* kan vælges frit. Dog skal man være opmærksom på, at ingen variabelnavne må være på mere end 8 karakterer. Man skal også være opmærksom på, at automatikken tilføjer prefixer til eventuelle justeringsled, målvariabler og eksogeniseringsdummys. For disse endogene variabler kan variabelnavnet ikke være længere end 6 eller 7 karakterer.

Relationsoperatorer skal skrives på en anden måde:

|      |                    |
|------|--------------------|
| .GT. | skal skrives som > |
| .GE. | >=                 |
| .LT. | <                  |
| .LE. | <=                 |
| .EQ. | ==                 |

Udtryk, hvor der anvendes relationsoperatorer skal omgives af parenteser, eks:

$a = b * (c > d)$

Der må ikke så at sige anbringes relationsoperatorer inden i hinanden.

$a = b * (c > (x * (f > g)))$  FORBUDT !!

Der må ikke forekomme #’ere i modellen.

Alle heltal i modellen omsættes til real-tal. Det indebærer, at maskinen ikke accepterer konstruktioner af formen:  $Y^{*-3}$  hvis Y kan antage negative værdier. det kan dog løses ved at skrive  $Y^{*int(-3)}$

Formler, der er baseret på division med 0, er helt YT. De indebærer nemlig, at der genereres "Not a Number" og alle relationer hvor denne NaN indgår får værdien NaN, osv, hvorefter hele løsningen er gået komplet i fisk.

## 5. Data

Data kan overføres fra en anden PCIM-bank. Hvis den gamle bank er dannet med samme programversion, dvs med samme antal max. variabler, kan en anden models bank umiddelbart læses. Fra PCIM version 7.04 er databank-formatet bagud kompatibelt - dvs at gamle databanker kan læses af nye PCIM versioner.

En ny databank kan derfor oprettes i PCIM meget nemt<sup>1</sup>:

```
READ hit<mm><åå>
...
Indlæs eller importør data for nye variabler
...
HDG Databank til ny model <model> pr <dato>
WRITE nybank
```

Arbejdsarealet fyldes op med data ved indtastning fra prompten eller fra fx. en anden version af PCIM eller fra AREMOS.

Data indtastes fra prompten (eller i en CMD-fil) med UPD-ordren eller dannes med GENS-ordren:

```
UPD nyvar1 = værdi1 værdi2 ...
GENS nyvar2 = f(nyvar1,..) $
```

Data fra andre PCIM-versioner kan udskrives ved hjælp af ordren

```
PIPE nyedata.cmd
UPDPRT = #ALL
PIPE con
```

i en kommando-fil. Denne kommando-fil kan anvendes til at indlæse data i den nye version:

```
NYEDATA
```

---

<sup>1</sup> I andre tilfælde skal der oprettes en helt ny databank. Det gøres med ordren BANK, fx:

```
BANK expand 40
```

PCIM vil nu spørge om 40 ID-variabler, og du skriver fx:

```
1983 1984 1985 ...
```

indtil der er indlæst 40 tal, evt på flere linjer. Det kan anbefales at lave denne del som en kommandofil, idet det erfaringsmæssigt er vanskeligt at indtaste 40 tal korrekt.

Data fra AREMOS udskrives i AREMOS ved hjælp af EXPORT ordren i TSD format og indlæses ved hjælp af TSDREAD i PCIM.

TSDREAD nyedata.tsd

Ordren TSDREAD kan også bruges til at oprette en databank. Hvis TSDREAD ordren bruges umiddelbart efter opstarten af PCIM, så vil arbejdsbanken automatisk blive oprettet.

Arbejdet med databanker og data i PCIM er beskrevet i brugerhåndbogens Del II, afsnit 2.