Computer Aided Parallelization Tools CAPTools

Richard Luczak

ASC MSRC PTES Lead

University of Tennessee

June 5, 2000

Outline

- Requirements
- History
- What is CAPTools ?
- Features (main components, input codes, design, dependencies)
- Parallelization (steps, session, start CAPTools, compile/link/run)
- Examples (applications, acceleration, benefits, info)

Requirements

- DoD users need help in parallelizing codes
- Accessing/harnessing new scalable systems
- Tools to convert serial Fortran code to portable parallel source form:
  - rapidly
  - almost automatically
  - accurately
  - efficiently and scalably
- Using either OpenMP directives or MPI calls or any combination

History

- **Jan 98 - Government sponsored visit from Prof. Mark Cross, Director of Research at the Univ of Greenwich, London, UK**
- **Aug 98 - collaborative evaluation of CAPTools defined and proposed**
- **Oct 98 - PET funding for tech support approved**

History (cont.)

- **Dec 98 - collaboration between UG and NASA Ames incorporated into ASC project through use of OpenMP directives**
- **Apr 99 - MPI/OpenMP beta release provided to ASC**
- **Evaluation status - UGC 99**
- **3 workshops by UG**

What is CAPTools?

A software environment to transform Fortran source code from scalar to parallel source form with message passing MPI or OpenMP directives in an interactive *semi-automatic* way

INPUT | **CAPTools** | **OUTPUT**

Fortran77 code → **Transformation to parallel code** → **Fortran77 + MPI calls**

OR

+ OpenMP Directives

Features

- **Main components**
- **Input codes**
- **Design**
- **Dependencies**

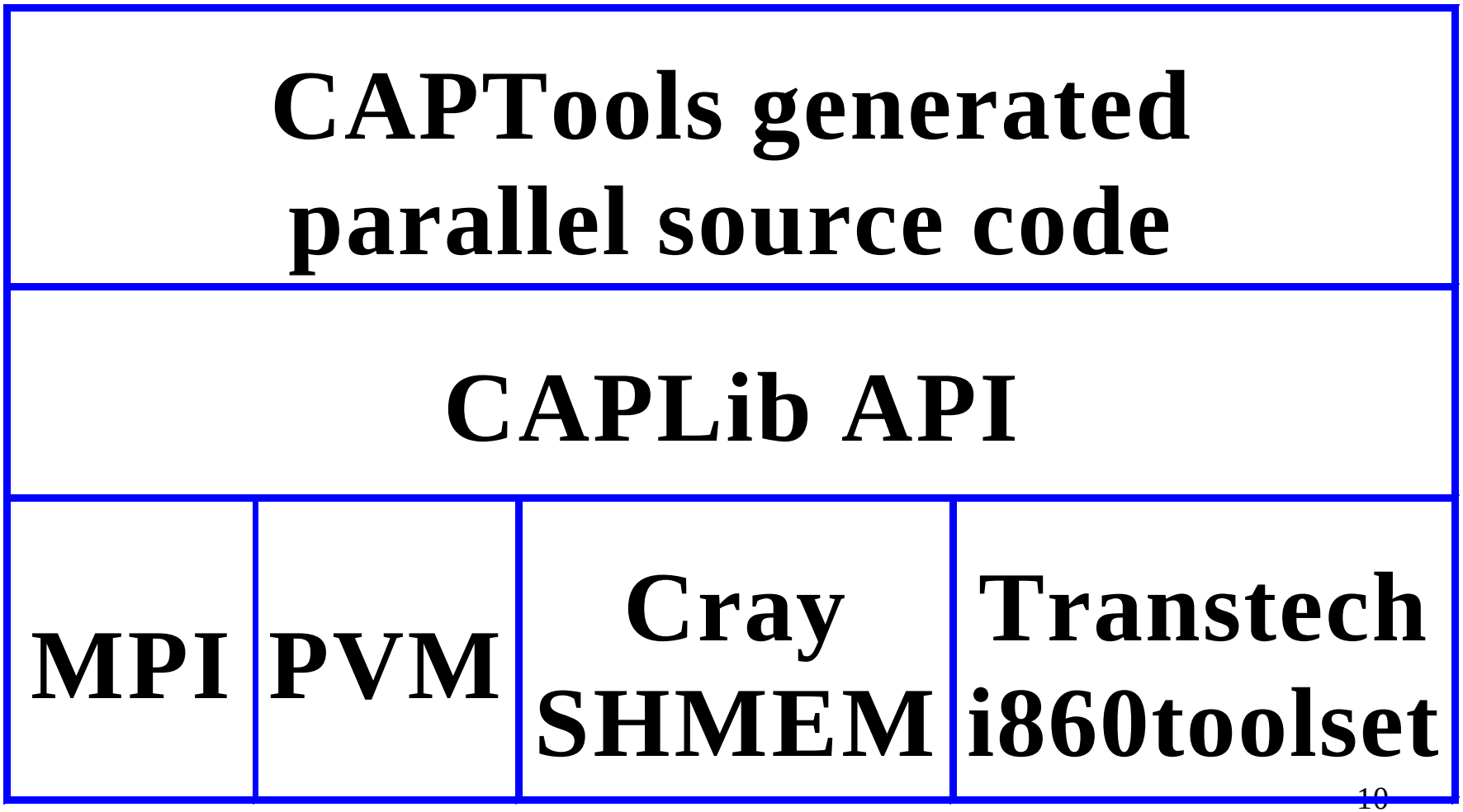
Main components

- State-of-the-art interprocedural dependence analysis techniques
- In depth analysis using symbolic algebra methods
- User supplied knowledge to assist parallelization
- GUI interactive front end
- Automatic user directed array partitioning
- Automatic parallel source code generation

Input codes

- **Fortran 77 (Fortran 90/95 in development)**
- **Multiple program files**
- **Include statements**
- **Conditional compilation directives should be preprocessed**

Design


CAPTools generated parallel source code
CAPLib API
MPI
PVM
Cray SHMEM
Transtech i860toolset

Dependencies

- **True**

```
A(I) = ...
... = A(I)
```

- **Control**

```
IF (A(I) .EQ. 5) THEN
   A(I) = A(I) + B(J)
```

- **Anti**

```
... = A(I)
A(I) = ...
```

- **Output**

```
A(I) = ...
A(I) = ...
```

Parallelization

- Steps
- Session
- Start CAPTools
- Compile/link
- Run

Steps

- **Load Fortran 77 code**
- **Volunteer knowledge to the system**
- **Perform dependence analysis**

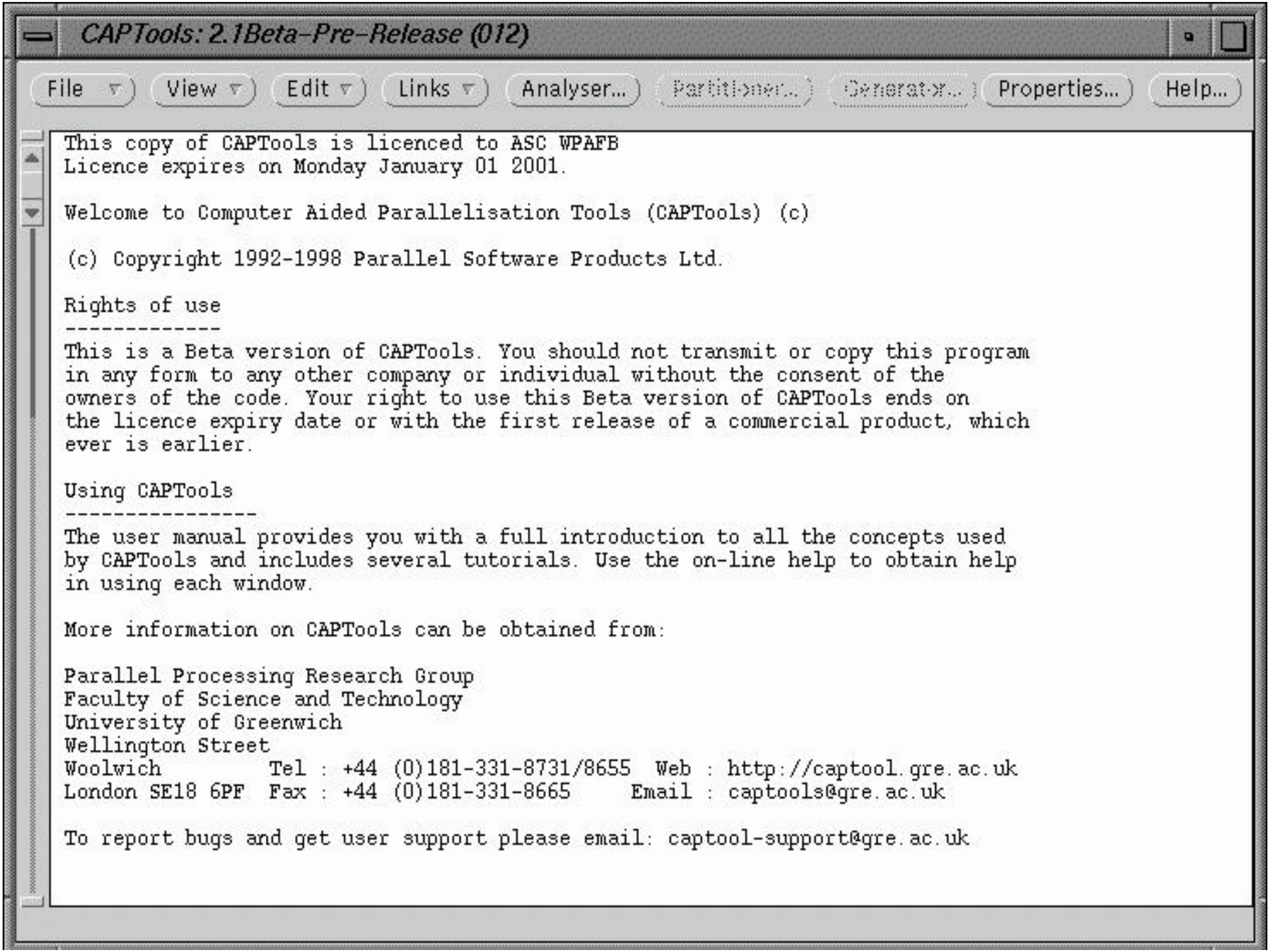
Steps (cont.)

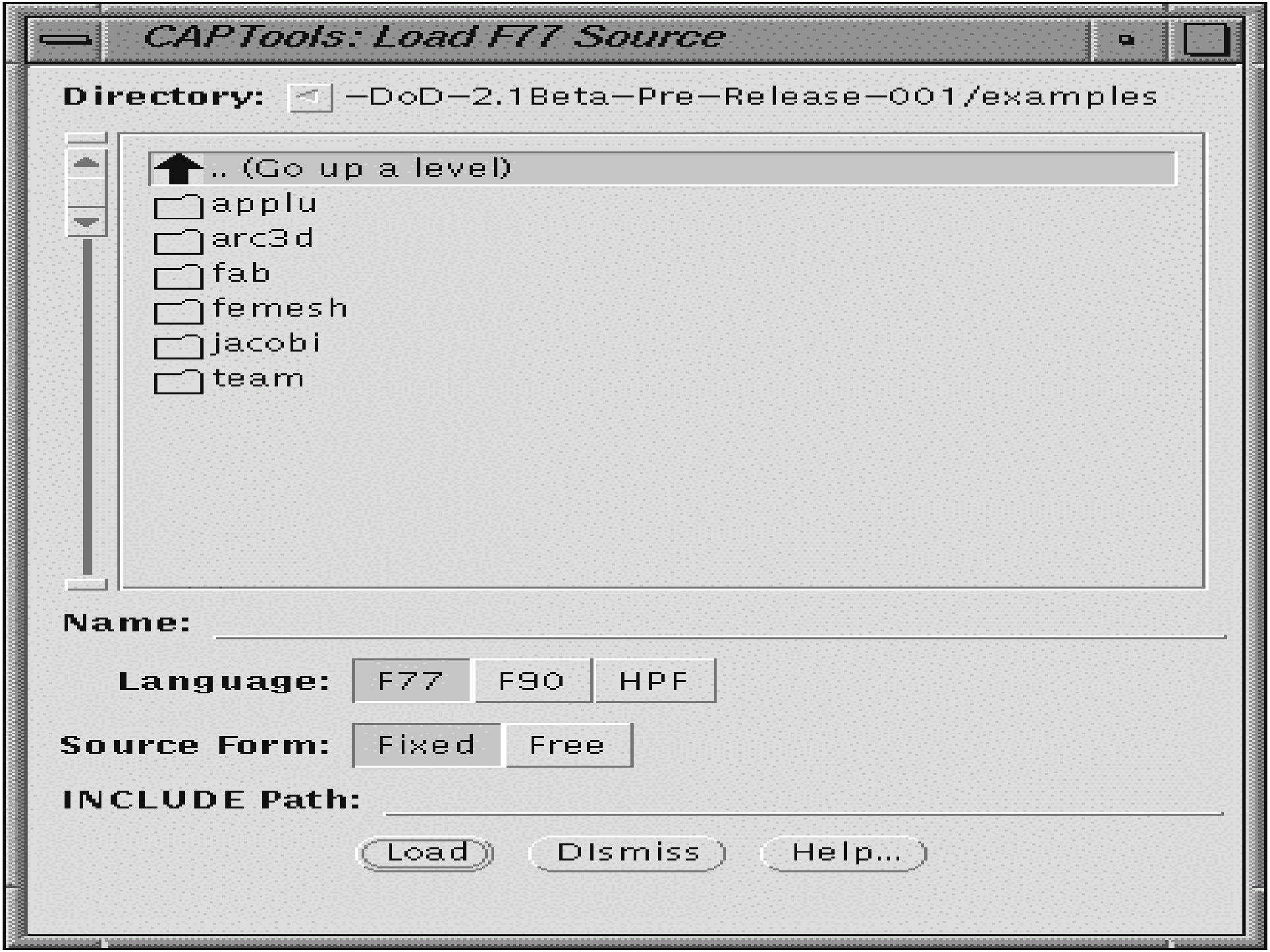
- Partition the data
- Generate control masks
- Generate communication calls

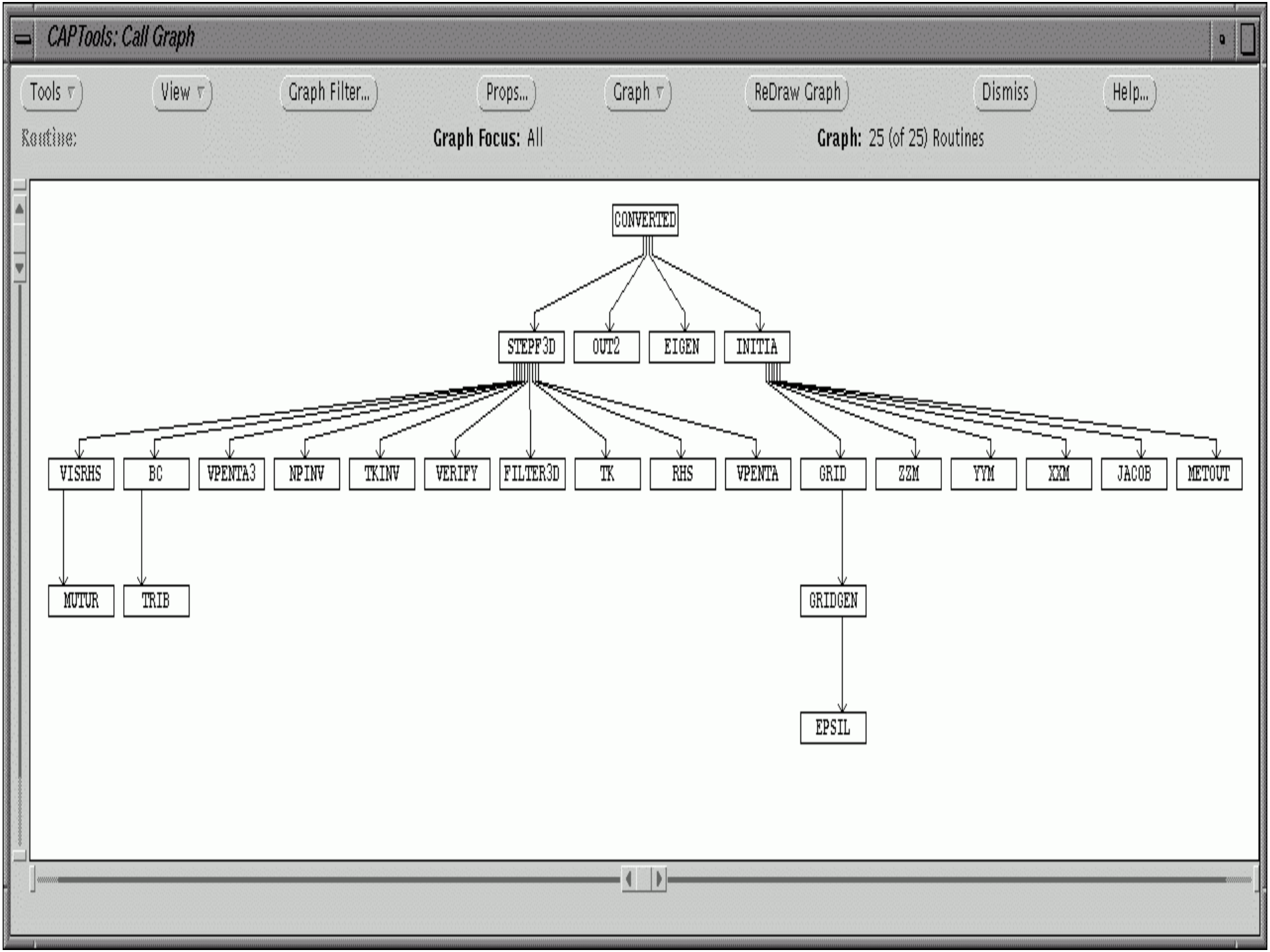
Steps (cont.)

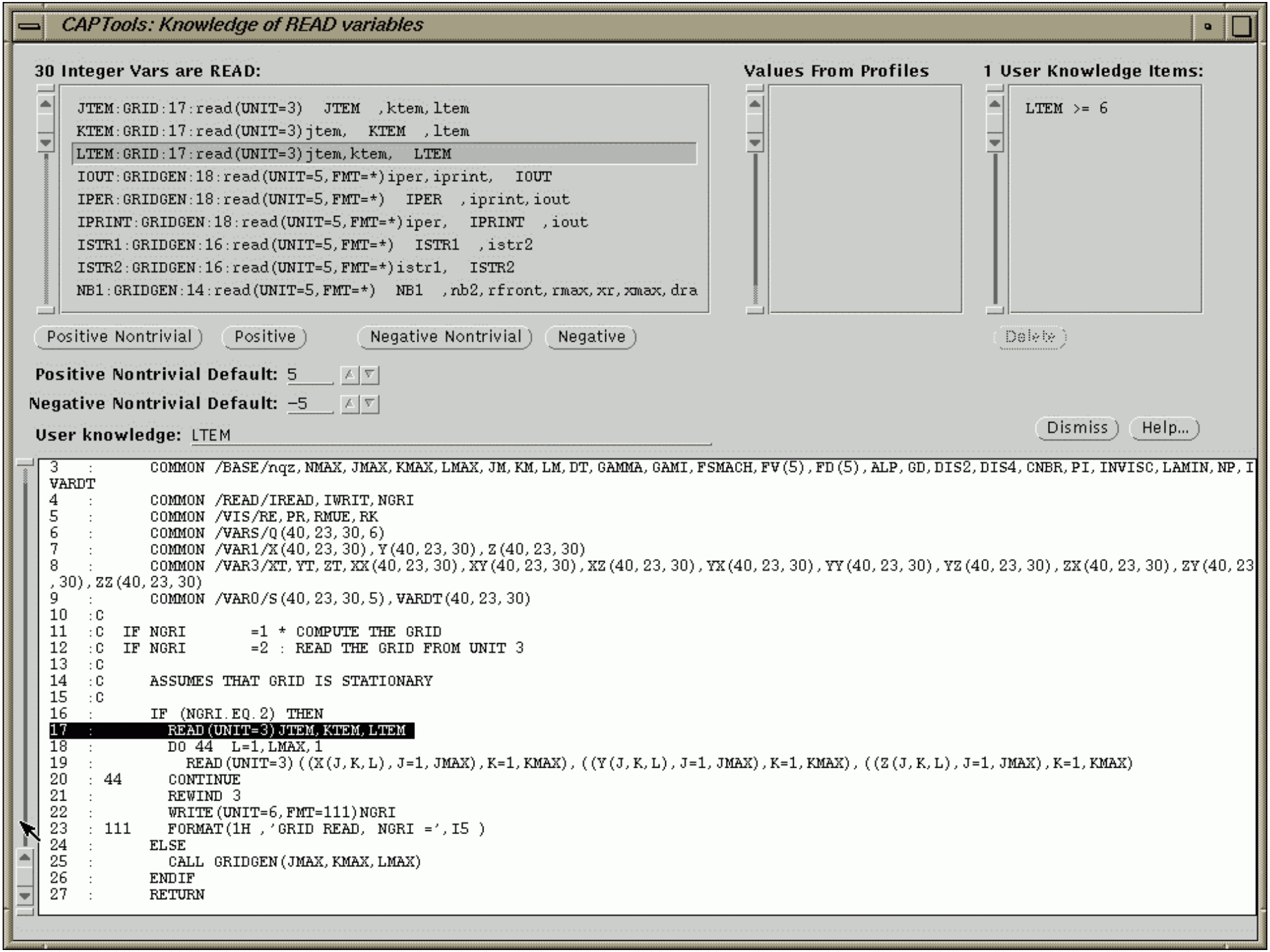
- Generate parallel source code
- Compile/link/run the code
- Validate the results

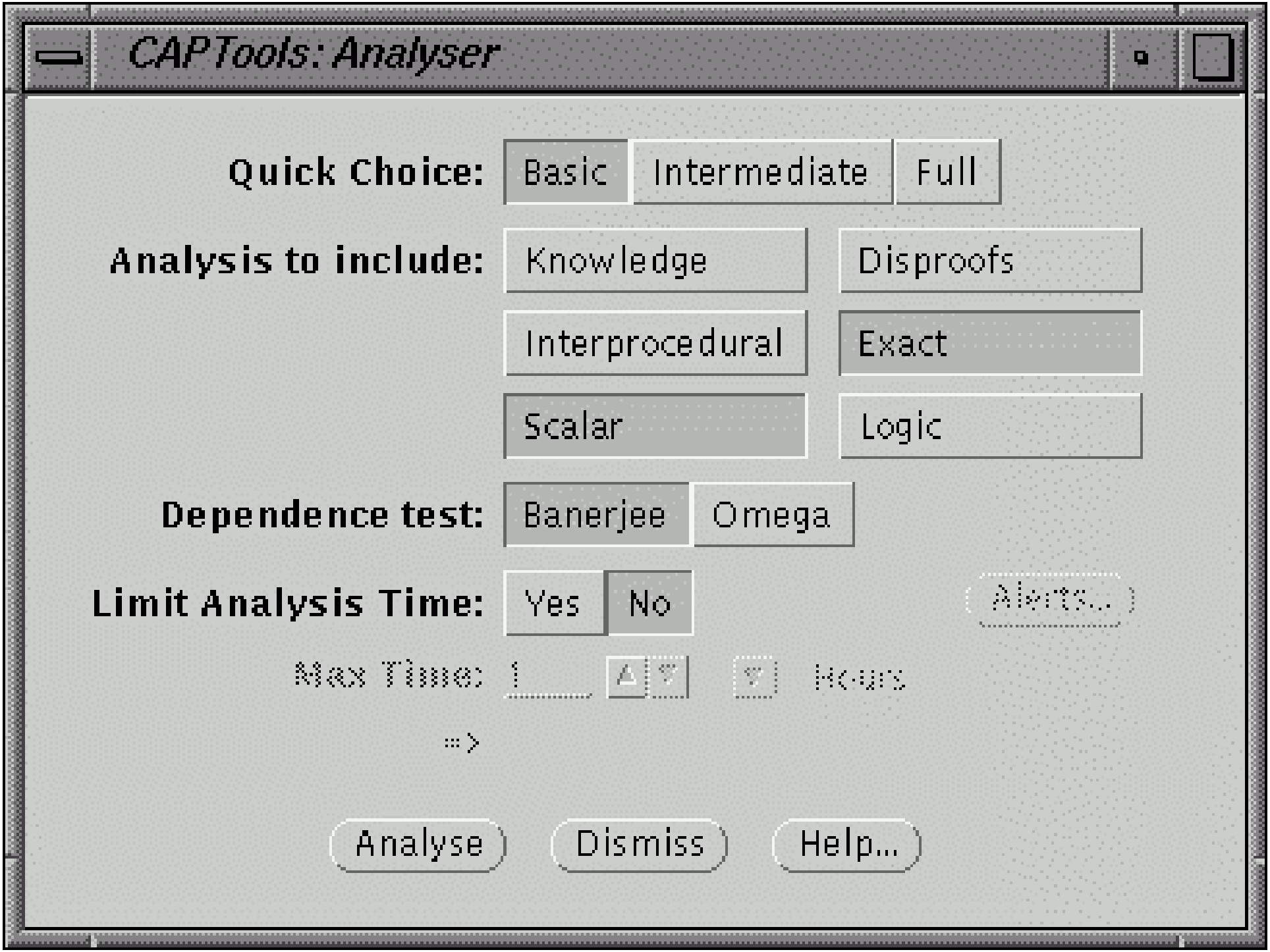
Session


CAPTools: 2.1Beta-Pre-Release (012)
File
View
Edit
Links
Analyser...
Partitioner...
Generator...
Properties...
Help...
This copy of CAPTools is licenced to ASC WPAFB
Licence expires on Monday January 01 2001.
Welcome to Computer Aided Parallelisation Tools (CAPTools) (c)
(c) Copyright 1992-1998 Parallel Software Products Ltd.
Rights of use
This is a Beta version of CAPTools. You should not transmit or copy this program in any form to any other company or individual without the consent of the owners of the code. Your right to use this Beta version of CAPTools ends on the licence expiry date or with the first release of a commercial product, which ever is earlier.
Using CAPTools
The user manual provides you with a full introduction to all the concepts used by CAPTools and includes several tutorials. Use the on-line help to obtain help in using each window.
More information on CAPTools can be obtained from:
Parallel Processing Research Group
Faculty of Science and Technology
University of Greenwich
Wellington Street
Woolwich Tel : +44 (0)181-331-8731/8655 Web : http://captool.gre.ac.uk
London SE18 6PF Fax : +44 (0)181-331-8665 Email : captools@gre.ac.uk
To report bugs and get user support please email: captool-support@gre.ac.uk


CAPTools: Load F77 Source
Directory:
-DoD-2.1Beta-Pre-Release-001/examples
.. (Go up a level)
applu
arc3d
fab
femesh
jacobi
team
Name:
Language:
F77
F90
HPF
Source Form:
Fixed
Free
INCLUDE Path:
Load
Dismiss
Help...


CAPTools: Call Graph
Tools
View
Graph Filter...
Props...
Graph
ReDraw Graph
Dismiss
Help...
Routine:
Graph Focus: All
Graph: 25 (of 25) Routines
CONVERTED
STEPF3D
OUT2
EIGEN
INITIA
VISRHS
BC
VPENTA3
NPINV
TKINV
VERIFY
FILTER3D
TK
RHS
VPENTA
GRID
ZZM
YYM
XXM
JACOB
METOUT
MUTUR
TRIB
GRIDGEN
EPSIL


CAPTools: Knowledge of READ variables
30 Integer Vars are READ:
JTEM:GRID:17:read(UNIT=3) JTEM ,ktem,ltem
KTEM:GRID:17:read(UNIT=3)jtem, KTEM ,ltem
LTEM:GRID:17:read(UNIT=3)jtem,ktem, LTEM
IOUT:GRIDGEN:18:read(UNIT=5,FMT=*)iper,iprint, IOUT
IPER:GRIDGEN:18:read(UNIT=5,FMT=*) IPER ,iprint,iout
IPRINT:GRIDGEN:18:read(UNIT=5,FMT=*)iper, IPRINT ,iout
ISTR1:GRIDGEN:16:read(UNIT=5,FMT=*) ISTR1 ,istr2
ISTR2:GRIDGEN:16:read(UNIT=5,FMT=*)istr1, ISTR2
NB1:GRIDGEN:14:read(UNIT=5,FMT=*) NB1 ,nb2,rfront,rmax,xr,xmax,dra
Values From Profiles
1 User Knowledge Items:
LTEM >= 6
Positive Nontrivial
Positive
Negative Nontrivial
Negative
Delete
Positive Nontrivial Default: 5
Negative Nontrivial Default: -5
User knowledge: LTEM
Dismiss
Help...
3 : COMMON /BASE/nqz,NMAX,JMAX,KMAX,LMAX,JM,KM,LM,DT,GAMMA,GAMI,FSMACH,FV(5),FD(5),ALP,GD,DIS2,DIS4,CNBR,PI,INVISC,LAMIN,NP,I
VARDT
4 : COMMON /READ/IREAD,IWRIT,NGRI
5 : COMMON /VIS/RE,PR,RMUE,RK
6 : COMMON /VARS/Q(40,23,30,6)
7 : COMMON /VAR1/X(40,23,30),Y(40,23,30),Z(40,23,30)
8 : COMMON /VAR3/XT,YT,ZT,XX(40,23,30),XY(40,23,30),XZ(40,23,30),YX(40,23,30),YY(40,23,30),YZ(40,23,30),ZX(40,23,30),ZY(40,23
,30),ZZ(40,23,30)
9 : COMMON /VAR0/S(40,23,30,5),VARDT(40,23,30)
10 :C
11 :C IF NGRI =1 * COMPUTE THE GRID
12 :C IF NGRI =2 : READ THE GRID FROM UNIT 3
13 :C
14 :C ASSUMES THAT GRID IS STATIONARY
15 :C
16 : IF (NGRI.EQ.2) THEN
17 : READ(UNIT=3)JTEM,KTEM,LTEM
18 : DO 44 L=1,LMAX,1
19 : READ(UNIT=3)((X(J,K,L),J=1,JMAX),K=1,KMAX),((Y(J,K,L),J=1,JMAX),K=1,KMAX),((Z(J,K,L),J=1,JMAX),K=1,KMAX)
20 : 44 CONTINUE
21 : REWIND 3
22 : WRITE(UNIT=6,FMT=111)NGRI
23 : 111 FORMAT(1H ,'GRID READ, NGRI =',I5)
24 : ELSE
25 : CALL GRIDGEN(JMAX,KMAX,LMAX)
26 : ENDIF
27 : RETURN


CAPTools: Analyser
Quick Choice:
Basic
Intermediate
Full
Analysis to include:
Knowledge
Disproofs
Interprocedural
Exact
Scalar
Logic
Dependence test:
Banerjee
Omega
Limit Analysis Time:
Yes
No
Alerts...
Max Time:
Hours
Analyse
Dismiss
Help...

- **Knowledge - combines knowledge entered by the user and that built up by the dependence analysis**
- **Disproofs - proves inequalities in dependence tests false**
- **Interprocedural - performs full interprocedural analysis**

- **Exact - performs tests to set the exact dependence attribute to dependencies**
- **Scalar - performs symbolic variable equality tests**
- **Logic - substitutes symbolic variables through the dependence graph**

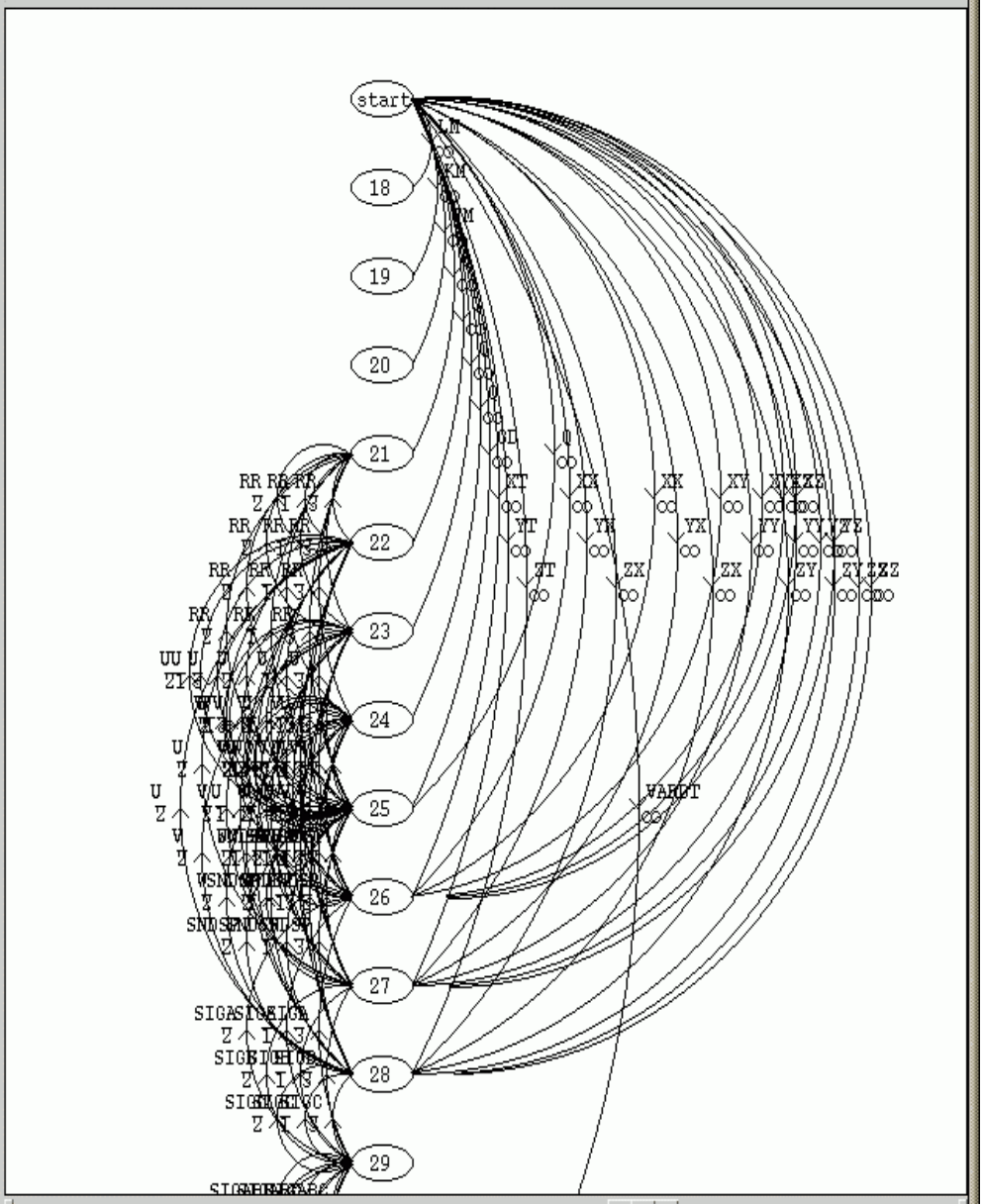
CAPTools: Dependence Graph

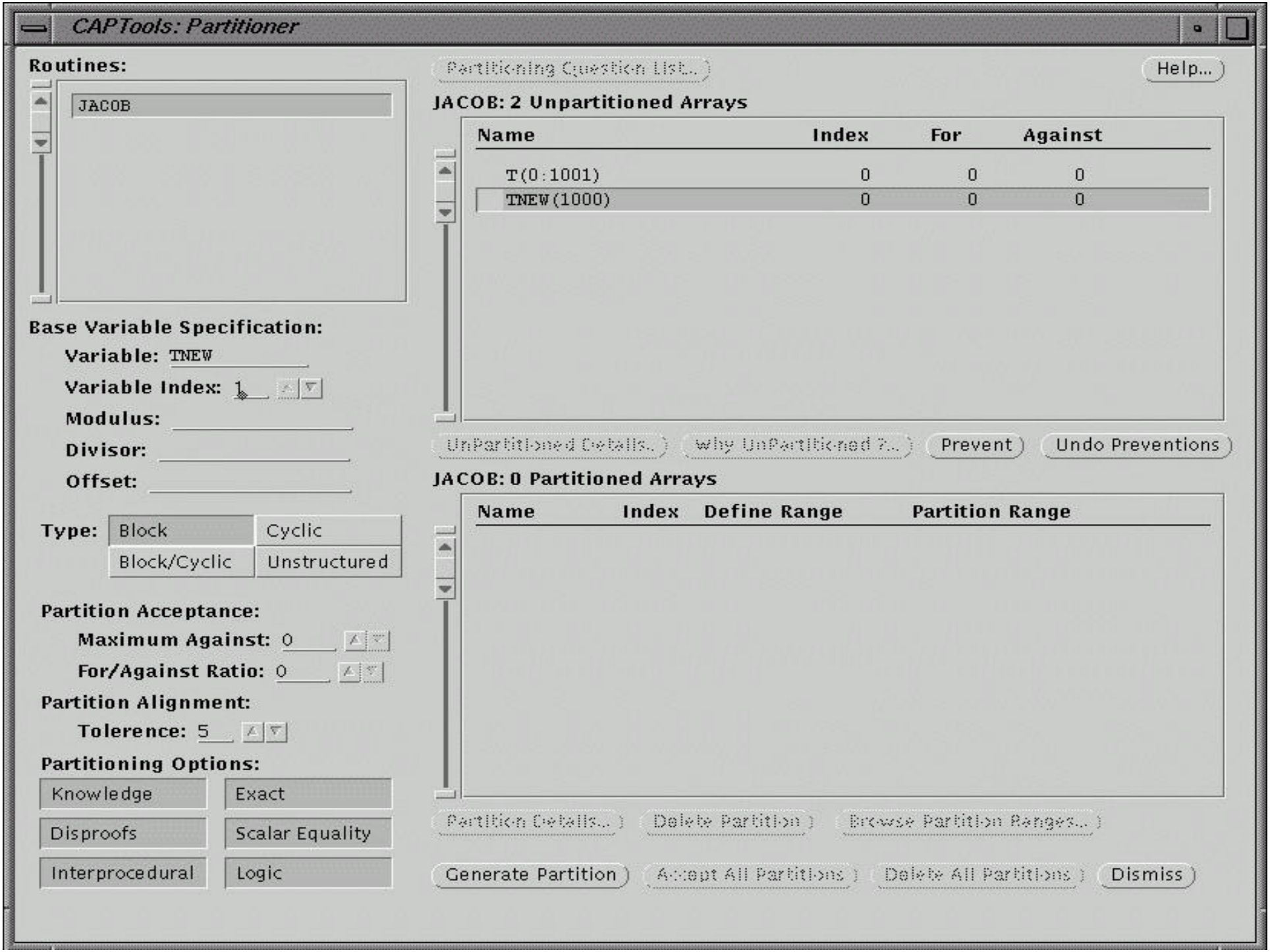
Statements... | Dependence Filter... | Normalise Text | Properties... | Tools | Graph | History | Why Dependence ? | ReDraw | Dismiss | Help...

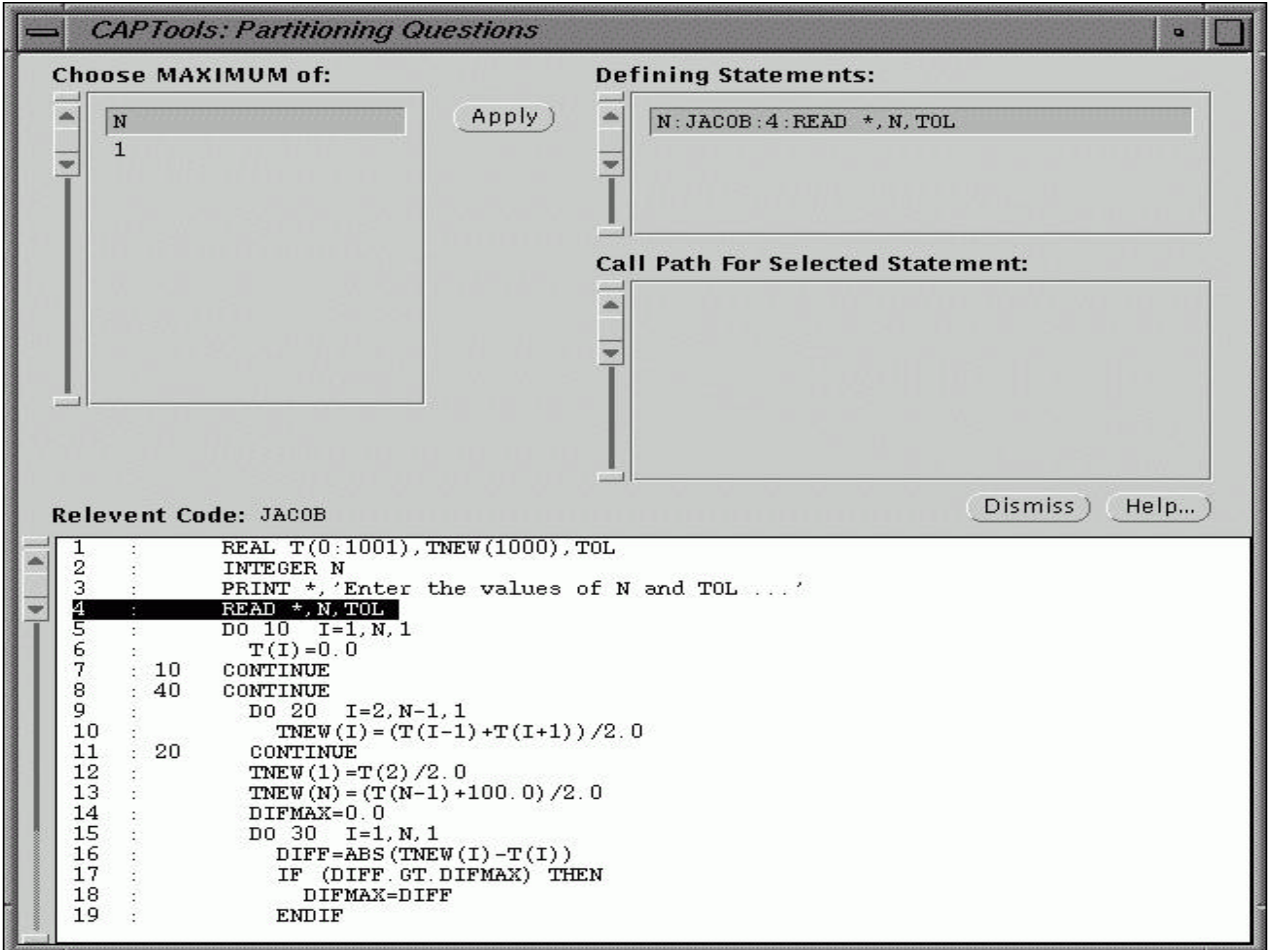
Current Routine: EIGEN

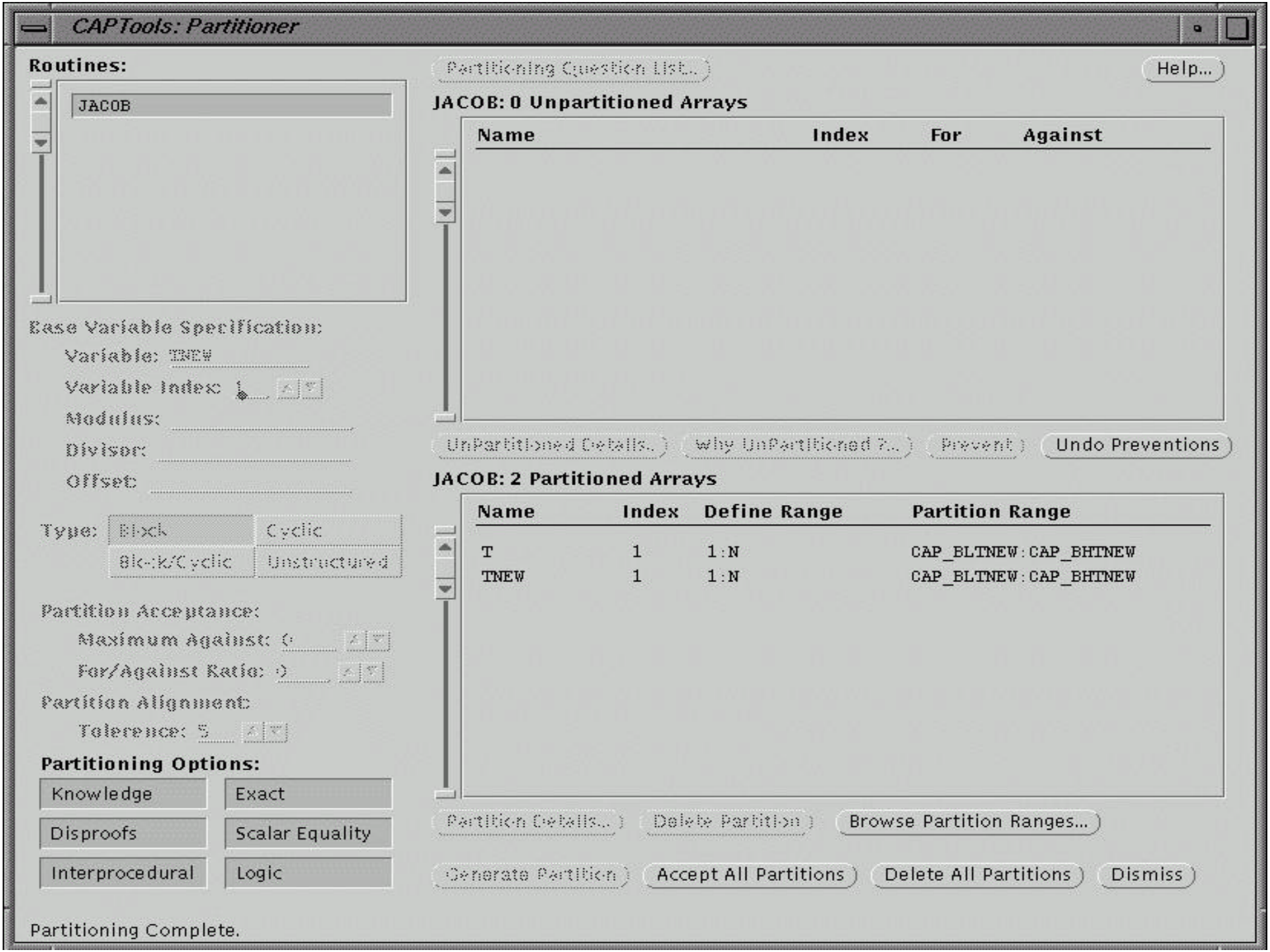
Current Graph: 18 statements, 127 dependencies displayed

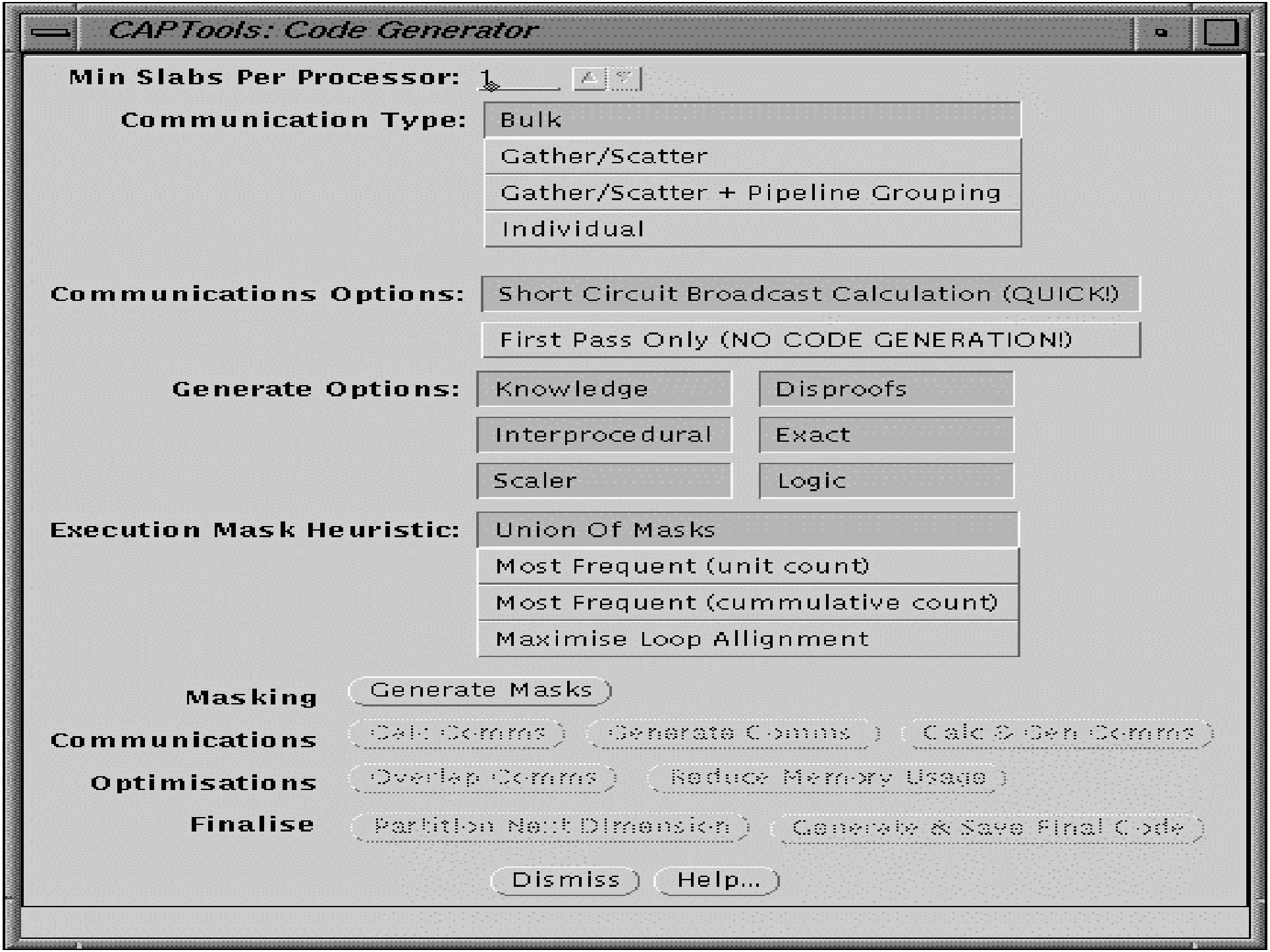
```
1    :       SUBROUTINE EIGEN
2    :       IMPLICIT REAL*8 (A-H,O-Z)
3    :       COMMON /BASE/nqz,NMAX,JMAX,KMAX,LMAX,JM,KM,LM,DT,GAMMA,GAMI,FSMACH
,FV(5),FD(5),ALP,GD,DIS2,DIS4,CNBR,PI,INVISC,LAMIN,NP,IVARDT
4    :       COMMON /READ/IREAD,IWRIT,NGRI
5    :       COMMON /VIS/RE,PR,RMUE,RK
6    :       COMMON /VARS/Q(40,23,30,6)
7    :       COMMON /VAR1/X(40,23,30),Y(40,23,30),Z(40,23,30)
8    :       COMMON /VAR3/XT,YT,ZT,XX(40,23,30),XY(40,23,30),XZ(40,23,30),YX(40
,23,30),YY(40,23,30),YZ(40,23,30),ZX(40,23,30),ZY(40,23,30),ZZ(40,23,30)
9    :       COMMON /VAR0/S(40,23,30,5),VARDT(40,23,30)
10   :C
11   :C  COMPUTE EIGENVALUES
12   :C
13   :C
14   :       JT=1
15   :       KT=1
16   :       LT=1
17   :       SIGMAX=0.D0
18   :       DO 10  L=2,LM,1
19   :         DO 10  K=2,KM,1
20   :           DO 10  J=2,JM,1
21   :             RR=1.D0/Q(J,K,L,1)
22   :             U=Q(J,K,L,2)*RR
23   :             V=Q(J,K,L,3)*RR
24   :             W=Q(J,K,L,4)*RR
25   :             SNDSP=SQRT(GD*(Q(J,K,L,5)*RR-.5D0*(U*U+V*V+W*W)))
26   :             SIGA=ABS(XT+U*XX(J,K,L)+V*XY(J,K,L)+W*XZ(J,K,L))+SNDSP*SQRT(
XX(J,K,L)**2+XY(J,K,L)**2+XZ(J,K,L)**2)
27   :             SIGB=ABS(YT+U*YX(J,K,L)+V*YY(J,K,L)+W*YZ(J,K,L))+SNDSP*SQRT(
YX(J,K,L)**2+YY(J,K,L)**2+YZ(J,K,L)**2)
28   :             SIGC=ABS(ZT+U*ZX(J,K,L)+V*ZY(J,K,L)+W*ZZ(J,K,L))+SNDSP*SQRT(
ZX(J,K,L)**2+ZY(J,K,L)**2+ZZ(J,K,L)**2)
29   :             SIGABC=MAX(SIGA,SIGB,SIGC)
30   :             IF (SIGABC-SIGMAX.LT.0) THEN
31   :               GOTO 10
32   :             ELSEIF (SIGABC-SIGMAX.GT.0) THEN
33   :               GOTO 38
34   :             ELSE
35   :               GOTO 10
36   :             ENDIF
37   : 38          JT=J
38   :             KT=K
39   :             LT=L
40   :             SIGMAX=SIGABC
41   : 10        CONTINUE
42   :         CONTINUE
43   :       CONTINUE
44   :       DXM=1.D0
45   :       CNBR=VARDT(JT,KT,LT)*DT*SIGMAX/DXM
46   :       WRITE(UNIT=6,FMT=600)JT,KT,LT,CNBR
47   : 600   FORMAT(1H ,30HJ,K,L,  AND MAX COURANT NBR     ,3I5,E14.5)
```




CAPTools: Partitioner
Routines:
JACOB
Partitioning Question List...
Help...
JACOB: 2 Unpartitioned Arrays
Name
Index
For
Against
T(0:1001)
0
0
0
TNEW(1000)
0
0
0
UnPartitioned Details...
Why UnPartitioned ?...
Prevent
Undo Preventions
Base Variable Specification:
Variable: TNEW
Variable Index: 1
Modulus:
Divisor:
Offset:
Type:
Block
Cyclic
Block/Cyclic
Unstructured
JACOB: 0 Partitioned Arrays
Name
Index
Define Range
Partition Range
Partition Acceptance:
Maximum Against: 0
For/Against Ratio: 0
Partition Alignment:
Tolerence: 5
Partitioning Options:
Knowledge
Exact
Disproofs
Scalar Equality
Interprocedural
Logic
Partition Details...
Delete Partition
Browse Partition Ranges...
Generate Partition
Accept All Partitions
Delete All Partitions
Dismiss


CAPTools: Partitioning Questions
Choose MAXIMUM of:
N
1
Apply
Defining Statements:
N:JACOB:4:READ *,N,TOL
Call Path For Selected Statement:
Dismiss
Help...
Relevent Code: JACOB
1 : REAL T(0:1001),TNEW(1000),TOL
2 : INTEGER N
3 : PRINT *,'Enter the values of N and TOL ...'
4 : READ *,N,TOL
5 : DO 10 I=1,N,1
6 : T(I)=0.0
7 : 10 CONTINUE
8 : 40 CONTINUE
9 : DO 20 I=2,N-1,1
10 : TNEW(I)=(T(I-1)+T(I+1))/2.0
11 : 20 CONTINUE
12 : TNEW(1)=T(2)/2.0
13 : TNEW(N)=(T(N-1)+100.0)/2.0
14 : DIFMAX=0.0
15 : DO 30 I=1,N,1
16 : DIFF=ABS(TNEW(I)-T(I))
17 : IF (DIFF.GT.DIFMAX) THEN
18 : DIFMAX=DIFF
19 : ENDIF


CAPTools: Partitioner
Routines:
JACOB
Partitioning Question List...
Help...
JACOB: 0 Unpartitioned Arrays
Name
Index
For
Against
UnPartitioned Details...
Why UnPartitioned ?...
Prevent
Undo Preventions
Base Variable Specification:
Variable: TNEW
Variable Index: 1
Modulus:
Divisor:
Offset:
Type: Block
Cyclic
Block/Cyclic
Unstructured
JACOB: 2 Partitioned Arrays
Name
Index
Define Range
Partition Range
T 1 1:N CAP_BLTNEW:CAP_BHTNEW
TNEW 1 1:N CAP_BLTNEW:CAP_BHTNEW
Partition Acceptance:
Maximum Against: 0
For/Against Ratio: 0
Partition Alignment:
Tolerence: 5
Partition Details...
Delete Partition
Browse Partition Ranges...
Partitioning Options:
Knowledge
Exact
Disproofs
Scalar Equality
Interprocedural
Logic
Generate Partition
Accept All Partitions
Delete All Partitions
Dismiss
Partitioning Complete.


CAPTools: Code Generator
Min Slabs Per Processor: 1
Communication Type:
Bulk
Gather/Scatter
Gather/Scatter + Pipeline Grouping
Individual
Communications Options:
Short Circuit Broadcast Calculation (QUICK!)
First Pass Only (NO CODE GENERATION!)
Generate Options:
Knowledge
Disproofs
Interprocedural
Exact
Scaler
Logic
Execution Mask Heuristic:
Union Of Masks
Most Frequent (unit count)
Most Frequent (cummulative count)
Maximise Loop Allignment
Masking
Generate Masks
Communications
Calc Comms
Generate Comms
Calc & Gen Comms
Optimisations
Overlap Comms
Reduce Memory Usage
Finalise
Partition Next Dimension
Compile & Save Final Code
Dismiss
Help...

Start CAPTools

- **Login to SGI Origin 2000 at ASC (hpc03.asc.hpc.mil)**
- **Type in:**

  `capo` **(OpenMP)**

  `captool` **(MPI)**

Compile/link

Code with OpenMP on O2K:

- `f77 -mp file.f`
- `f90 -mp file.f`

Compile/link (cont.)

Code with MPI on O2K:

- `capf77 -m sgi_64 -p mpi file.f`

Code with MPI on SP2:

- `capf77 -m sp2 -p mpi file.f`

Use “`capf77 -help`” for more info

Run

OpenMP, interactive, SGI O2K:

```
setenv MP_SET_NUMTHREADS 2
a.out
```

Run (cont.)

MPI, interactive, O2K:

```
caprun -m sgi_64 -p mpi \
       -top pipe2 a.out
```

MPI, interactive, SP2:

```
caprun -m sp2 -p mpi \
       -top pipe2 a.out
```

Use “`caprun -help`” for details

Examples

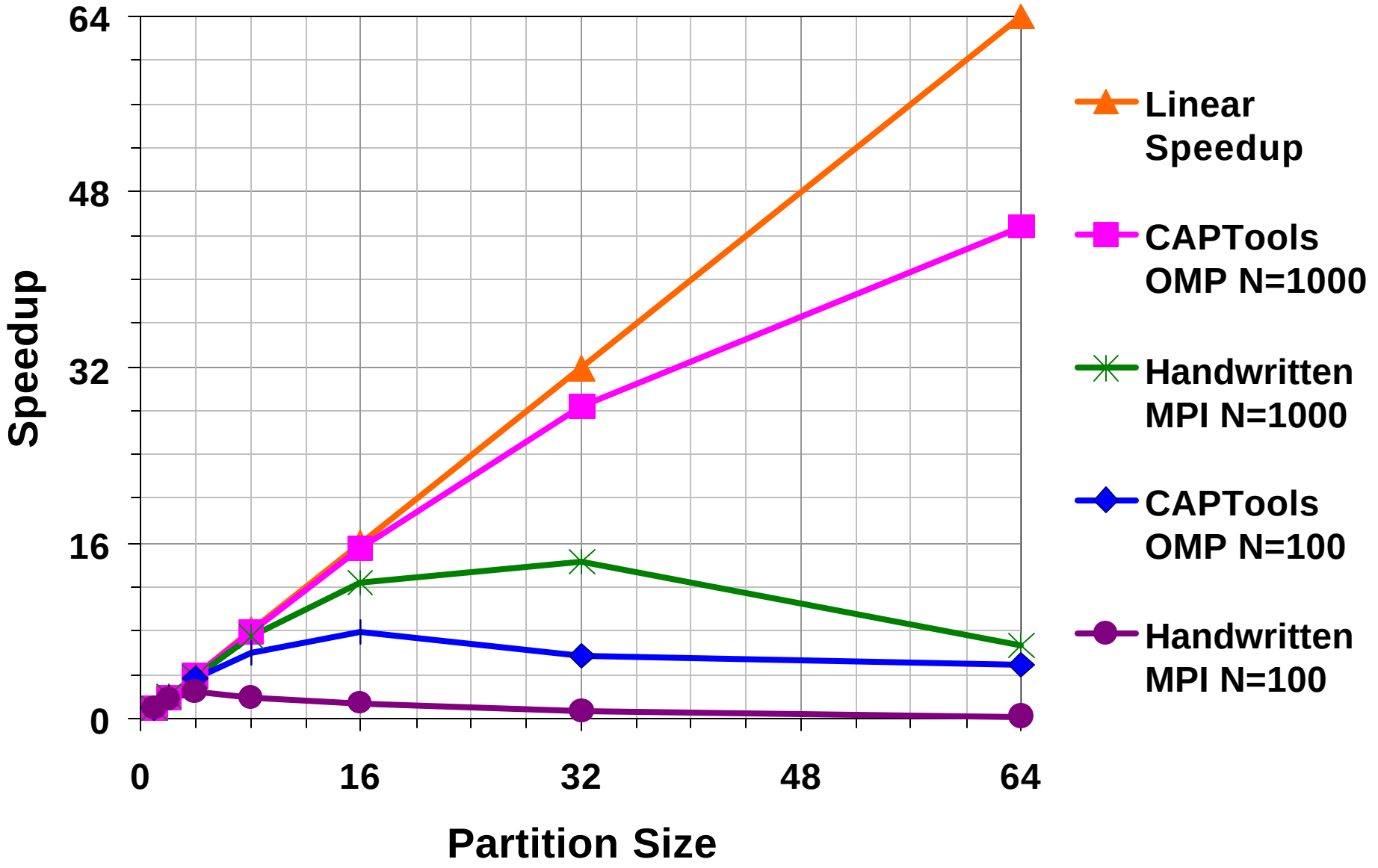
- **Applications**
- **Accelerating code parallelization with CAPTools**
- **Benefit to DoD**
- **More info**

Applications

- **N-Body**
- **PFEM**
- **R-Jet**

N-Body

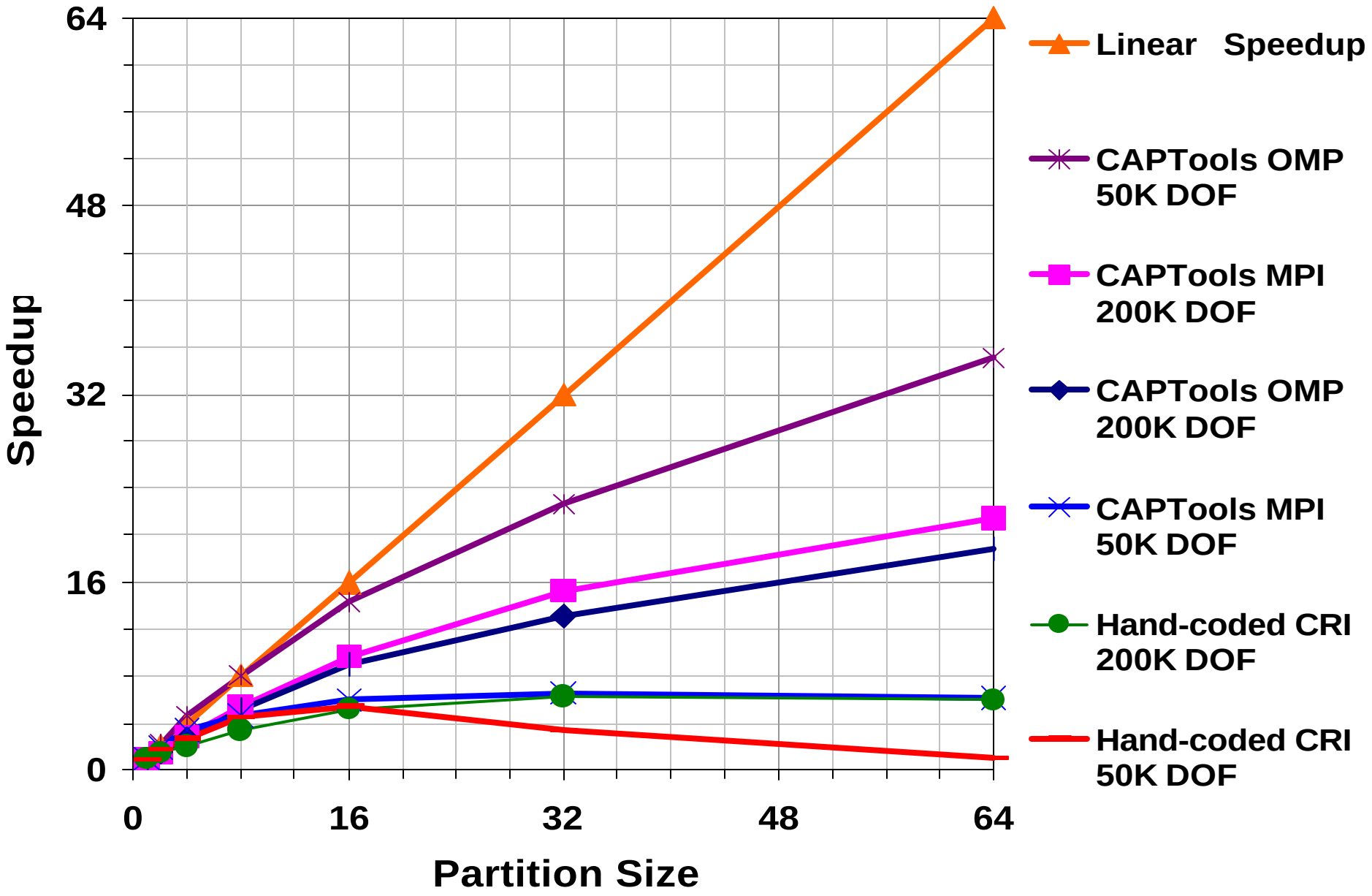
- Distributes points evenly on the unit sphere
- 3D n-body model
- Hand-written MPI
- CAPTools generated OpenMP



Speedup Curves for the N-Body Code

PFEM

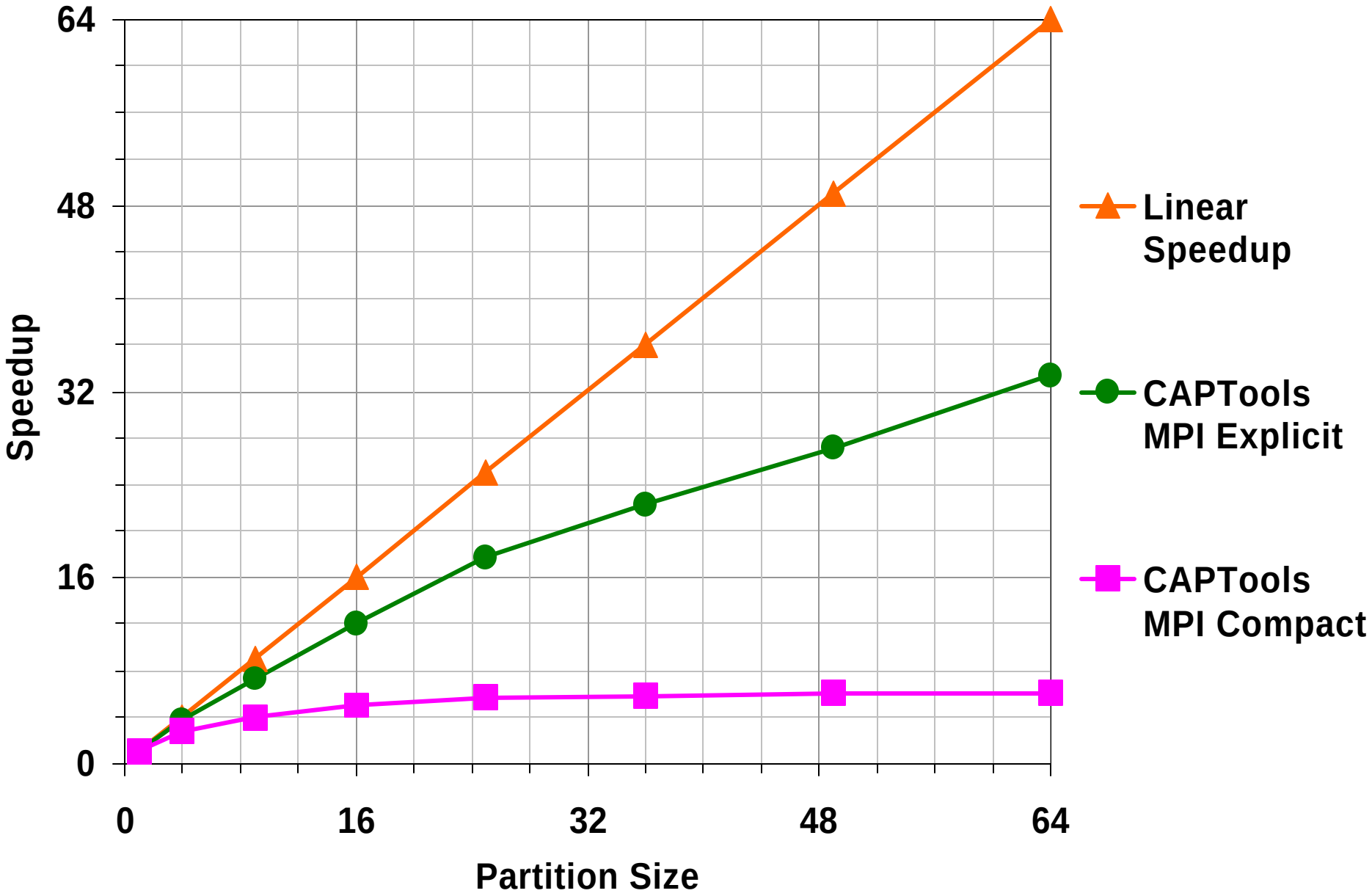
- **Solves 2D highly nonlinear BVP in CSM**
- **Domain decomposition (two-colorings)**
- **Hand-coded directive-driven**



Speedup Curves for the PFEM Code

R-Jet

- **Simulates vortex dynamics and breakdown in turbulent jets**
- **Hybrid, high-order, compact finite difference spectral method**



Speedup Curves for the R-Jet Code

Accelerating code parallelization with CAPTools

	Code analysis	Parallel strategy	Parallel code generation	Numerical validation debugging	Optimiz ation
Hand	1 mth	1 mth	3 mths	6 mths	3 mths
CAP Tools	2 days	2 days	< 1 hour each time	1 week	1 week

Parallelization process reduced from ~1 year to less than 1 month

Benefit to DoD

- **Effective utilization of SMPs and DMPs now and in any combination in future**
- **Minimize the time required to parallelize codes**
- **Enable a range of parallelization strategies**
- **Produce scalable and efficient codes**
- **Provide more reliable simulation of complete warfighting systems**

http://www.asc.hpc.mil/PET/PTES/CAPTools/index.html

http://captools.gre.ac.uk

luczakr@asc.hpc.mil