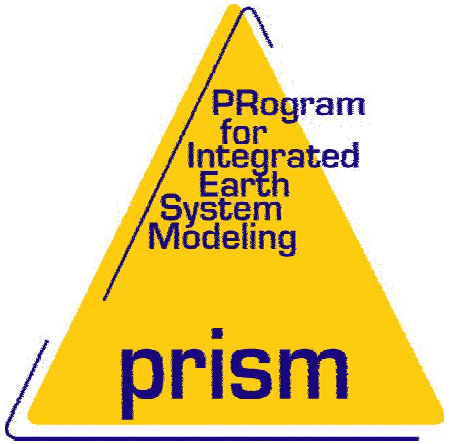
The OASIS4 coupler



The current status

- Presented by S. Valcke

Outline

- Historical background
- Some key notes
- Driver
- Communication
- Regridding/transformations
- Grids supported
- Conclusion
- Future developments

Historical background

OASIS: developed since 1991 to couple existing GCMs

1991 **2001**

|--→ |--- PRISM →

OASIS 1 → OASIS 2 → OASIS3 (demo runs) →

→ OASIS4 →

OASIS1, OASIS2, OASIS3:

- low resolution, low nbr of 2D fields, low coupling frequency:
  - ➔flexibility very important, efficiency not so much!
  - ➔~15 user groups in Europe, Australia, USA, Canada, Japan, etc.

OASIS4:

- higher resolution parallel models, massively parallel platforms, 3D fields
  - ➔need to optimise and parallelise the coupler
  - ➔Present users: SMHI (Sweden), IFM-GEOMAR (Kiel), and GEMS

Some key notes

- Developers: CERFACS; NEC CCRLE; SGI; NEC HPCE
- Public domain; open source license (LGPL)
- Programming language: Fortran 90 and C
- Public domain libraries; vendor optimized versions may exist:
  - MPI1 and/or MPI2; NetCDF/parallel NetCDF; libXML
  - mpp_io; SCRIP

OASIS4 Driver

- Manages a static coupling:

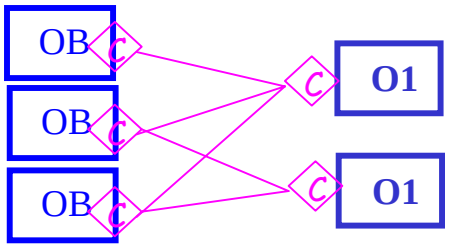
  Driver launches models at the beginning of the run (MPI2) or models are started directly in the run script (MPI1)

- Participates to communicator set-up

- Reads the XML user-defined configuration information and distributes it to the component PSMILes

OASIS4 communication (1/2)

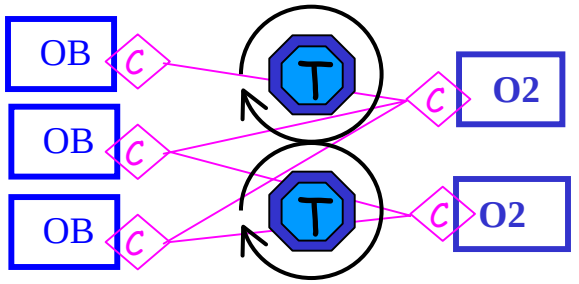
- Model interface library: PSMILe based on MPI1 or MPI2
- Parallel communication **including repartitioning**:
  - based on geographical description of the partitions
  - parallel calculation of communication patterns in source PSMILe



Same grid, different decomposition

⇒ direct repartitioning:

• for each target point, parallel calculation of source matching point in source PSMILe



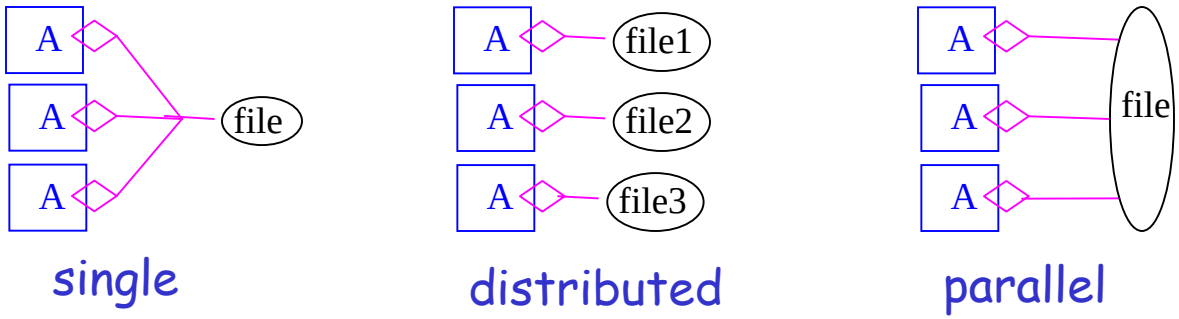
Different grid and decomposition

⇒ via parallel Transformer:

• for each target point, parallel calculation of source interpolation neighbours in source PSMILe

OASIS4: communication (2/2)

- end-point communication
- parallel 3D neighbourhood search in each source process PSMILe
- parallel multigrid search algorithm: high efficiency
- extraction of useful part of source field only
- one-to-one, one-to-many (source data can be consumed by different target models or files at different frequencies)
- parallel I/O (vector, bundles, vector bundles) : GFDL mpp_io, parNetCDF



OASIS4 regridding/transformations

- source time transformations (prism_put):
  - average, accumulation
- target time transformations (prism_get)
  - time interpolation (for I/O only)
- statistics
- local transformations:
  - addition/multiplication by scalar
- interpolation/regridding (3D):
  - nearest-neighbour 2D in the horizontal, "none" in the vertical
  - nearest-neighbour 3D
  - bilinear in the horizontal, "none" in the vertical
  - bicubic (gradient, 16 nghbrs) in the horizontal, "none" in the vertical
  - trilinear

Grids supported by OASIS4

- Regridding, repartitioning, I/O:
  - Regular in lon, lat, vert ("Reglonlatvrt"):
    - lon(i), lat(j), height(k)
  - Irregular in lon and lat, regular in the vert ("irrlonlat_regvrt"):
    - lon(i,j), lat(i,j), height(k)
  - Irregular in lon, lat, and vert ("irrlonlatvrt") *(not fully tested)*
    - lon(i,j,k), lat(i,j,k), height(i,j,k)
  - Gaussian Reduced in lon and lat, regular in the vert ("Gaussreduced_regvrt")
    - lon(nbr_pt_hor), lat(nbr_pt_hor), height(k)
- Repartitioning and I/O only:
  - "Gridless" fields
    - no geographical information attached
    - local partitions described in the global index space (prism_def_partition)
- I/O only:
  - Unstructured grids ("unstructlonlatvrt")
    - lon(npt_tot), lat(npt_tot), height(npt_tot)

OASIS4: conclusions

- OASIS4 tested and run with toy examples on:
  - Intel Pentium 4 Workstation Cluster
  - SGI O3000/2000 server with MIPS 4 processors and IRIX64
  - SGI IA64 Linux server Altix 3000
  - NEC SX6
  - AMD 2800 Cluster
  - IBM Power 4
- OASIS4 now being used in a reduced number of real applications:
  - MOM4 + toy atmosphere
  - At SMHI, for regional coupling
  - At IFM-GEOMAR (Kiel) in pseudo-models to interpolate high-resolution fields.
  - … and in GEMS !
- OASIS4 full public release planned within a year.

OASIS4: future developments

- Short term (~March/Sept 2006)
  - Source management: Subversion and Trac
  - Regridding/transformations:
    - Full validation of schemes currently implemented (incl. scattering/gathering)
    - 2D1D with linear and nearest-neighbour in the vertical
    - 2D conservative remapping
    - Tricubic interpolation
    - Improve Transformer efficiency
  - Communication:
    - Implementation of the global search (not just local process search)
- Medium term (~March 2007)
  - PSMILe API for model access to more SMIOC info
  - PSMILe API for model access to calendar info
  - Support calendars other than proleptic Gregorian
  - Full support of vectors, bundles, subgrids
  - Non-blocking sending and receiving routines
  - Add more coherence checks in the Driver

OASIS4: future developments

- Long term:
  - Support types of exchange dates other than fixed frequency
  - 3D conservative remapping
  - User-defined 3D and 2D remapping
  - Field reduction, combination
  - Full support of unstructured grid
  - Support of adaptive grids

The end

OASIS4: component model description

Application and component description (XML files):

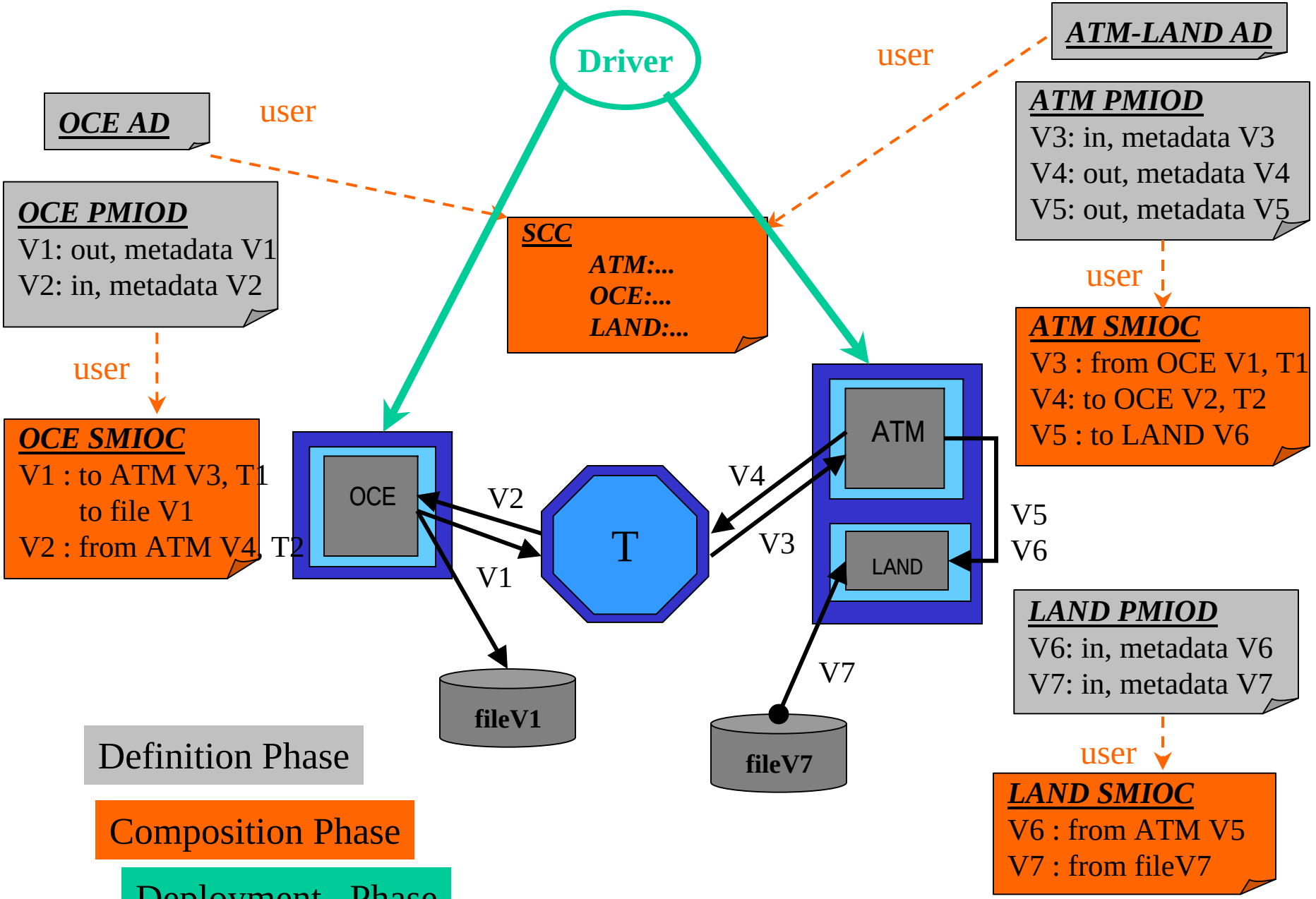
- **For each application** (code):
  - **one Application Description (AD):**
    - possible number of processes, components included, etc.
- **For each component** in the application:
  - **one Potential Model Input and Output Description (PMIOD)**
    - component general characteristics: name, component simulated, …
    - grid information: domain, resolution(s), grid type, …
    - **potential I/O or coupling variables**:
      - local name, standard name (NetCDF CF convention)
      - units, valid min and max
      - numerical type
      - associated grid and points
      - intent –input and/or output

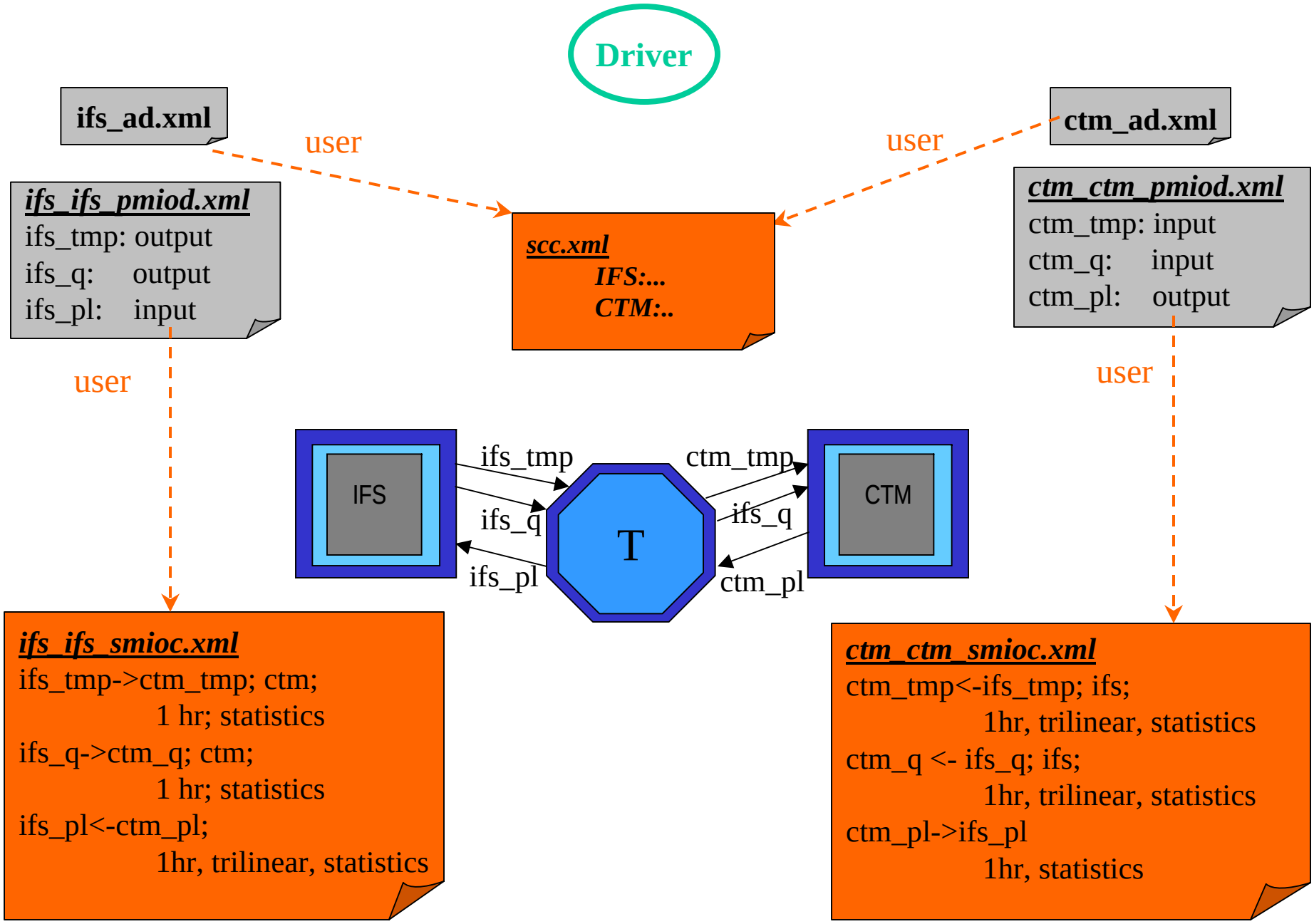
OASIS4: coupled model configuration

Coupled model configuration (XML files):

(Through a GUI,) the user produces

- a **Specific Coupling Configuration (SCC):**
  - start date and end date
  - start mode (MPI1, MPI2)
  - applications, components for each application
  - host(s), number of processes per host, ranks for each component
- **For each component,**
  a Specific Model Input and Output Configuration (SMIOC)
  - **grid** information: **chosen resolution**, …
  - **I/O or coupling variables**:
    - local and standard name, units, valid min max, numerical type, grid
    - **activated intent –input and/or output**
    - **source and/or target** (component and/or file)
    - **coupling or I/O dates**
    - **transformations/interpolations**


Driver
ATM-LAND AD
user
OCE AD
user
ATM PMIOD
V3: in, metadata V3
V4: out, metadata V4
V5: out, metadata V5
OCE PMIOD
V1: out, metadata V1
V2: in, metadata V2
SCC
ATM:...
OCE:...
LAND:...
user
ATM SMIOC
V3 : from OCE V1, T1
V4: to OCE V2, T2
V5 : to LAND V6
user
OCE SMIOC
V1 : to ATM V3, T1
to file V1
V2 : from ATM V4, T2
OCE
V2
V1
T
V4
V3
ATM
LAND
V5
V6
fileV1
V7
fileV7
LAND PMIOD
V6: in, metadata V6
V7: in, metadata V7
user
LAND SMIOC
V6 : from ATM V5
V7 : from fileV7
Definition Phase
Composition Phase
Deployment Phase

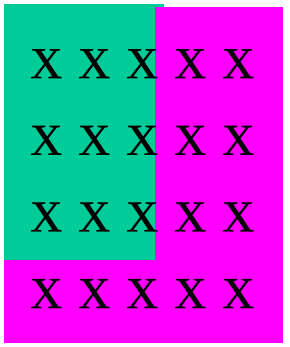

Driver
ifs_ad.xml
user
ctm_ad.xml
user
ifs_ifs_pmiod.xml
ifs_tmp: output
ifs_q: output
ifs_pl: input
scc.xml
IFS:...
CTM:..
ctm_ctm_pmiod.xml
ctm_tmp: input
ctm_q: input
ctm_pl: output
user
user
IFS
ifs_tmp
ifs_q
ifs_pl
T
ctm_tmp
ifs_q
ctm_pl
CTM
ifs_ifs_smioc.xml
ifs_tmp->ctm_tmp; ctm;
1 hr; statistics
ifs_q->ctm_q; ctm;
1 hr; statistics
ifs_pl<-ctm_pl;
1hr, trilinear, statistics
ctm_ctm_smioc.xml
ctm_tmp<-ifs_tmp; ifs;
1hr, trilinear, statistics
ctm_q <- ifs_q; ifs;
1hr, trilinear, statistics
ctm_pl->ifs_pl
1hr, statistics

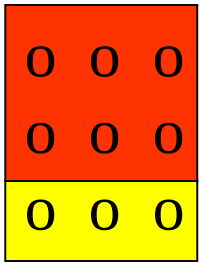
The end

Oasis4: communication (2/3)

Parallel calculation of communication patterns in source PSMILe

For each pair of source and target processes:





1/ Envelop exchange

2/ Detailed neighbourhood search

3/ EPIOS and EPIOT definition