

Kickoff COSTA project PSE for Data-Assimilation

Nils van Velzen

velzen@dutita2.twi.tudelft.nl

Delft 26 nov 2004

Overview

- What is COSTA and its benefits
- Components
 - Example: observations
- Interfaces
 - Example: state vector
- Parallel computing
- Users' group

What is COSTA

- Common Set of Tools for Assimilation of data
- Problem solving environment:
 - Definition of components and interfaces
 - Tools for rapidly building components
 - Data assimilation methods
 - Development philosophy
- Free software / documentation

Purpose

- ◌ Save on time and costs for the application of data assimilation
- ◌ Reusability of methods and models across institutions
- ◌ Fewer bugs in code
- ◌ Commercial models and assimilation methods

Components and Interfaces

- A component is a nontrivial, nearly independent and replaceable part of a system that fulfills a clear function in the context of a well-defined architecture. A component conforms to and provides the physical realisation of a set of interfaces

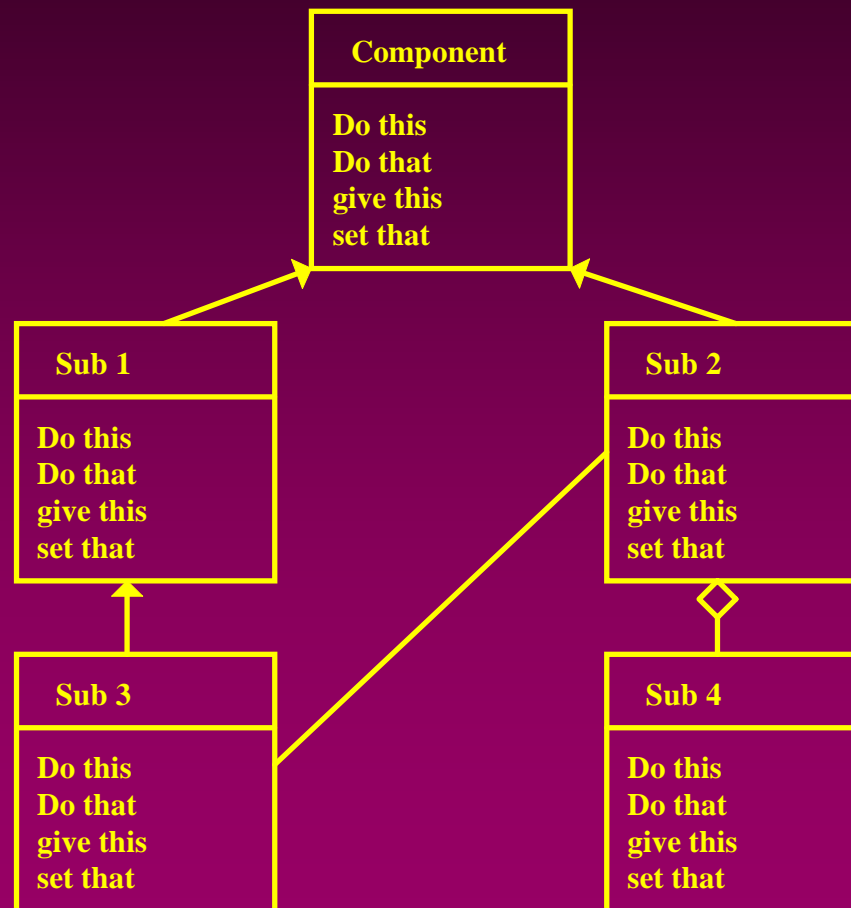
Components I

- Can data assimilation components be reused?
- Ideal situation reusable components e.g.
 - Model
 - Uncertainty model
 - Observations
 - Assimilation method
 - State vector

Components II

- ◌ Reuse the interfaces of components
- ◌ Reusability and replaceability of subcomponents
 - Data structures
 - Interpolation-methods
 - (Special) Linear algebra
 - Communication library
- ◌ Reusability of design patterns

Replace and Reuse



Components: Observations I

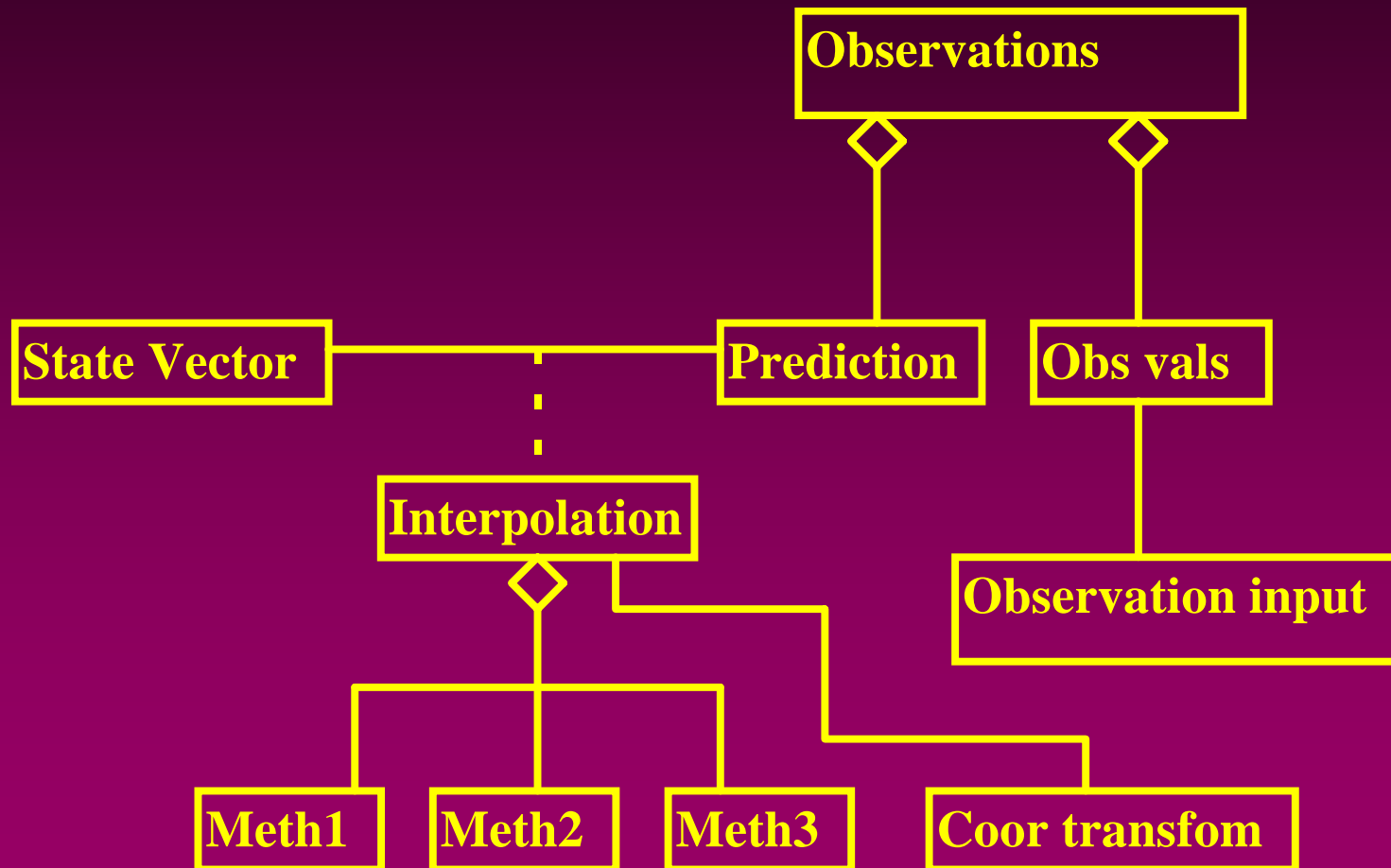
- Observation component handles:
 - The raw observations in different formats and from different sources
 - File
 - Memory
 - etc
 - Link between model and observations

Components: Observations II

◌ (output) Interface:

- Are there observations at time $t=?$
- Number of observations at time $t=?$
- Give list of station names
- Interpolated this state for given location
- Which subset of the state vector is used in interpolation to given station location
- Give interpolation matrix (H)

Components: Observations III



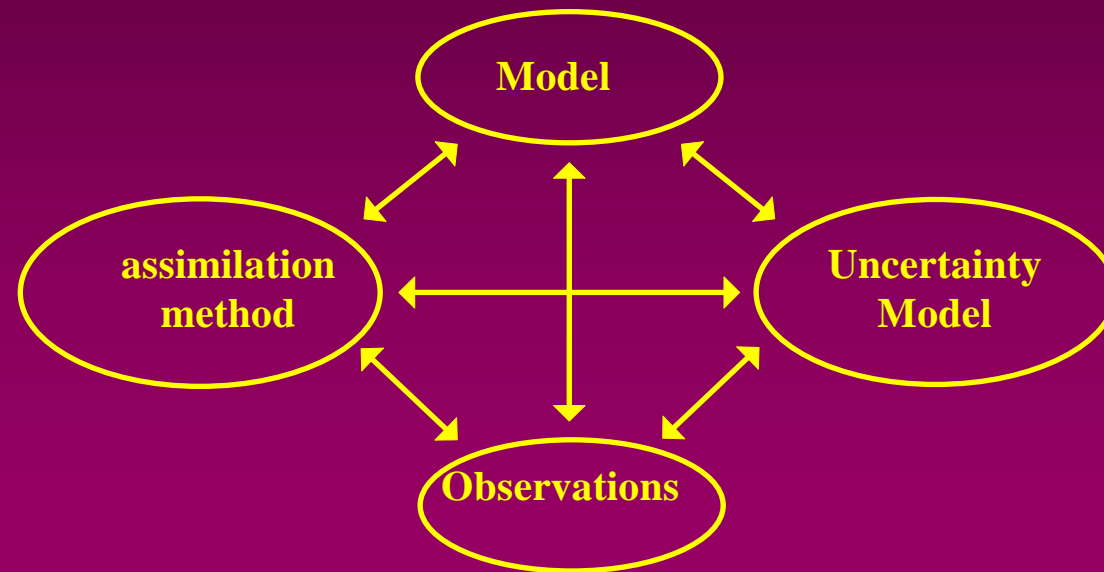
Components: Observations IV

Challenges:

- Defining methods/interfaces such that COSTA creates the interpolations
- How to handle nonlinear interpolations
- ?

Interfaces I

- What information and requests are “sent” between the components?



Interfaces II

- ◌ What do we want to ask a component?
 - Model
 - Give me your state-vector
 - Perform a timestep using this state
 - Perform adjoint timestep
 - ?
 - Observations
 - Give me the interpolated value at given observation location using this state
 - etc

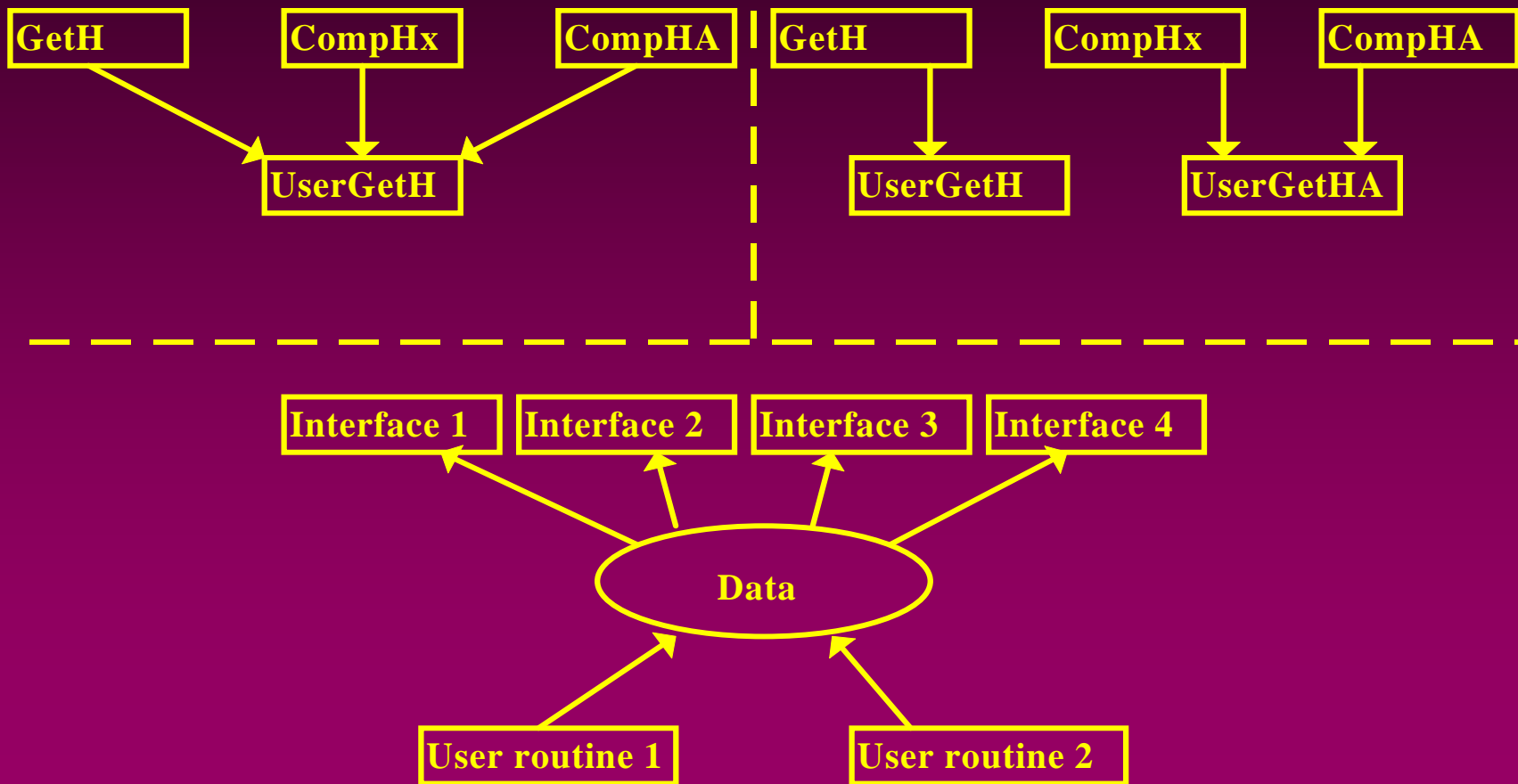
Interfaces III

- Uncertainty model
 - Multiply covariance matrix of ... with given vector
 - Give me the covariance matrix of ...
 - Give me L of the LL^T decomposition
 - ?
- Assimilation method
 - Assimilate data for given state and obs
 - ?

Interfaces: the other side

- Coding the interface completely
 - computationally efficient
 - expensive in terms of development
- Let COSTA handle the interface and “teach” COSTA to do it
 - rapid development
 - cheap
 - some overhead
- Only a relevant subset “works”

Let COSTA handle it



Interfaces: State vector I

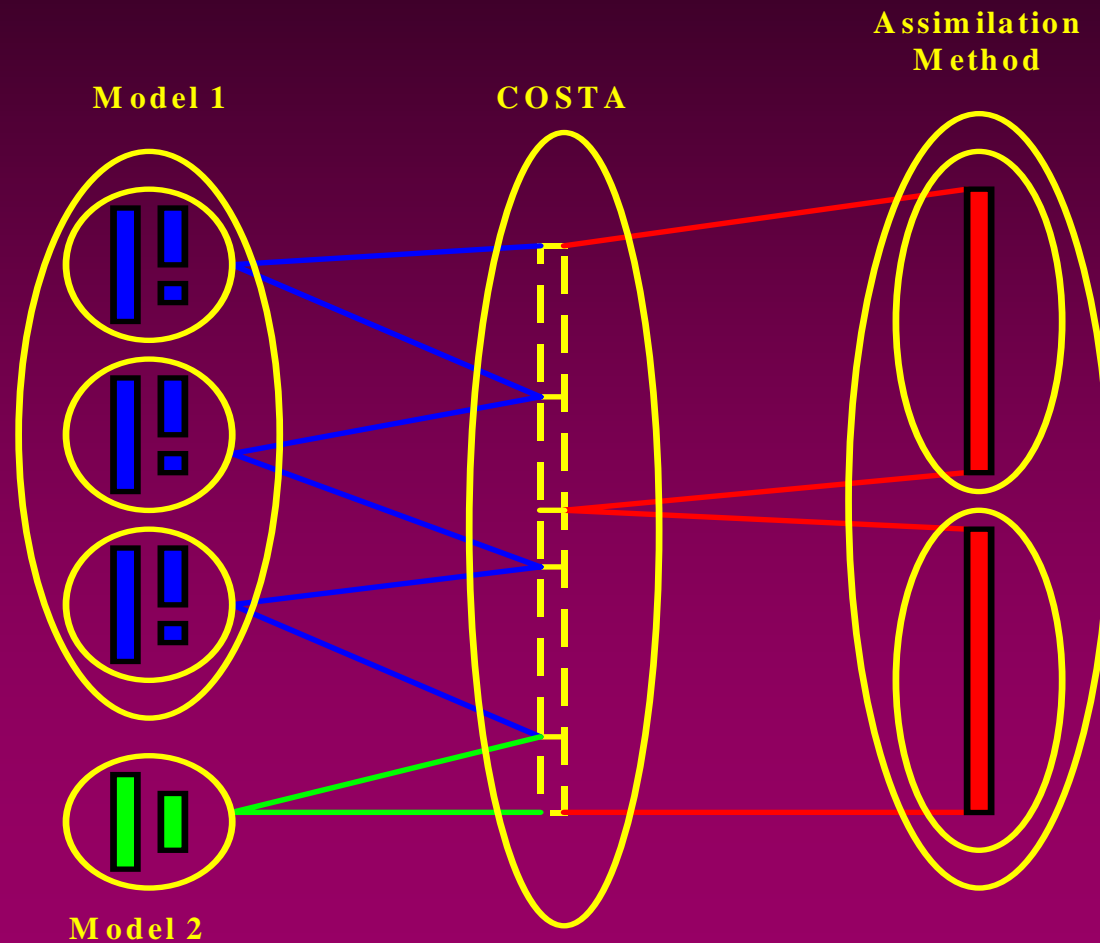
- Numbers that represent model state
- In assimilation method:
 - represented as a single vector
 - No knowledge on meaning of values only types/classification

Interfaces: State vector II

◉ In model:

- Scattered over (large) number of arrays
- Not available at one place
- Scattered over different processes (parallel computing and domain decomposition)
- Available through
 - Memory
 - Communication library
 - File
 - ?

Interfaces: State vector III



Interfaces: State vector IV

Challenges:

- Creating a sufficient and efficient interface between several (distributed) representations of the state vector
- Cooperation with communication libraries
- Domain decomposition
- Handling classification and lumping of variables

Costa and Parallel Computing

- COSTA project does not want to create a new communication library
- Use/extend free available libraries e.g.
 - PETSc (MPI)
 - ?
- Cooperate with used libraries

Users' Group I

- What is their role
 - No users, no need to create COSTA
 - Setting scope of COSTA
 - Real codes, methods and complications
 - Future developments
 - Languages?
 - FORTRAN77, Fortran90, C/C++
 - Perl
 - ?

Users' Group II

- Attend workshop/meeting twice a year
- Share information on your applications and assimilation methods
- Contribute methods to COSTA
- Apply COSTA in its development stage on your models