Development of generic data assimilation software for programmers and end-users: COSTA and DATools

N. van Velzen, S. Hummel, M. Verlaan, A. Weerts,
G El Serafy, J. Beckers, H. Gerritsen

WL | delft hydraulics

Ministerie van Verkeer en Waterstaat

Rijkswaterstaat

TU Delft

Outline

- Introduction (Albrecht Weerts)
- COSTA (Nils van Velzen)
- DATools (Albrecht Weerts)
- Conclusions (Albrecht Weerts)

Introduction

RIKZ/TUDELFT

- Need for generic approach data assimilation for RIKZ models
- Need for generic tools at Delft University to avoid duplication of work

WL

- Data assimilation was successfully applied in SOBEK (EKF; since 1992!!) and HBV (PF/EnKF); DA functionality is promising also for other models, specifically in FEWS => Need for a flexible and generically applicable DA tool(box)

Introduction

- COSTA: focus on software for programmers and researchers (users group: TNO-MEP, TNO-NITG, HKV, WLDelft, VORTECH)
-
- DATools: focus on end-user (users group: flood and other forecasters around the world that use Delft-FEWS or stand-alone process models for forecasting purposes)

Introduction

- DATools is (partly) funded from doelsubsidie provided by Dutch Government which is used after consultation with Rijkswaterstaat (M. Verlaan) (vraagsturing/Delta instituut etc.)
  ->link with COSTA was suggested

- Goals: to learn from each other and to avoid duplication of work, agree upon generic interfaces so we can make use of each others work

COSTA

- What is COSTA
- Why COSTA
- Components and interfaces
- Developments
- User's group and Cooperation

What is COSTA

- Problem solving environment for data assimilation and model calibration
  - Definition of components and interfaces
  - Tools for easy creation of components
  - “Off the shelf” data assimilation and calibration methods
  - Development philosophy
- Free software (LGPL)

Aims

- Save on developments costs
- Reuse methods and models developed by others or for other projects
- Quickly try out alternative methods
- Less errors in the software
- Application for open and commercial simulation models
- Relatively easy to migrate existing software

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 realization of a set of interfaces

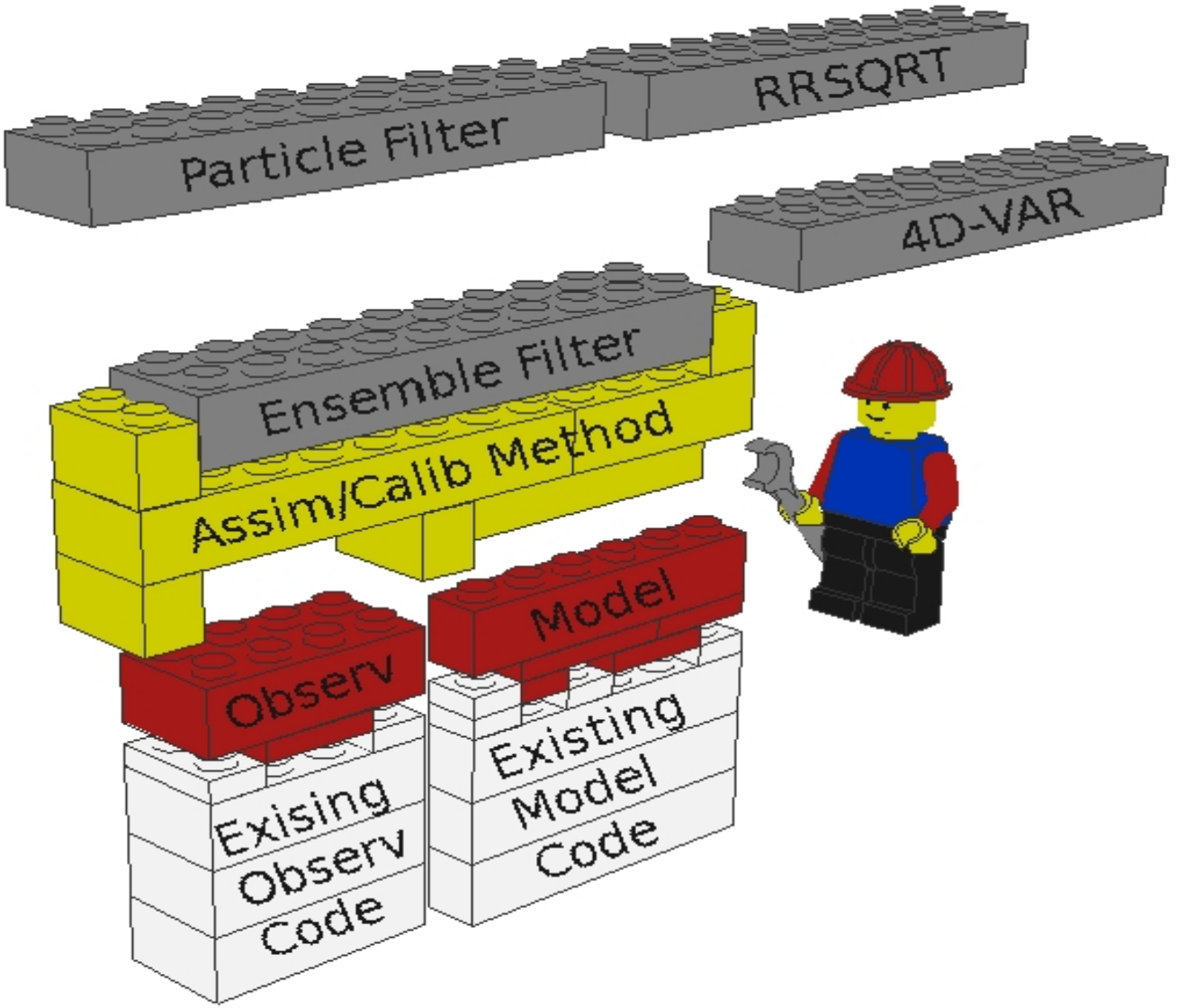
Components and Interfaces

- Components are the building blocks of the system
- Identification of components in data assimilation and calibration systems:
  - Model
  - Method
  - Observations
  - State

Components and interfaces

- Object oriented approach
  - Multiple instances of a component
  - Parallel computing
- Definition of powerful interface that is manageable (usage and performance)
- Testing of developed ideas

COSTA components



Building Components

- Use generic implementation or create own
- Interface potentially consists of large set of methods. No need to implement them all!
  - Redundancy: COSTA can implement the method
  - Missing: Component can only be used for limited set of assimilation calibration methods

Example model AXPY

- The analysis step of a Kalman filter
  - Pseudo code of x=x+dx:
    state=model.getstate
    state=state+delta_state
    model.setstate(state)
  - NOT FLEXIBLE
- Preferred in COSTA
  - model.axpy(1.0,delta_state)

Example model AXPY

- Preserve physical values in state:
  - Handle in filter
  - Handle in the setstate method
  - Handle in the axpy method
- COSTA has default implementation
  state=model.getstate
  state.axpy(1.0,delta_state)
  model.setstate(state)

COSTA developments (present)

- Base system available
  - Large number of small components
  - Model interface
  - Observation interface (Stochastic observer)
- Costa in combination with real models
  - WAQUA/TRIWAQ
  - Lotos-Euros

COSTA developments (near future)

- Model builders
  - Easy creation of COSTA model components
  - Easy creation of stochastic models from deterministic models
- Parallel computing in combination with COSTA
- Model calibration

COSTA users' group

- Group of data assimilation users from different institutes:
  - HKV
  - Rijkswaterstaat RIKZ
  - TNO (MEP, Geo wetenschappen)
  - TU Delft
  - VORtech
  - WL|Delft Hydraulics

Cooperation

- Kalimero (RIKZ)
  - Usage of COSTA in KALMINA:
    - Application of alternative methods
    - Usage of data assimilation and calibration for other applications than WAQUA/TRIWAQ
- DATools WLDelft
  - Remaining part of this presentation

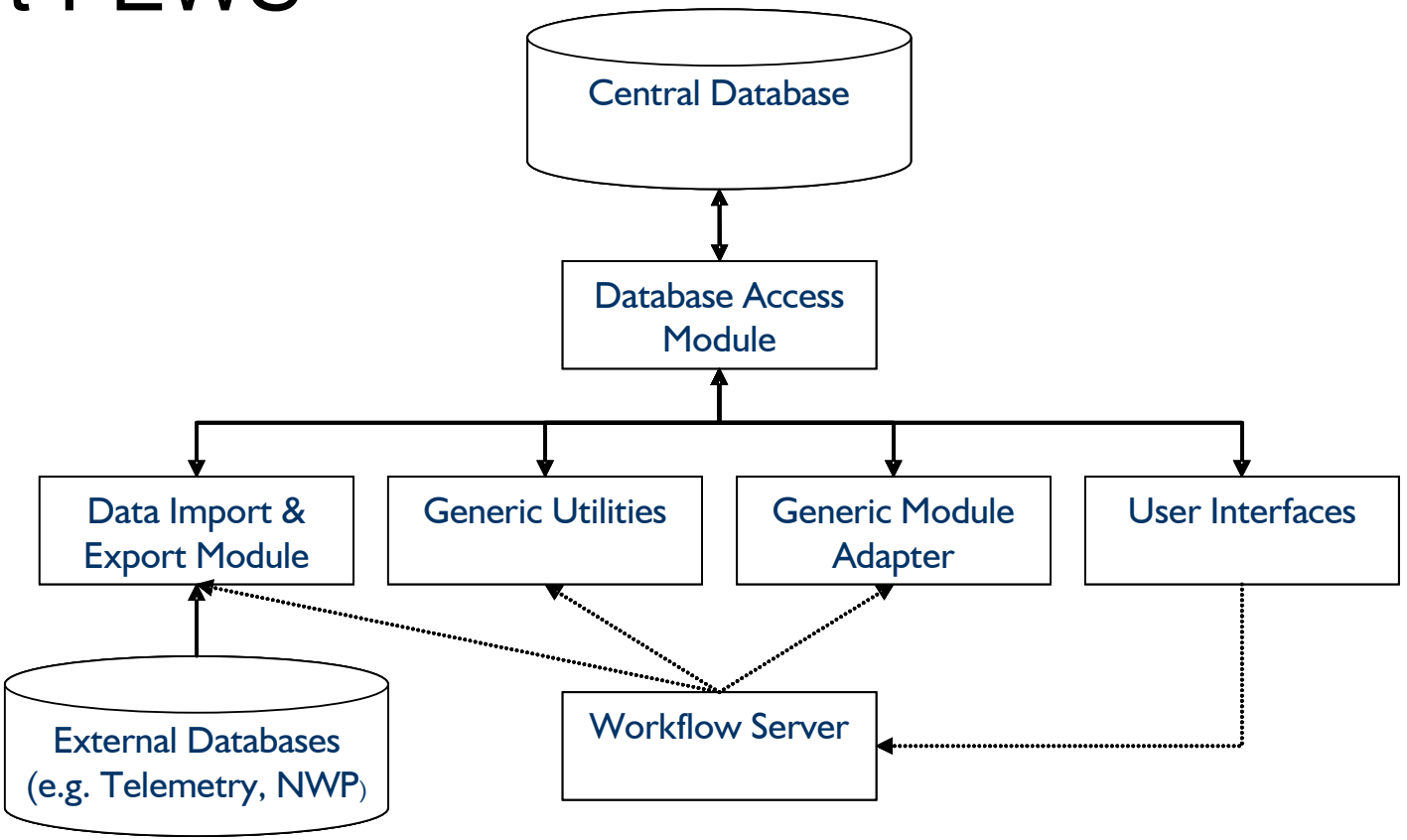
DATools

- Delft-FEWS:

- **Open shell system for managing forecasting process**
- **Shell provides general functional utilities**
- **Open interface to forecasting modules**
- **Being used in England& Wales (EA), Scotland (SEPA), Meuse & Rhine (RIZA), Switzerland (FOWG), Germany (BfG), Austria, Taiwan, Pakistan, EU (JRC Ispra) , RIKZ (algea)**
- **To be used in Italy (Po river), Singapore,...**
- **need for a data assimilation module**

DATools

- Delft-FEWS

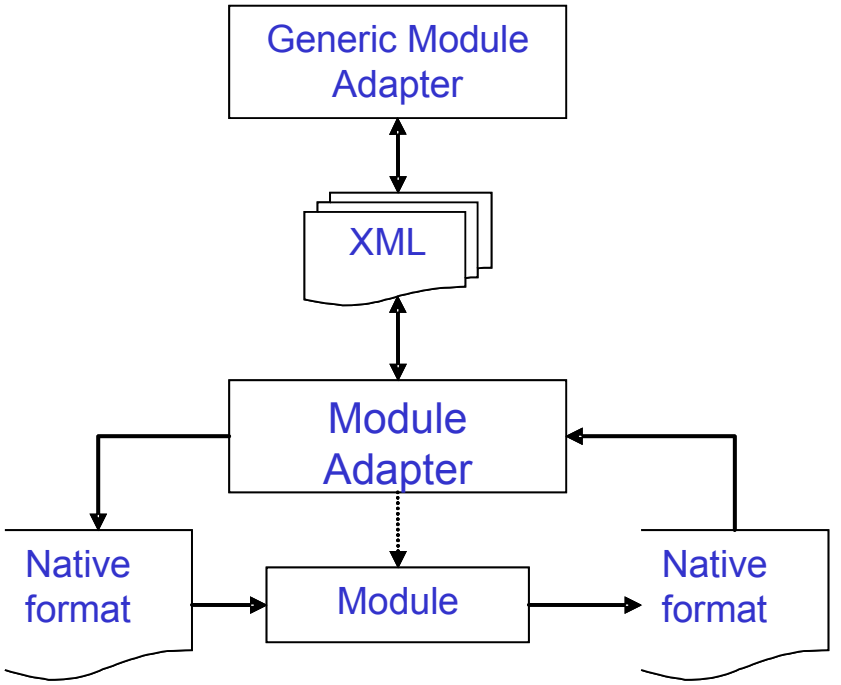


DATools

Linking Hydrological and Hydraulic models

General Adapter Module

- Published interface (XML)
- Module adapters
- HarmonIT interface
  (under development)



examples of models linked…

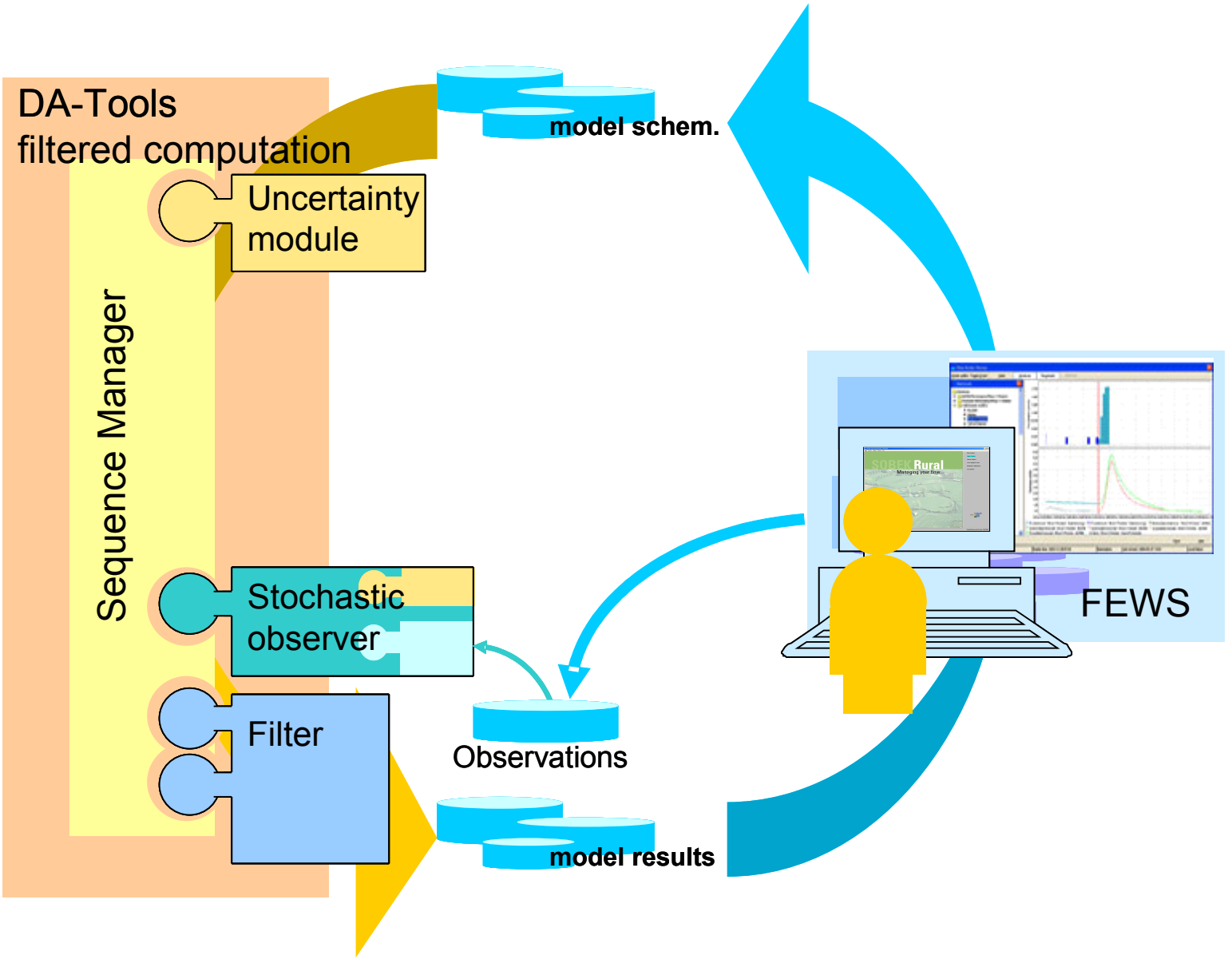
- SOBEK & Sacramento (Delft Hydraulics)
- ISIS (HR), PDM & KW (CEH), HBV (SMHI), SYNHP (BfG), Mike11 (DHI), PRMS (Karslruhe), Vflo (USA), etc…

DATools

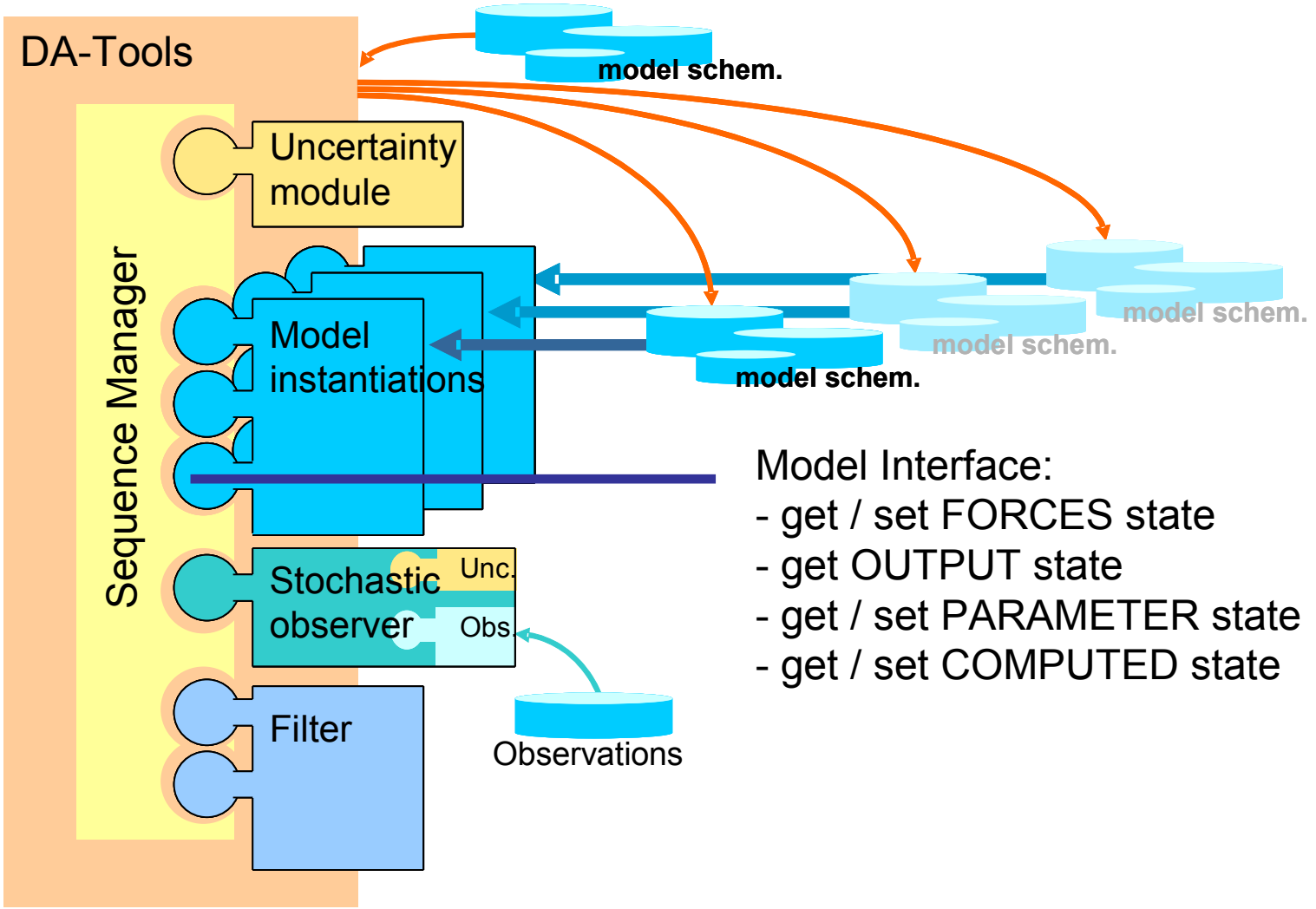
- Four components:

- Sequence Manager
- Filter (PF/EnKF)
- Stochastic Observer (COSTA/DATools)
- Stochastic Modeler (COSTA/DATools)

DATools

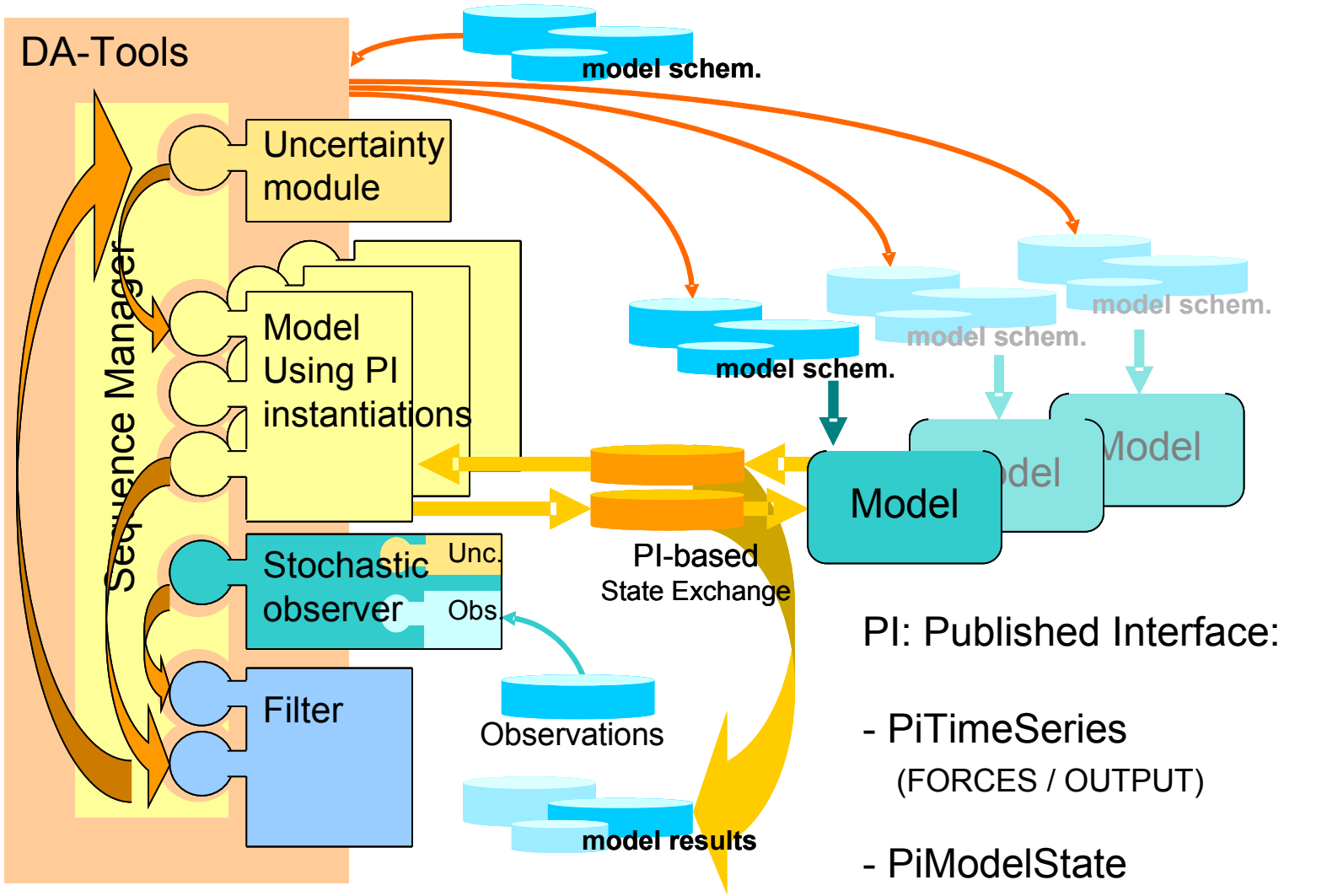

DA-Tools
filtered computation
Sequence Manager
Uncertainty module
Stochastic observer
Filter
model schem.
Observations
model results
FEWS

DATools



Model Interface:
- get / set FORCES state
- get / set OUTPUT state
- get / set PARAMETER state
- get / set COMPUTED state

DATools



PI: Published Interface:

- PiTimeSeries
  (FORCES / OUTPUT)

- PiModelState
  (PARAMETER / COMPUTED)

DATools

- DUE data uncertainty engine developed by James Brown (UVA, CBBG): user friendly GUI to specify uncertainties
- Model using PI-timeseries or “OpenMI”-approach
- Filter: EnKF and Particle Filter (Residual Resampling)
- Test are now being conducted with HBV-96, SOBEK & Delft-3D
- User friendly (configuration of XML files)

Conclusions

- Two lines of development but with interaction
- development of generic interfaces => stochastic observer & modeler (model component)
- COSTA is available from internet: www.costapse.org

- Future work: Testing DATools in COSTA and vice versa