Adapting the WAQUA/TRIWAQ model for application in COSTA

Nils van Velzen

7th International Workshop on Adjoint Applications in Dynamic Meteorology

October 2006

TU Delft

VORTECH BV
COMPUTING AND SIMULATION SOFTWARE TECHNOLOGY

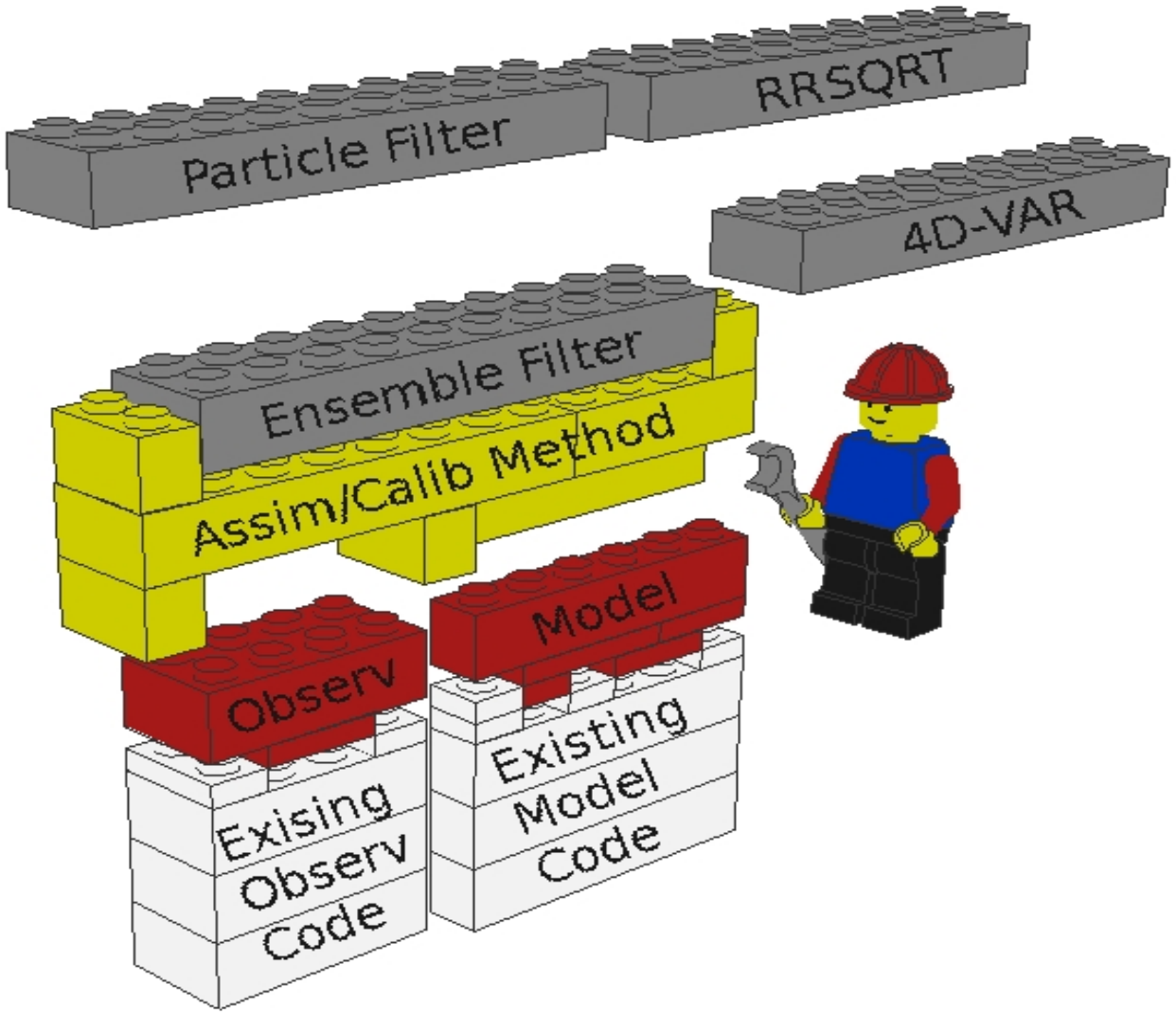
Outline

- COSTA overview
- COSTA model component
- The WAQUA/TRIWAQ model
- RRSQRT in a nutshell
- Deterministic and stochastic model
- Drying and flooding
- Conclusions

COSTA

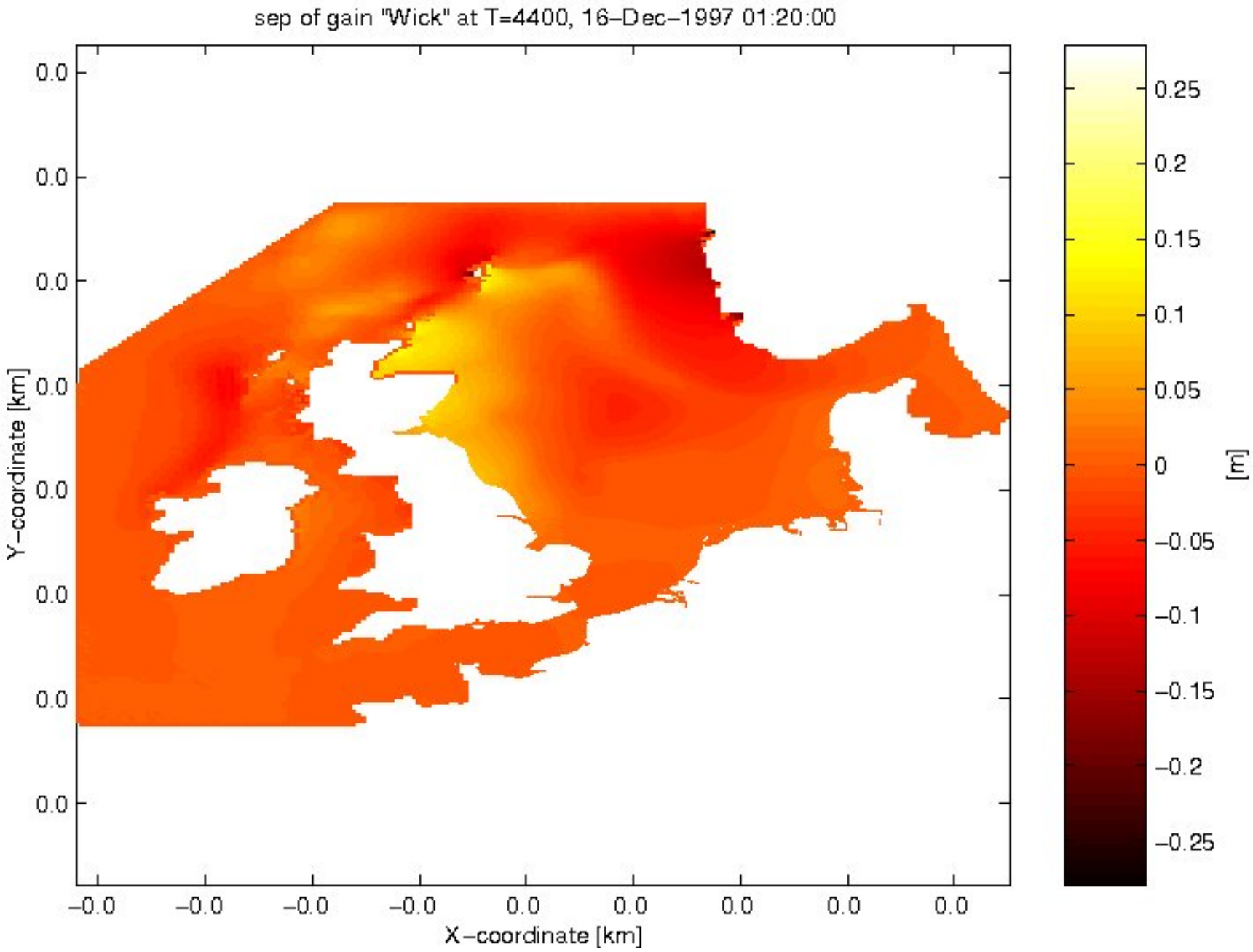
- A problem solving environment for data assimilation and calibration
- Object oriented approach
  - Components and their interface
  - Data assimilation methods
  - (Software) development philosophy
- Free software (LGPL)

COSTA


Particle Filter
RRSQRT
4D-VAR
Ensemble Filter
Assim/Calib Method
Observ
Model
Exising
Observ
Code
Existing
Model
Code

The WAQUA/TRIWAQ model

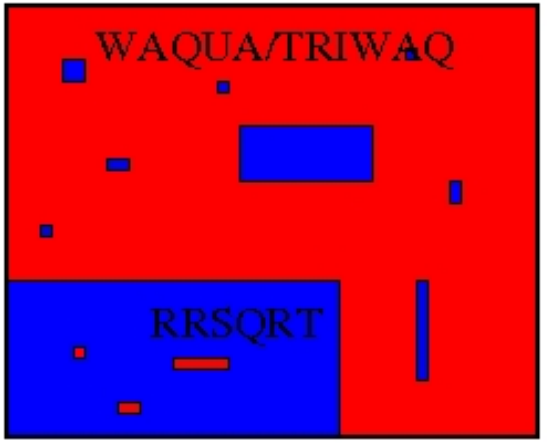
- 2-D and 3-D shallow water simulation
- Data assimilation:
  - RRSQRT Kalman
  - Ensemble Kalman
  - Steady state Kalman
  - Calibration (adjoint/finite difference)
- Real model including:
  - Scaling
  - Discontinuities (drying and flooding)


sep of gain "Wick" at T=4400, 16–Dec–1997 01:20:00
Y–coordinate [km]
X–coordinate [km]
0.0
0.0
0.0
0.0
0.0
0.0
0.0
–0.0
–0.0
–0.0
–0.0
0.0
0.0
0.0
0.0
0.0
0.25
0.2
0.15
0.1
0.05
0
–0.05
–0.1
–0.15
–0.2
–0.25
[m]

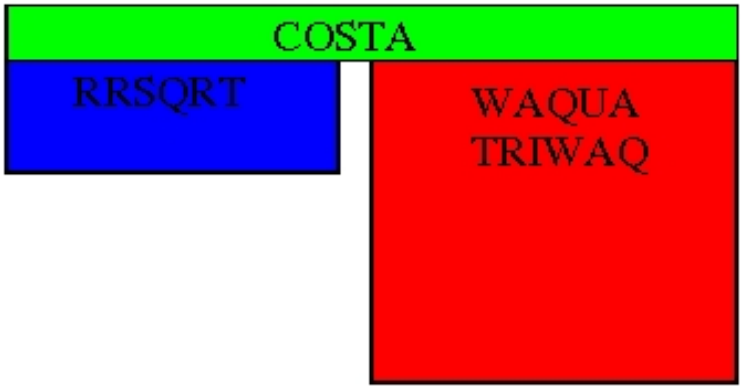
Challenge

- Can we:
  - separate filter and model?
  - have same results?
  - similar performance?
- Is the COSTA model interface sufficient for this task?

Before:



After:

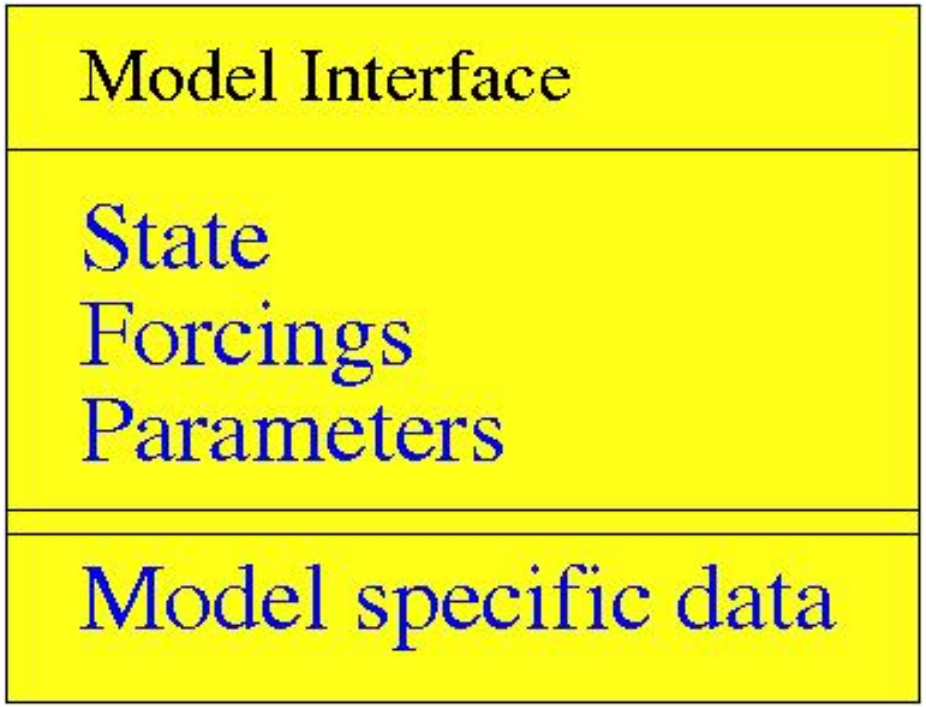


Model Component

- Form of a model in COSTA (currently)

$$\phi(t_{i+1}) = A\left[\phi(t_i), u(t), g\right]$$

- Model interface:
  - Get, set
  - Level 1 BLAS
  - Compute
  - GetObsValues



The WAQUA/TRIWAQ model

- Deterministic model:

$$\phi(t_{i+1}) = A\left[\phi(t_i), u(t), g\right]$$

- Stochastic model:

$$\begin{pmatrix} \phi(t_{i+1}) \\ p^u(t_{i+1}) \\ p^A(t_{i+1}) \end{pmatrix} = \begin{pmatrix} A\left[\phi(t_i), u(t_{i+1}) + W^u p^u(t_i), g\right] + W^u p^u(t_i) \\ \mathrm{diag}\left(\alpha^u\right) p^u(t_i) \\ \mathrm{diag}\left(\alpha^A\right) p^A(t_i) \end{pmatrix} + \begin{pmatrix} 0 \\ \eta^u(t_i) \\ \eta^A(t_i) \end{pmatrix}$$

or

$$x^t(t_{i+1}) = M_i\left[x^t(t_i)\right] + \eta(t_i)$$

RRSQRT in a nutshell

$$P_{i+1}^{f} = M_i P_i^{a} M_i^{T} + Q_i$$

$$x_{i+1}^{a} = x_{i+1}^{f} + K_{i+1}^{o}\left(y^{o} - H_{i+1} x_{i+1}^{f}\right)$$

$$K_{i+1}^{o} = P_{i+1}^{f} H_{i+1}^{T}\left(H_{i+1} P_{i+1}^{f} H_{i+1}^{T} + R\right)^{-1}$$

$$P_{i+1}^{a} = P_{i+1}^{f} - K_{i+1}^{o} H_{i+1} P_{i+1}^{f}$$

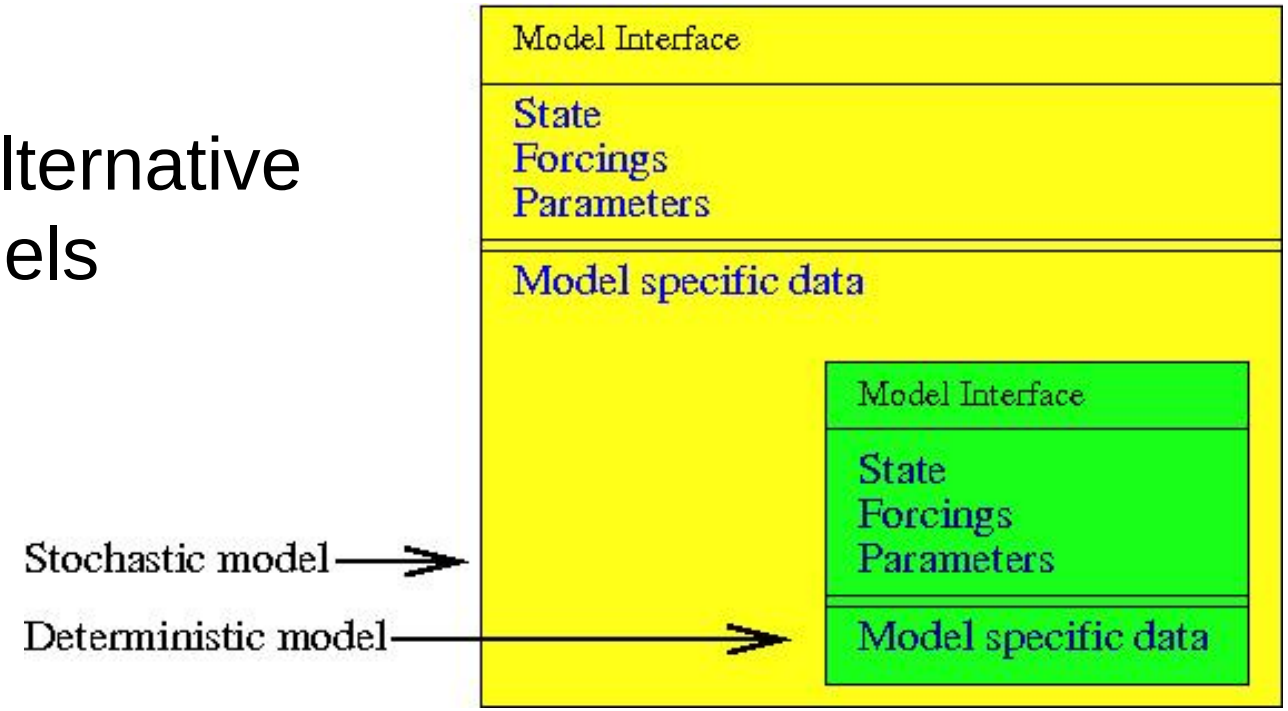
$$L_{i+1}^{w} = \left[M_i L_i^{a}, \sqrt{Q_i}\right]$$

$$\left(sL_{i+1}^{w}\right)^{T}\left(sL_{i+1}^{w}\right) = V_i D_i V_i^{T}$$

$$L_{i+1}^{f} = \left[L_{i+1}^{w} \quad V_i\right]_{:,1:n}$$

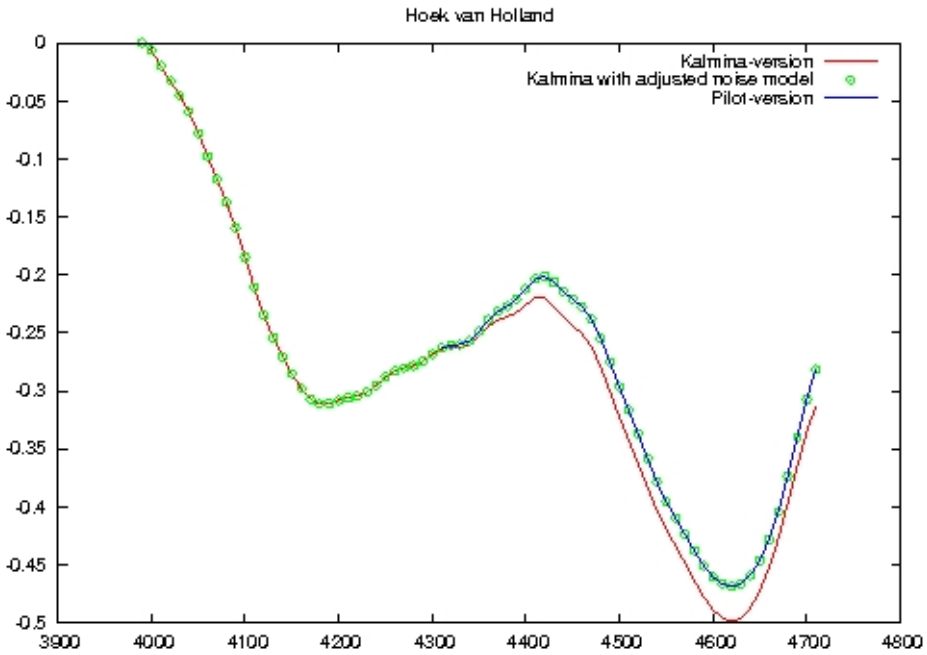
Deterministic and Stochastic

- We want a separation between deterministic and stochastic model
  - Reuse deterministic model
  - Play with alternative noise models
  - Smoother

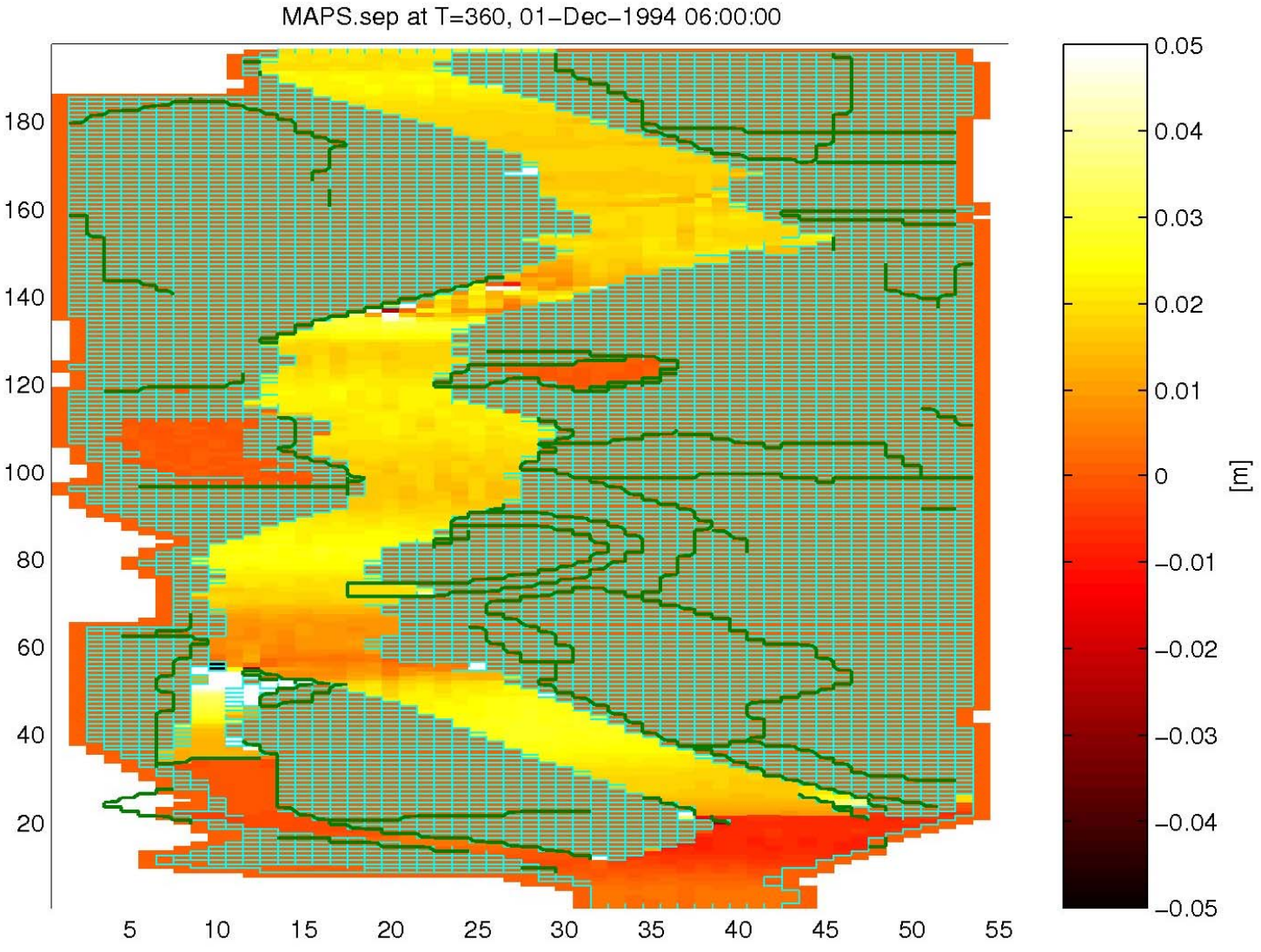


Deterministic and Stochastic

- WAQUA/TRIWAQ uses ADI-scheme:
  - One time step consist of two halve time steps
  - Time correlation applied each halve time step
- New approach:
  - Time correlation each whole time step
  - Different results!



Drying and flooding



Drying and flooding

- Screens cannot computed from state
- State contains (integer) flags $\phi_Z^t\left(t_i\right)$

$$x\left(t_i\right)=\begin{pmatrix}\phi_R^t\left(t_i\right)\\ \phi_Z^t\left(t_i\right)\\ p^u\left(t_i\right)\\ p^A\left(t_i\right)\end{pmatrix}$$

- RRSQRT cannot handle $\phi_Z^t\left(t_i\right)$

Drying and flooding

- Propagation of L-matrix

$$L_{i+1}^{w} = \left[M_i L_{i+1}^{a}, \sqrt{Q_i} \right]$$

  - Compute for all modes $l_j = L(:, j)$

$$l_j^w\left(t_{i+1}\right) = M_i\left[x^t\left(t_i\right) + l_j\left(t_i\right)\right] - x^t\left(t_{i+1}\right)$$

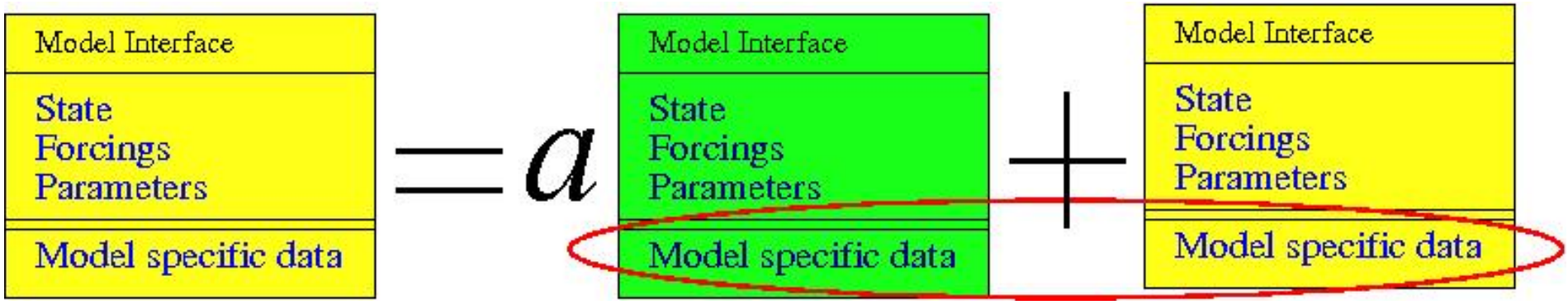
  - Set elements of $l_j\left(t_{i+1}\right)$ to 0 when corresponding element in $\phi_Z\left(t_i\right)$ is different

- Model specific RRSQRT implementation!

Drying and flooding

- Move knowledge into the model
- Use Axpy $\left(y = ax + y\right)$ for models in the filter to implement :

$$l_j^w\left(t_{i+1}\right) = M_i\left[x^t\left(t_i\right) + l_j\left(t_i\right)\right] - x^t\left(t_{i+1}\right)$$



The Results

- WAQUA/TRIWAQ model component:
  - Deterministic model
  - Noise model
  - Smoother
- Create equivalent COSTA RRSQRT filter into a COSTA filter, already used for Lotos-Euros air pollution model!
- Correct simulation results

Conclusions

- It is possible to create generic reusable data assimilation for general applications
  - Keep information where it belongs in the software!
- Existing non-object oriented simulation models can easy be transformed into COSTA models