Deltares
Enabling Delta Life

OpenDA, a generic toolbox for data assimilation in numerical modelling

Martin Verlaan[1,2], Nils van Velzen[3,2], Stef Hummel[1], Herman Gerritsen[1],

and many more colleagues of

[1] Deltares, [2] VORtech, [3] TU Delft

[1,2,3] info@openda.org

What do we mean by “data assimilation”?

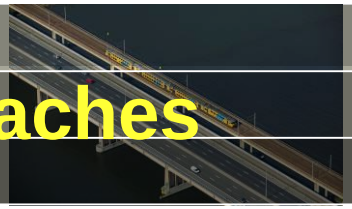
Techniques and methodologies for :

- Improvement of real time forecasts
- Calibration of uncertain model parameters
- Uncertainty analysis
- Study the potential value of new observations
- Estimation of uncertain sources

“formal techniques for structured integration of data and models”

Deltares

Common / “traditional” approaches



Sensitivity analysis / Model calibration:

- single parameter variation simulations → parameter optimisation
- tailored (ad-hoc / built-in) parameter optimisation routines
- generally no accounting of uncertainty
- dedicated built-in adjoint model codes; e.g. WAQAD

Model forecast improvement:

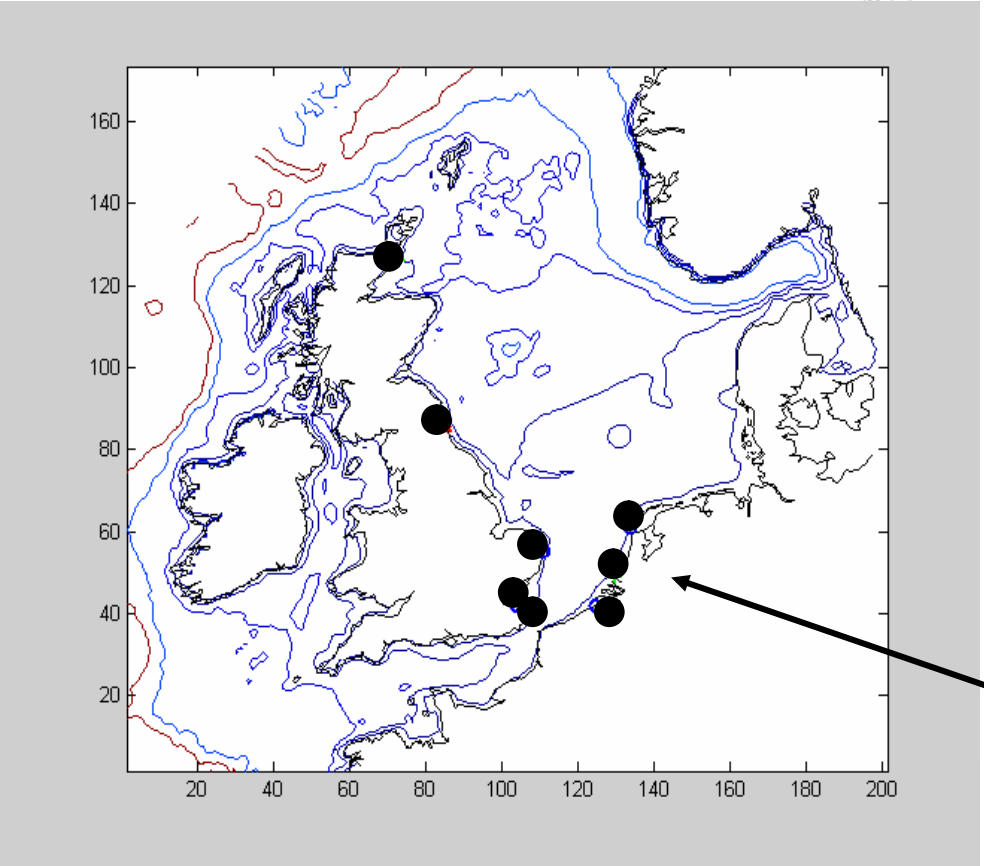
- built-in (extended) Kalman filter routines; e.g. in WAQUA (~ 1990)
- 3D-var; 4D var, dedicated for a specific model and a specific domain (mainly in atmospheric modelling)
- large investments in monitoring network to match assimilation needs

→ *ad-hoc and dedicated (single purpose) developments*

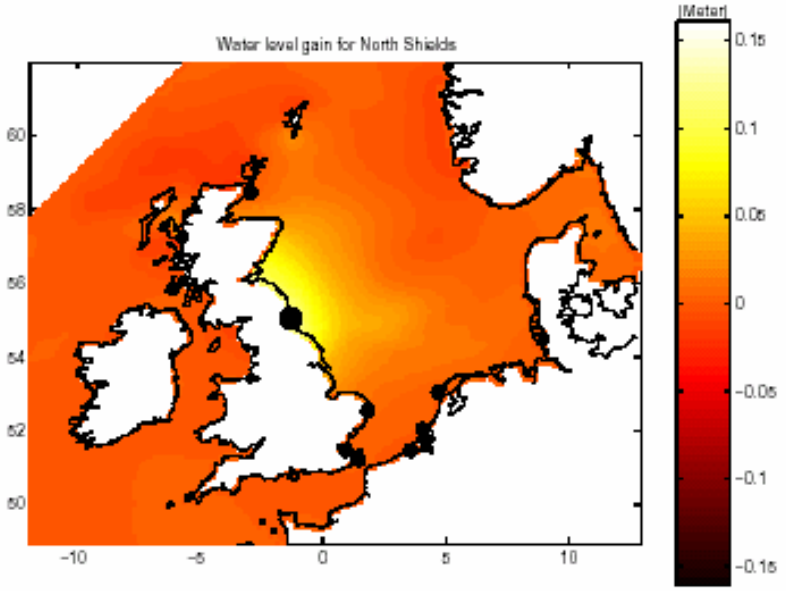
Deltares

Example: Dutch operational storm-surge model

Steady state Kalman Filter



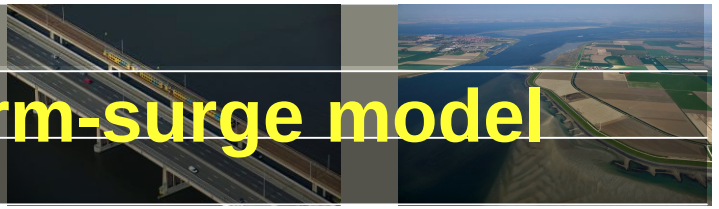
Kalman Gain Station Wick



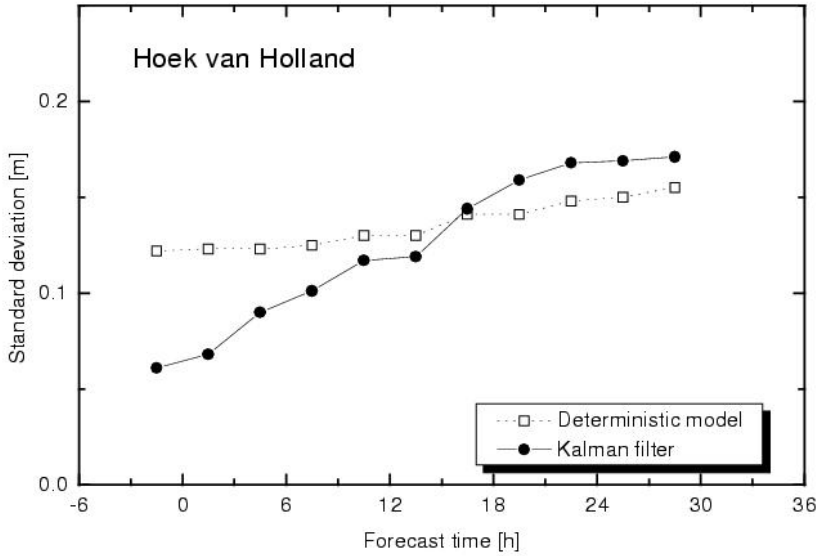
Observation stations

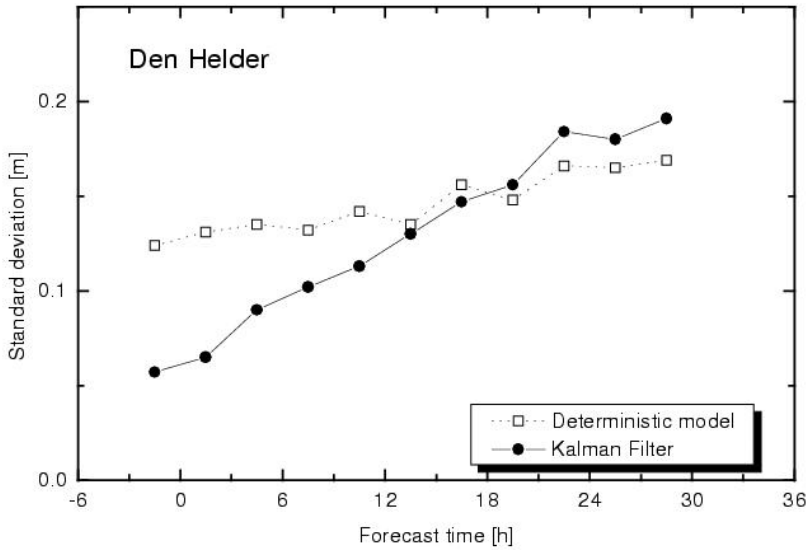
Deltares

Example: Dutch operational storm-surge model



Forecast improvement as function of forecast time





Deltares

OpenDA

What is OpenDA?

- A generic toolbox for data-assimilation
  - library of data-assimilation algorithms
  - set of interfaces that define interactions between components

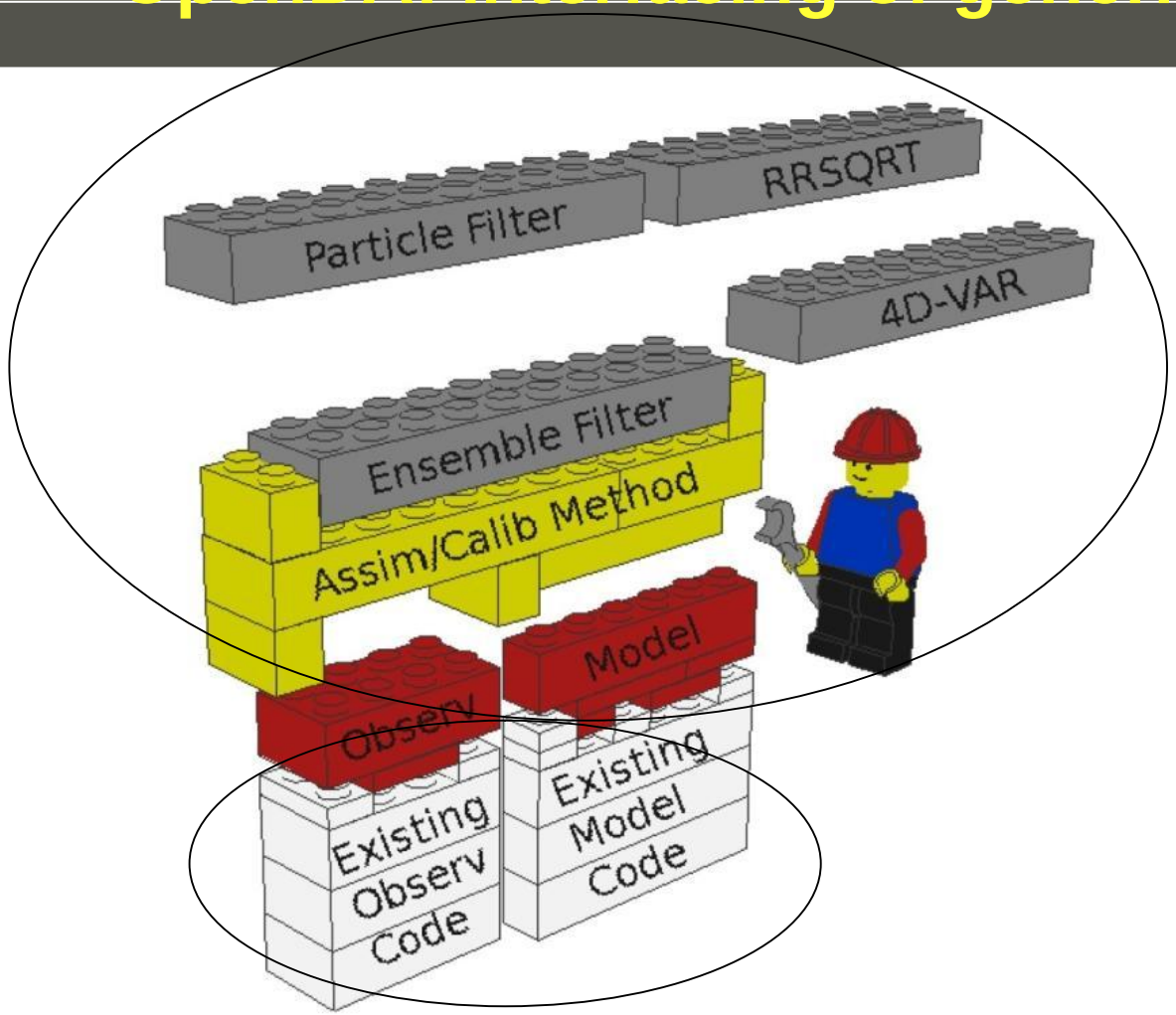
Why OpenDA?

- Provides an environment for development of algorithms
- Reuse of components – robustness
- Portability
- More efficient than dedicated development for each application

→ Overall cost effectiveness

Deltares

OpenDA: interfacing of generic elements

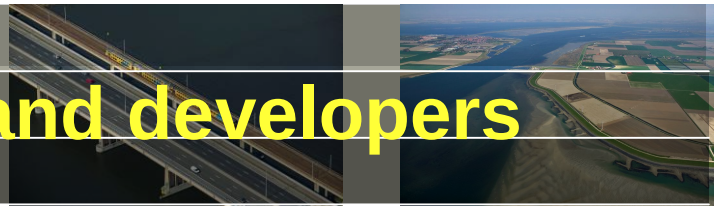


there is full separation
of
process model parts
and
data assimilation parts

need for one time definition of the interface (“wrapper”)
between the process model code and OpenDA

Deltares

Benefits for user community and developers



- Generic applicability and flexibility
- Access to a range of available algorithms
- Access to a range of available intuitive uncertainty prescription options
- Same look and feel, independent of process model and (type of) application
- Steep learning curve
- Portability

- Efficient environment for data assimilation developers
- Easy switching between and testing of algorithms
- Re-use and therefore robustness of components

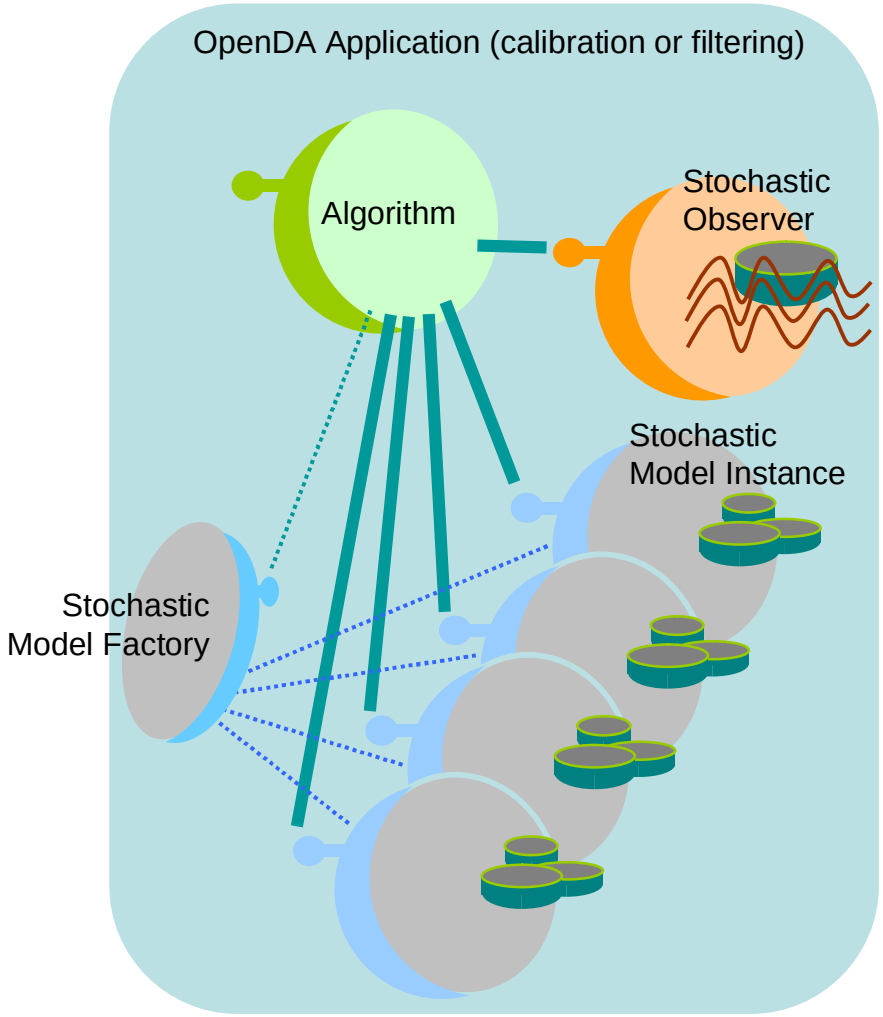
Deltares

OpenDA development approach

- OpenDA merges the COSTA and DATools developments which started in 2002
- → Champion users in Deltares, VORtech, TUDelft/EWI, plus BMT-Argoss, TNO-MEP, KNMI
- Architecture is based on java – thoroughly tested
- First optimisation algorithms: DUD (Ralston&Jennrich, 1978), Powell (1964), Simplex (Nelder-Mead, 1965)
- First filters: EnKF, RRSQRT, ensrf, particle filters
- Presently, wrappers exist for the model codes WAQUA, SWAN, Delft3D-FLOW, SOBEK-RE, FEWS, LOTOS-EUROS, CHIMERE, ...

Deltares

Models in an OpenDA Application with wrapper

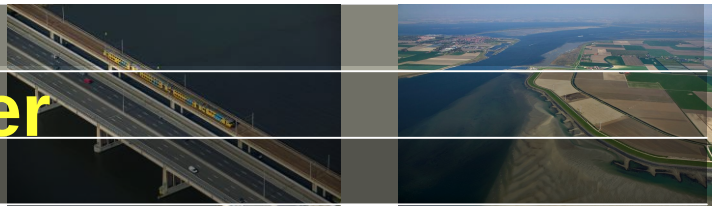


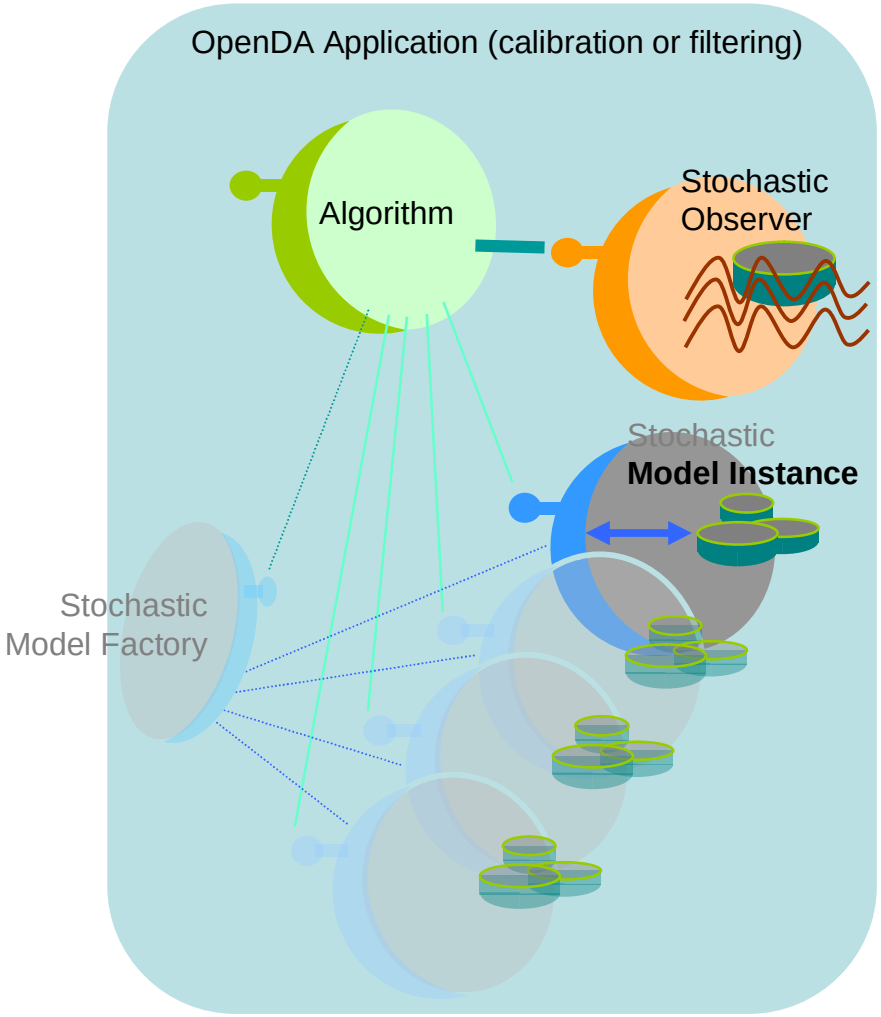
To let a model run in an OpenDA application, the user must:

- configure the **StochModelInstance** interface:
  - > Add stochastic information to parameters, computed state, and/or results
  - > Get and/or Set the necessary values from and/or into the model
  - > Let the model compute one or more time steps
- configure the **StochModelFactory** interface:
  - > Create model instance (let the model clone itself)

Deltares

Tasks of the black box wrapper





The Black Box wrapper:

- Only requires one ‘instance’ (Model Template)
  - > clones the template to create instances
- Standardizes the way values are set to and retrieved from the model files
- Takes care of the stochastic aspects:
  - > uncertainty specification
    - add uncertainty to parameters, computed state, and/or results

Deltares

Example: OpenDA calibration of depth in DCSM

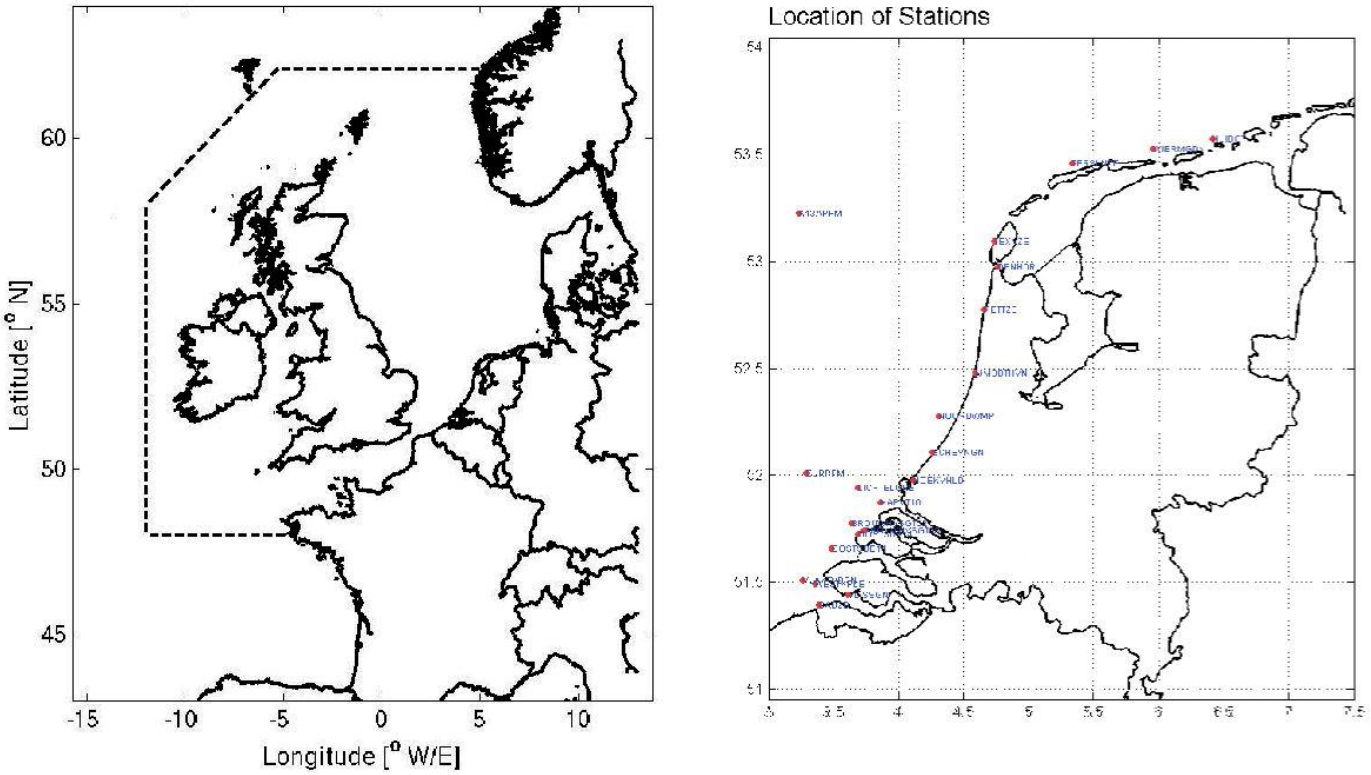
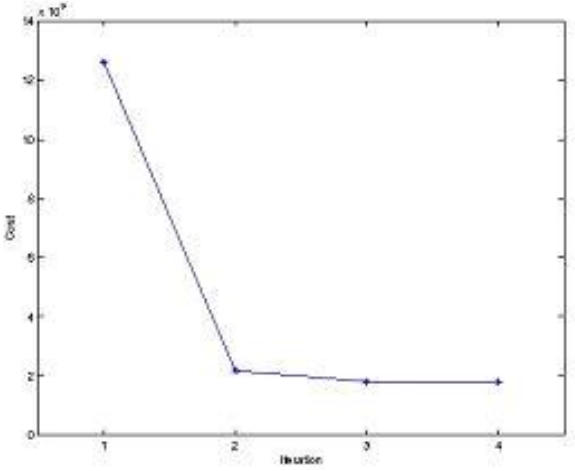


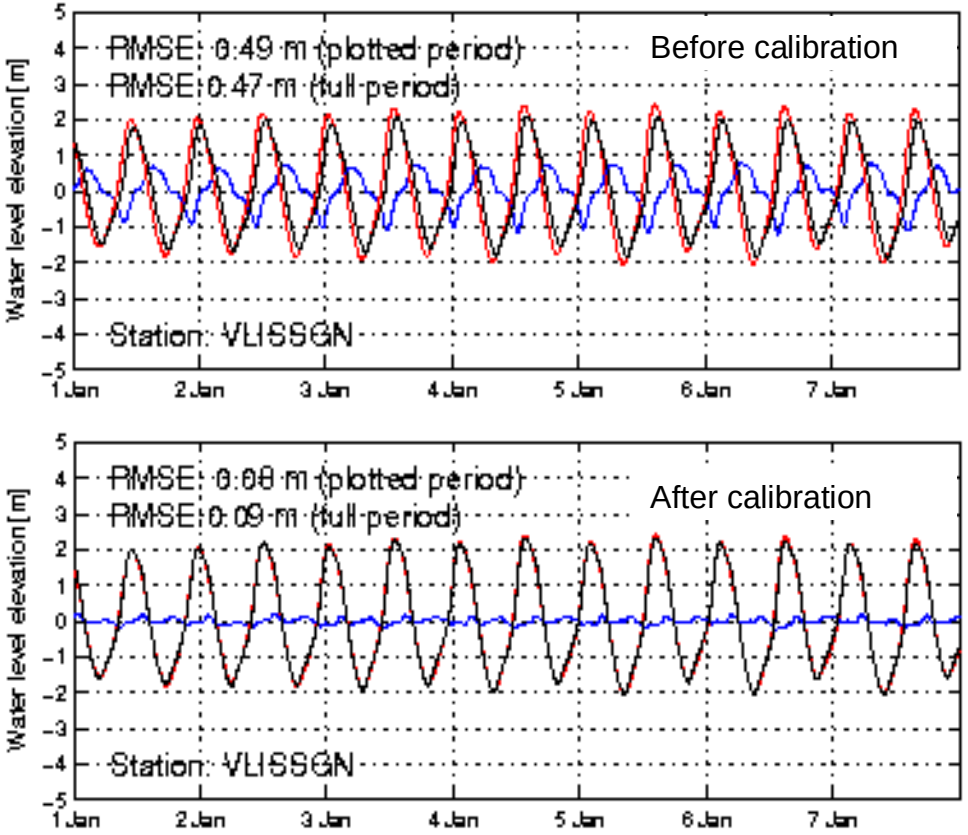
Figure 1: Dutch Continental Shelf Model: version 5 (dashed-line, grid cell ~10x10 km2, 12oW-13oE and 48oN-62o) and version 6 (grid cell ~2x2 km2, 12oW-13oE and 48oN-62oN). The right-hand side picture shows area of interest (Dutch coast) and water-level observation locations used for calibration.

Deltares

Example: OpenDA calibration of depth in DCSM

$$GoF = \frac{1}{2}\sum_{r=1}^{r=R\max}\sum_{s=1}^{s=S\max}\sum_{n=1}^{n=N\max} w_{r,s}\left(H_{r,s,n}^{sim}(t) - H_{r,s,n}^{obs}(t)\right)^2 \Big/ \left(\sigma_{Hobs}\right)^2$$





above: GoF decrease with DUD iteration;

right: Improvement of tidal results for Vlissingen

Deltares

Present status

- OpenDA will be available as open source under LGPL conditions
- OpenDA association presently consists of Deltares, VORtech, TUDelft; the legal documents signed
- Presently available:

  - Windows distribution of OpenDA1.0
  - Documentation and examples

- Release date of OpenDA1.0 – Today, 11 May 2010
- Shortly (in a few weeks):

  - Linux distribution + sources of OpenDA1.0

Deltares

So?

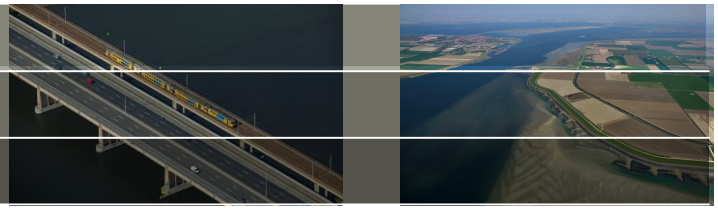
- Release date of OpenDA1.0 – Today, 11 May 2010
- Several speakers will present further developments and examples
- Several flyers with example applications are available (in the rear of the auditorium / on website)

→ it is there for you to use, for all to share, why not give it a try?

www.openda.org

info@openda.org

Deltares



Comments?

Questions?

Deltares