Sustainable scientific software: experiences of the PCRaster research and development team

Oliver Schmitz, Kor de Jong and Derek Karssenberg
Department of Physical Geography, Utrecht University, The Netherlands (o.schmitz@uu.nl)

Motivation and approach
The PCRaster modelling framework
- Developed since 1989, open source since 2007
- Is targeted at the development of spatio-temporal simulation models
- Fast model development and execution
- Scripting environments: PCRcalc and Python
- Rich set of model building blocks for manipulating raster maps
- Framework for stochastic spatio-temporal model building and data assimilation
- Tool for visualisation of spatio-temporal stochastic data

Application examples
Hydrological modelling
PCRaster-GLOBWB 2, a 5 arc-minute global hydrological & water resources model
Daily time step, spatial resolution of 0.5° (~50 km), 5° (~10 km) or ~1 km for regional studies

Land use change
Potential for bioenergy production with PLUC, the PCRaster Land Use Change model

Research & Development
Our current work focuses on two topics, the development of one modelling environment for the construction of integrated field-based and agent-based models, and the development of operations making advantage of multi-core systems and computer clusters.

Integrating fields and agents
We develop a unifying conceptual data model to store continuous fields and discrete agents:
- Phenomenon: e.g. birds, groundwater
- Property set: Collection of properties sharing the same spatial and temporal domain
- Domain: Information on space and time (e.g. location of birds, subsoil volume)
- Property: Attribute (e.g. weight of bird, groundwater pollution level)
- Value: Magnitude of a property
- Item: Identifies an individual

Sustainability challenges
Sustaining the development
- Acquiring funding is major issue, software development is often a miniscule part of research proposals
- Providing tailored consultancy or maintenance contracts
- Full-time software development not always guaranteed

Maintaining our user base
- High initial effort to create documentation or courses
- Unpredictable moments and amount of user support
- Open source is not automatically the silver bullet: reported issues yes, but no pull requests so far

Maintaining the code base
- Version control and unit testing less familiar to environmental scientists
- Aligning with 3rd party dependencies (Qt, Boost, Gdal, ...)
- Packaging or building for different operating systems
- Dealing with legacy code, oldest C code from early 1990s
- Programming languages advance: C11, C++20; Python 2, Python 3
- Support of new hardware (e.g. GPU) requires entirely new code

Attribution of research software
- Traditional by publications, but these were often on environmental research topics
- Nowadays changing: DOI assignment, ‘Software availability’ sections, RSE communities, Computer Science oriented journals
- Still, software development is not part of a measurement, e.g. the h-index

Additional information
http://www.pcraster.eu
http://www.pcraster-project.org
http://www.pcraster-globe.org

Parallel and distributed computing
We enhance the PCRaster model building framework with built-in capabilities to run models on various hardware platforms, resulting in hardware scalable models that can be constructed by environmental modelers.

PCRaster on shared memory systems
A binding between PCRaster and Fern provides about 50 parallel operations making advantage of multi-core systems and computer clusters.

Integrating activities and mobility patterns to estimate personal exposure to air pollution;

Environmental health
Calculation of high resolution (5x5 m) land use regression models for different air pollutants.

Uncertainty estimates in direct or indirect land use change (Knooren 2015)

Hydrological modelling

PCRaster modelling framework
for each n in Monte Carlo samples: for each t in time steps:

\[
S_t^{(n)} = f(S_{t-1}^{(n)}, I_t^{(n)}, P_t^{(n)})
\]

Solution scheme for stochastic modelling
- state variables
- inputs
- parameters
- transition function

Solution framework (Python)

```python
from pcraster import * from pcraster.framework import *

class SnowModel(domain, boundarystate): def __init__(self): self.energy = 0 self.temperature = 0

def process(self):... def setup(self):... def report(self):... def __str__(self):...

def runtime_template():
```

Solution scheme for data assimilation
- state variables
- inputs
- parameters
- transition function

Solution framework (Python)

```python
from pcraster import * from pcraster.framework import *

class SnowModel(domain, boundarystate): def __init__(self): self.energy = 0 self.temperature = 0

def process(self):... def setup(self):... def report(self):... def __str__(self):...

def runtime_template():
```

Land use change

Potential for bioenergy production with PLUC, the PCRaster Land Use Change model

Sustainable scientific software: experiences of the PCRaster research and development team

Oliver Schmitz, Kor de Jong and Derek Karssenberg
Department of Physical Geography, Utrecht University, The Netherlands (o.schmitz@uu.nl)