

IS STREAMING AN ALTERNATIVE FOR POST-PROCESSING WORKFLOWS?

4TH ENES HPC WORKSHOP, TOULOUSE 2016

Luis Kornblueh, Uwe Schulzweida, Oliver Heidmann, Asela Rajapakse
April 11, 2016



Max-Planck-Institut
für Meteorologie

I am neither paid by WMO nor any NWP centre!

I always use libraries to access meta-data and data packed in grib!

I always use libraries to access meta-data and data packed in netCDF/hdf5!

MOTIVATION

- Optimization of data handling: `map(reduce(input data))` — functional aspect
importance of sequence often not understood by users

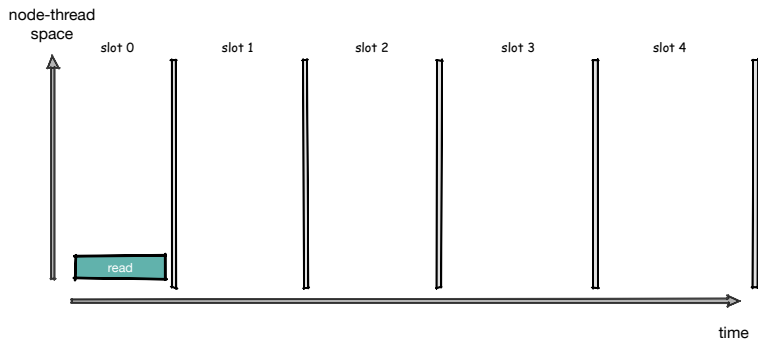
- Optimization of data handling: `map(reduce(input data))` — functional aspect
importance of sequence often not understood by users
- Increase of resource utilization
make resources available as early as possible

- Optimization of data handling: `map(reduce(input data))` — functional aspect
importance of sequence often not understood by users
- Increase of resource utilization
make resources available as early as possible
- Fully distributed processing model, and

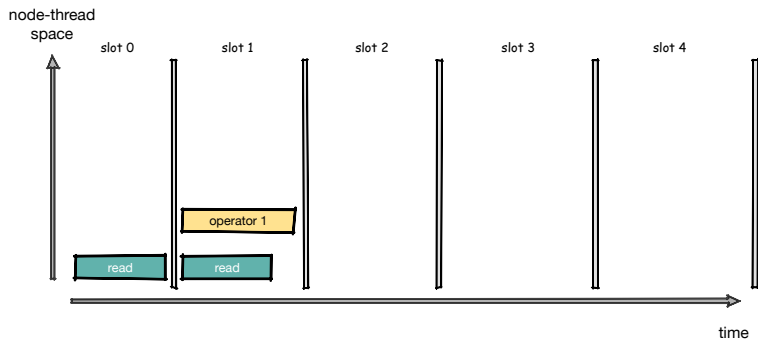
- Optimization of data handling: `map(reduce(input data))` — functional aspect
importance of sequence often not understood by users
- Increase of resource utilization
make resources available as early as possible
- Fully distributed processing model, and
- Functional formulation of processing chain allows for using compiler optimization techniques

A VECTOR PIPELINE PROCESSING MODEL

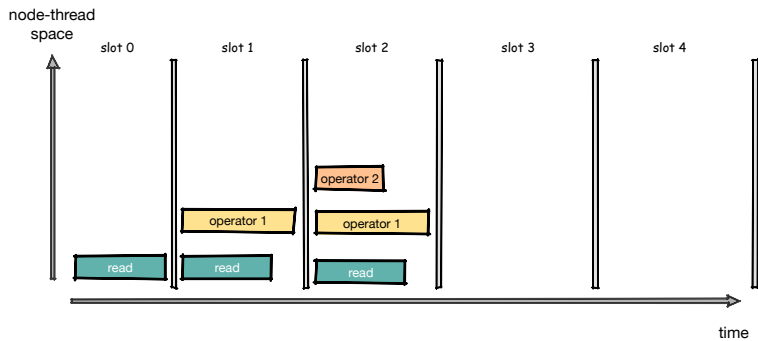
HOW A VECTOR PIPELINING PROCESSING MODEL WORKS



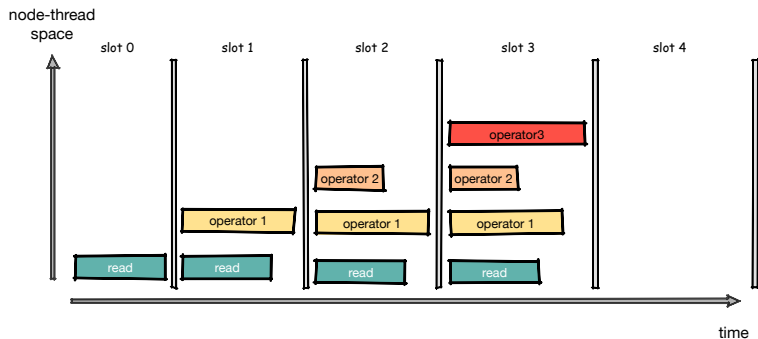
HOW A VECTOR PIPELINING PROCESSING MODEL WORKS



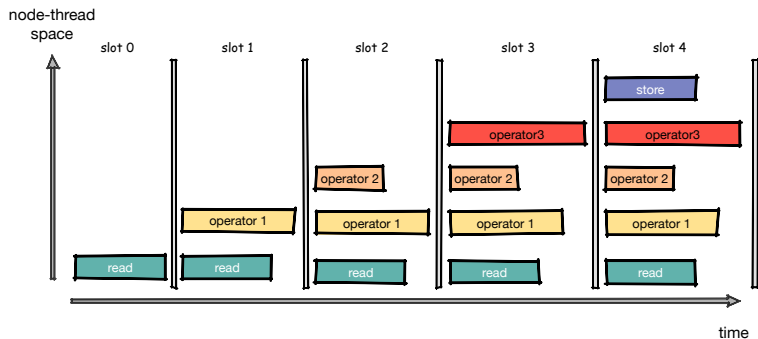
HOW A VECTOR PIPELINING PROCESSING MODEL WORKS



HOW A VECTOR PIPELINING PROCESSING MODEL WORKS



HOW A VECTOR PIPELINING PROCESSING MODEL WORKS



- Parallelization split in two dimensions:
 - POSIX threads for vectors of operators
 - OpenMP in some operators (*not all gain something from parallelization!*)

- Parallelization split in two dimensions:
 - POSIX threads for vectors of operators
 - OpenMP in some operators (*not all gain something from parallelization!*)
- The run time of the evaluation of a *vector of operators* is determined by the operator running longest

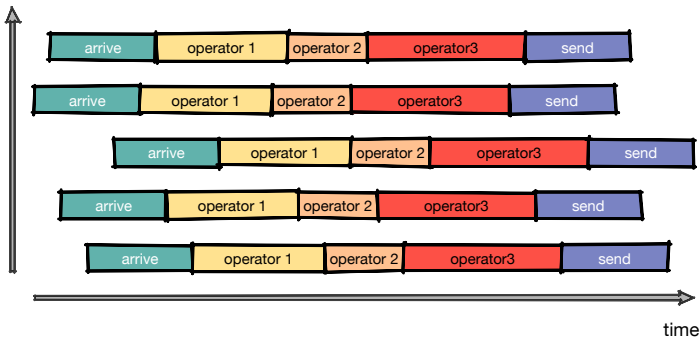
- Parallelization split in two dimensions:
 - POSIX threads for vectors of operators
 - OpenMP in some operators (*not all gain something from parallelization!*)
- The run time of the evaluation of a *vector of operators* is determined by the operator running longest
- Limited to a node

THE STREAMING MODEL

1. Stream processing
2. Distribute work
3. Continuous computation

HOW A VECTOR PIPELINING PROCESSING MODEL WORKS

node-thread
space



1. Functional description of processing algorithm
2. Direct acyclic graph representation (DAG): can be used for optimization and parallelization
3. Due to streaming much reduced disk load/store activity
requirements are shifted to memory and network
4. netCDF/hdf5 not suitable for streaming
(file systems in a file)
5. grib can be streamed - actually: this was the original primary design goal
(self contained, smallest process-able unit)

CURRENT ACTIVITIES

- Development of a DAG based worker/broker toolkit with arithmetic operators
Hermes, Florian Rathgeber and Tiago Quintino (ECMWF)

- Development of a DAG based worker/broker toolkit with arithmetic operators
Hermes, Florian Rathgeber and Tiago Quintino (ECMWF)
- Refactorization of cdo by moving to C++ and disentangling command line and operator handling

- Development of a DAG based worker/broker toolkit with arithmetic operators
Hermes, Florian Rathgeber and Tiago Quintino (ECMWF)
- Refactorization of cdo by moving to C++ and disentangling command line and operator handling
- Develop an evaluation hierarchy for cdo operators

FUTURE

- Combine Hermes and cdo,

- Combine Hermes and cdo,
- Get a working prototype, and finally

- Combine Hermes and cdo,
- Get a working prototype, and finally
- Optimize DAG based distributed post-processing