

# Towards Performance Portability for Atmospheric and Climate Models with the GGDML DSL

Nabeeh Jum'ah, Julian Kunkel, Günther Zängl,  
Hisashi Yashiro, Thomas Dubos, Yann Meurdesoif

Scientific Computing  
Department of Informatics  
University of Hamburg

ISC HPC Research Poster Presentation  
2017-06-20



# Towards Higher-Level Code Design

## Goals

- Improve code quality and scientists productivity
  - Enhance source repositories maintainability
  - Reduce complexity in optimized-code development
- Provide better performance-portability of code

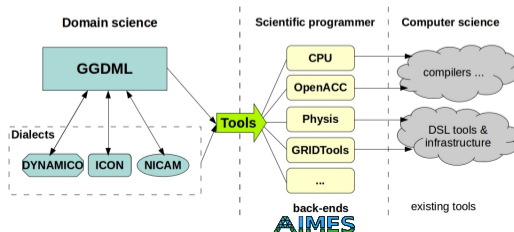
## Constraints

- The existing codebases should be preserved
- Tools should be lightweight, flexible, and easily maintainable

# Improving Code Quality and Performance-Portability

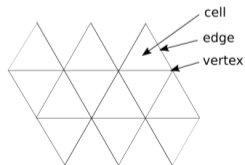
## Strategy

- Foster separation of concerns
  - Domain scientists develop domain logic in source code
  - Scientific programmers write hardware configurations
- Allow coding domain logic with a Domain-Specific Language
  - Extending an existing GPL with domain science concepts
  - Free of any lower level (e.g., architecture) details
- Provide the tools to implement S2S translation
  - Guided by configurations provided by scientific programmers

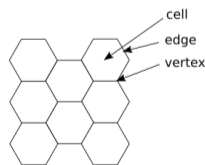


# GGDML: Our Developed DSL

- **GGDML**: *General grid definition and manipulation language*
- Development: Co-design in collaboration with domain scientists
- Features
  - Hides memory access details
  - Abstracts higher concepts of grids, hiding connectivity details
- Constructs for the abstraction of grids
  - Grid definition
  - Grid-bound variable declaration
  - Grid-bound variable access/update
  - Stencil operations



a) Triangular grid



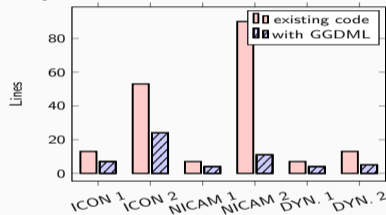
b) Hexagonal grid

# GGDML Impact on Code Quality

## Evaluation

- We estimated changes on code size and complexity

Model, kernel	lines (LOC)		words		characters	
	before DSL	with DSL	before DSL	with DSL	before DSL	with DSL
ICON 1	13	7	238	174	317	258
ICON 2	53	24	163	83	2002	916
NICAM 1	7	4	40	27	76	86
NICAM 2	90	11	344	53	1487	363
DYNAMICO 1	7	4	96	73	137	150
DYNAMICO 2	13	5	30	20	402	218
total	183	55	911	430	4421	1991
percentage	30.05%		47.20%		45.04%	



- We investigated potential cost savings using COCOMO

Software project	DSL?	Effort Applied	Dev. Time (months)	People require	dev. costs (M€)
Semi-detached	without	2462	38.5	64	12.3
	with	1133	29.3	39	5.7
Organic	without	1295	38.1	34	6.5
	with	625	28.9	22	3.1