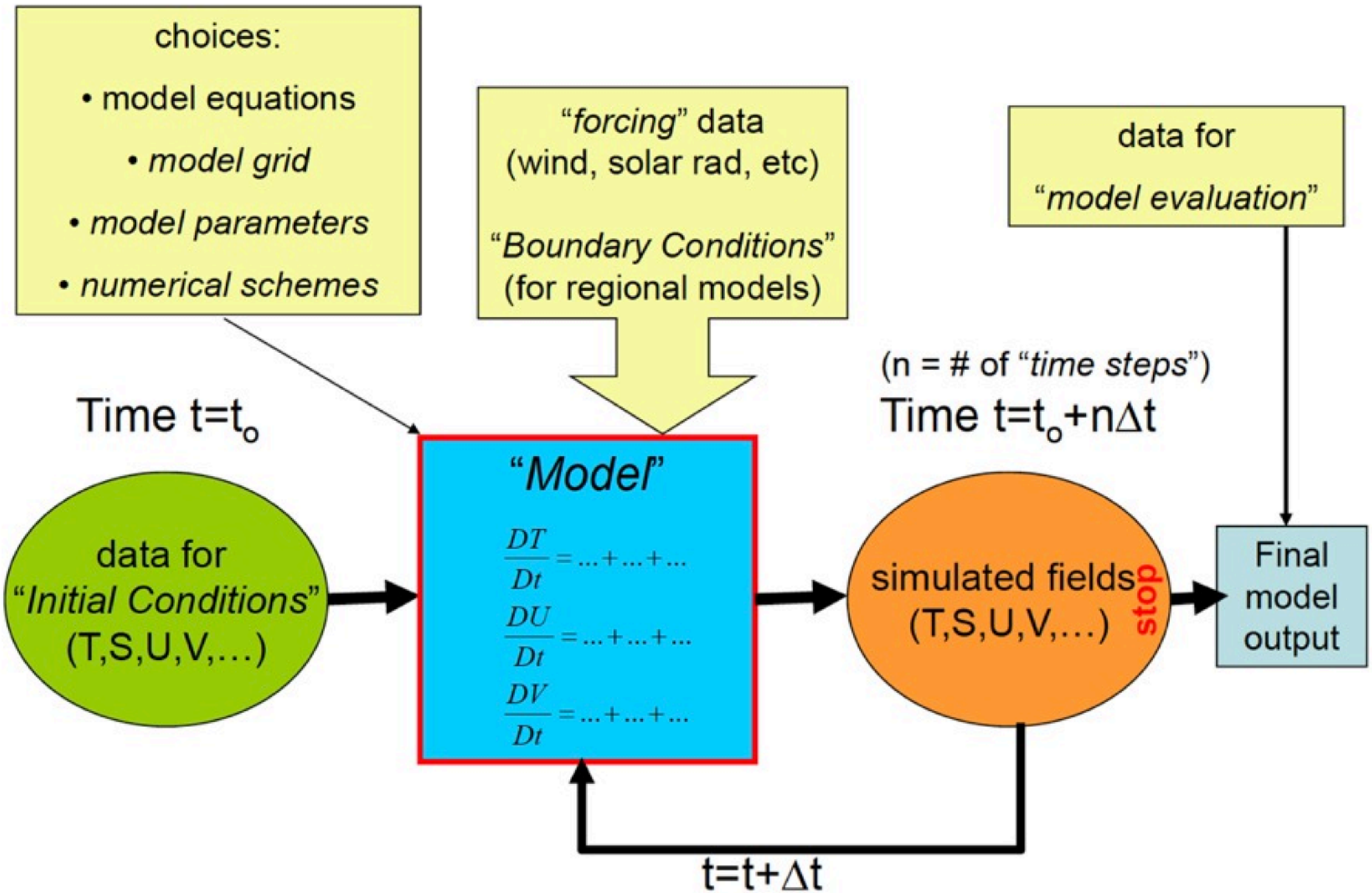
高性能计算与海洋环流模式

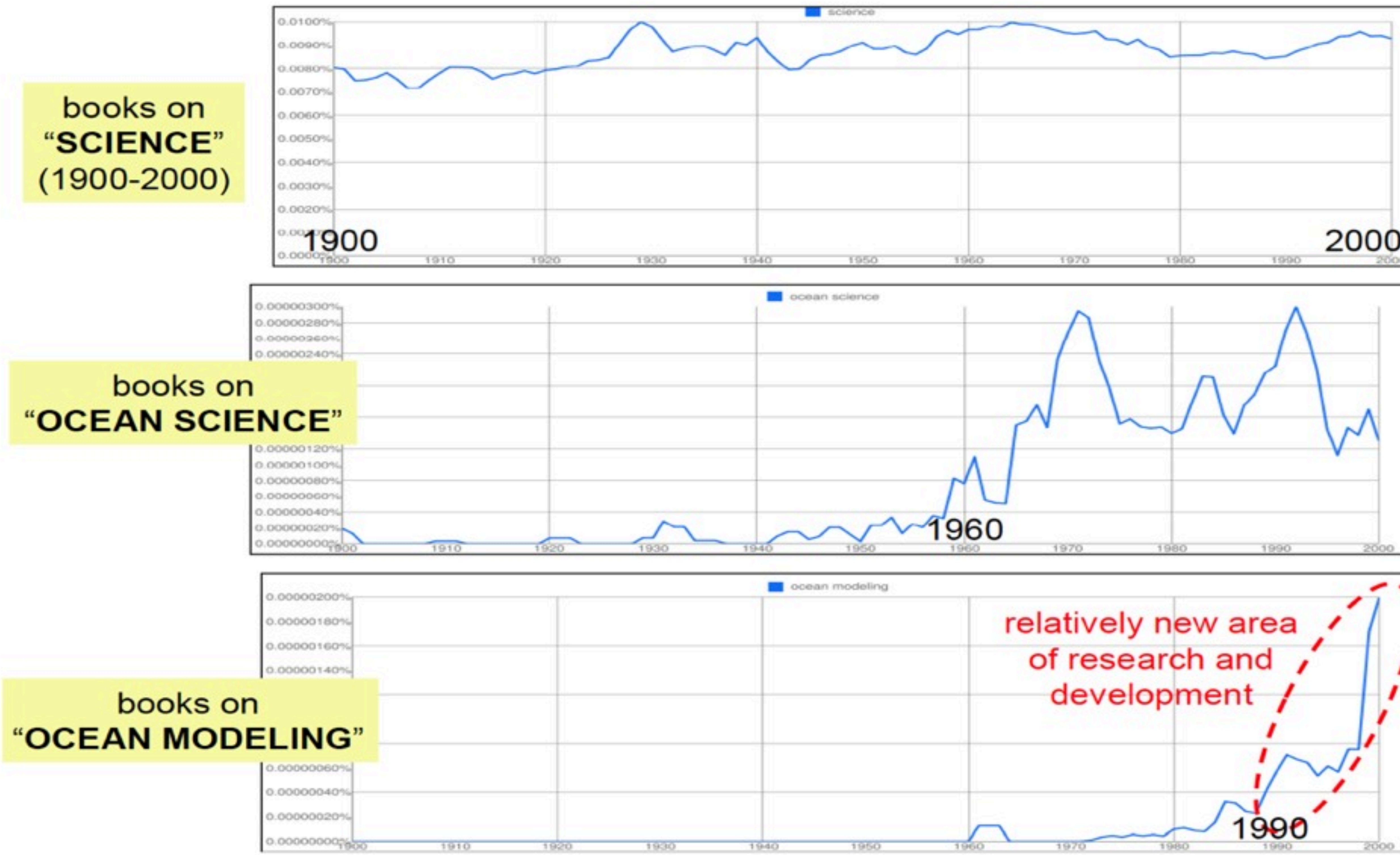
黄小猛
清华大学地球系统科学研究中心
hxm@tsinghua.edu.cn

What is a numerical ocean model?



13年3月25日星期一

Historical perspective and the interest in ocean modeling

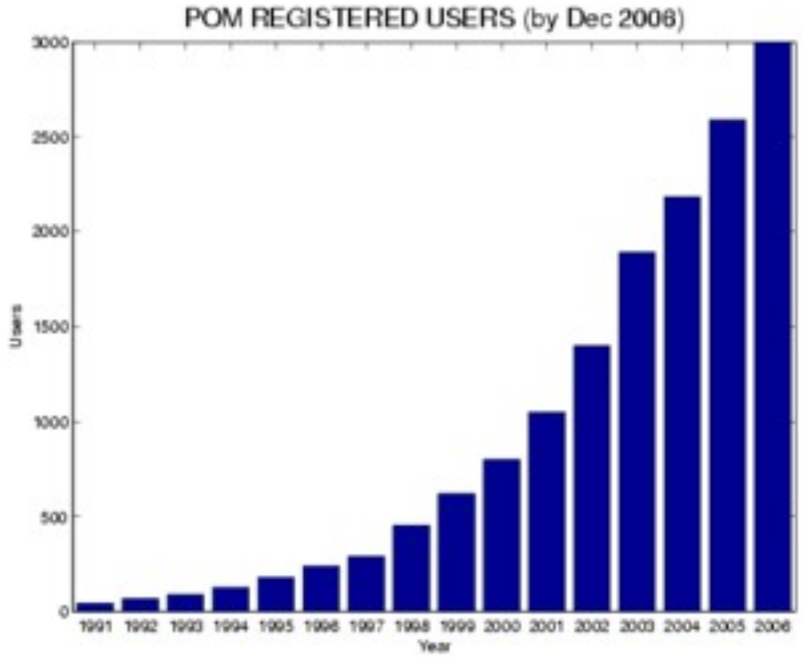


Google labs Books Ngram Viewer

What is the purpose of ocean modeling?

- process studies (examples: simple, more complex, realistic)
- environmental monitoring, impact studies(examples: how would sea-level rise impact coastal flooding?, how would building a dam impact river plume?, oil spill prediction, etc.)
- forecast systems
  - nowcast/hindcast (study past or present events)
  - real-time short-term ocean forecast (days-weeks)
  - long-term prediction (seasonal to future climate)
- coupling with other models:
  - coupled ocean-atmosphere-land models
  - coupled physical-biological or ecosystem models
  - couple with oil/pollution spill models

Why is this growing interest in numerical ocean modeling and prediction?

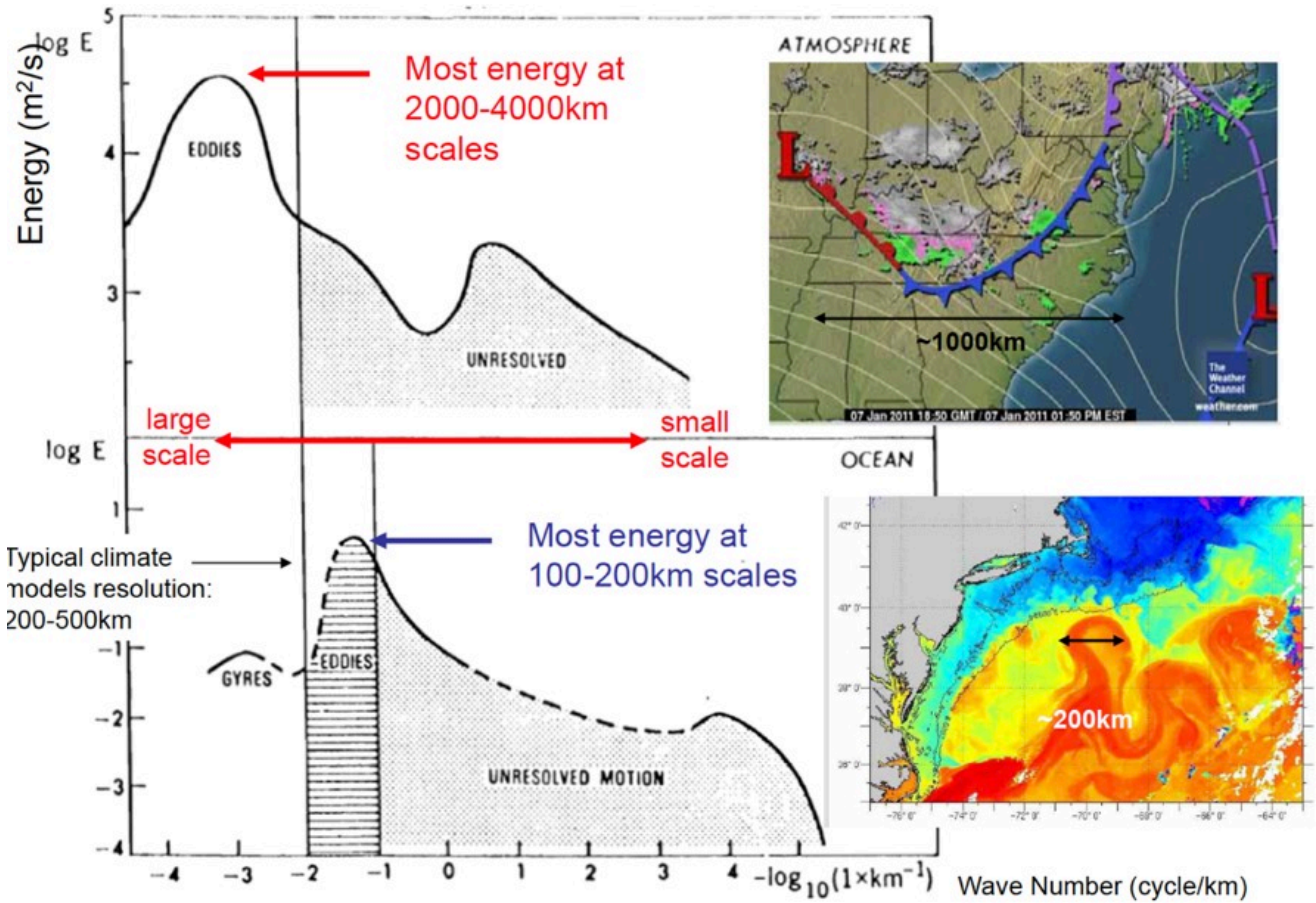


- Exponential growth in computers' speed and memory (supercomputers& PCs): Moore's Law: doubling every 2y
  - Allowing more realistic models with higher resolution
- Advances in satellite observations of the global ocean
  - Allowing data assimilation, evaluation and prediction

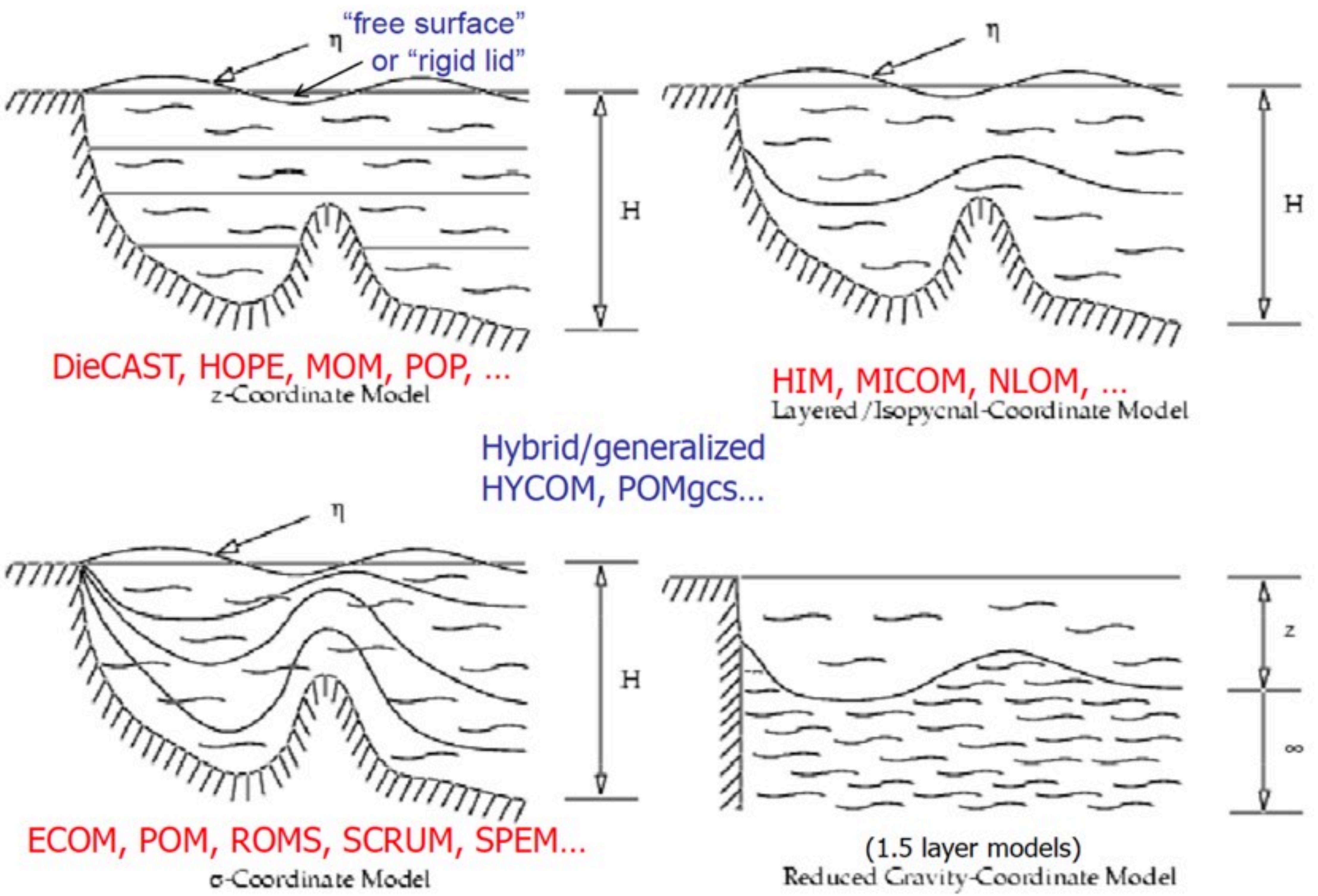
Ocean v.s. Atmospheric

- Note that advances in ocean prediction systems is decades behind atmospheric weather forecasting:
  - Numerical weather forecasting started in the 1950s (Charney, von Neumann and others)
  - Ocean forecasting only in the 1980s-1990s!
- Why?
  - public demand?
  - oceanic scales
  - data availability
  - data assimilation methodologies
  - insufficient computational capabilities

Kinetic Energy Spectrum for the Atmosphere and the Ocean (Woods, 1985; Bryan, 1990)



Class of ocean models(vertical grid)



So many ocean models

- There are many different ocean models now freely available:
  - we do not need to write the computer code from scratch
- but…
  - we need to know the differences between the models
  - we need to understand how ocean models work and what type of model may be better for particular problem

Some Questions

- What is the advantages or disadvantages of each model?
- Are differences in the models make any difference in the results? (there is only one "real ocean"…)
- What model should we choose for our own application?
  - a single model may not provide all the answers
  - model-data or model-model comparisons are useful evaluation methods

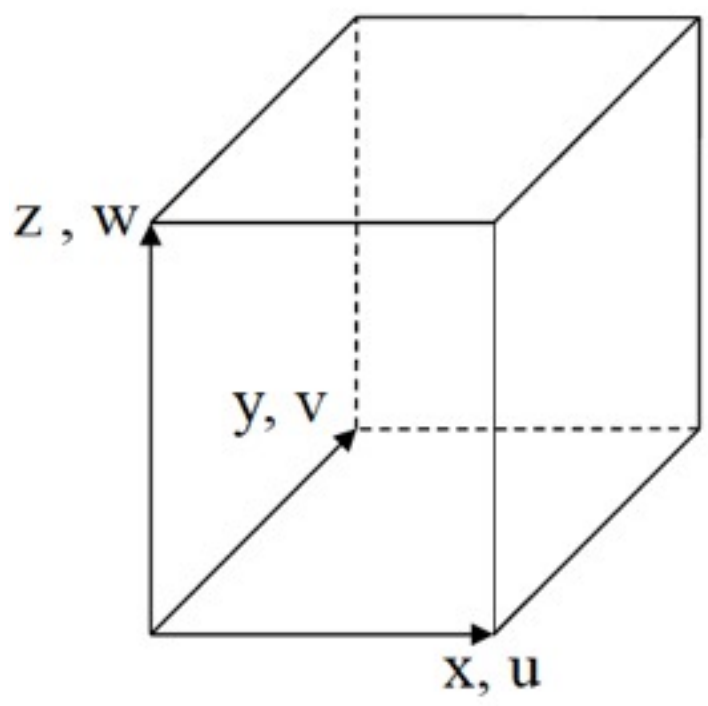
Review of Physical Oceanography Properties and Definitions (with special emphasis on those used in ocean modeling)

- Note that this talk is focused on modeling of the physical properties of the ocean (other ocean models may also include aspects of biology, chemistry, ecology, etc.). Therefore, we are mainly interested in predicting:

Temperature, T (^oC, $^oK=273+^oC$)
(models often use potential temperature, **θ**, instead of in-situ, T)

Salinity, S (‰, ppt, psu)
(total amount of dissolved material in grams in one kilogram of sea water)

Velocity components u, v, w or V (vector) (m/s)



Cartesian coordinates (x, y, z)

Primitive Equations Ocean Models

- What equations numerical ocean models need to solve?
- How do models solve these equations?
- What assumptions are made in each model and how they may affect the results?

Physical Oceanography Equations

- Conservation of momentum for solving velocity (u,v,w)
  - (based on Newton's 2nd law: F=ma)
- Conservation of heat for solving temperature (T)
- Conservation of salt for solving salinity (S)
- Conservation of mass for solving density (ρ)

Eventually the goal is to find the state of the ocean,
i.e., to solve the 3-dimensional (3D) time-dependent fields:
u(x,y,z,t), v(x,y,z,t), w(x,y,z,t), T(x,y,z,t), S(x,y,z,t), ρ(x,y,z,t)

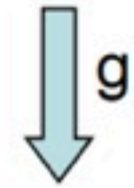
So what are the main forces acting on a parcel of water in the ocean?

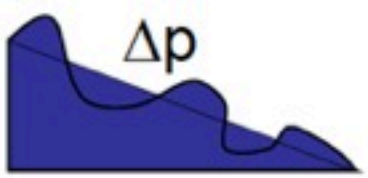
$$\frac{du}{dt} = \frac{1}{\rho}\sum F_x$$

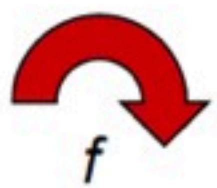
$$\frac{dv}{dt} = \frac{1}{\rho}\sum F_y$$

$$\frac{dw}{dt} = \frac{1}{\rho}\sum F_z$$

Force	Important for
• **gravity-**	buoyancy currents, waves, tides
• **pressure gradient-**	geostrophic currents
• **friction-**	wind-driven currents, bottom boundary layers
• **Coriolis-** (rotation)	ocean circulation, inertial oscillations







The Equations of Motion

Local acceleration | Advection (non-linear) | pressure gradient | Coriolis | Friction (viscosity/mixing) terms: Horizontal, vertical

u & v -equation

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} + w\frac{\partial u}{\partial z} = -\frac{1}{\rho}\frac{\partial p}{\partial x} + fv + \frac{\partial}{\partial x}\left[A_x\frac{\partial u}{\partial x}\right] + \frac{\partial}{\partial y}\left[A_y\frac{\partial u}{\partial y}\right] + \frac{\partial}{\partial z}\left[A_z\frac{\partial u}{\partial z}\right]$$

$$\frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} + w\frac{\partial v}{\partial z} = -\frac{1}{\rho}\frac{\partial p}{\partial y} - fu + \frac{\partial}{\partial x}\left[A_x\frac{\partial u}{\partial x}\right] + \frac{\partial}{\partial y}\left[A_y\frac{\partial u}{\partial y}\right] + \frac{\partial}{\partial z}\left[A_z\frac{\partial u}{\partial z}\right]$$

w-equation → **Hydrostatic balance**

$$\frac{\partial w}{\partial t} + u\frac{\partial w}{\partial x} + v\frac{\partial w}{\partial y} + w\frac{\partial w}{\partial z} = -\frac{1}{\rho}\frac{\partial p}{\partial z} \overbrace{- g}^{gravity} + \frac{\partial}{\partial x}\left[A_x\frac{\partial w}{\partial x}\right] + \frac{\partial}{\partial y}\left[A_y\frac{\partial w}{\partial y}\right] + \frac{\partial}{\partial z}\left[A_z\frac{\partial w}{\partial z}\right]$$

density eq. (conservation of mass) → **The Continuity Equation**

$$\frac{1}{\rho}\left(\frac{\partial \rho}{\partial t} + u\frac{\partial \rho}{\partial x} + v\frac{\partial \rho}{\partial y} + w\frac{\partial \rho}{\partial z}\right) + \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = 0$$

Incompressible/Boussinesq assumption

Conservation of heat and salt: The Temperature and Salinity Equations

$$\overbrace{\frac{\partial T}{\partial t}}^{\text{Local change}} + \overbrace{u\frac{\partial T}{\partial x} + v\frac{\partial T}{\partial y} + w\frac{\partial T}{\partial z}}^{\text{advection}} = \overbrace{\frac{\partial}{\partial x}\left[K_x\frac{\partial T}{\partial x}\right] + \frac{\partial}{\partial y}\left[K_y\frac{\partial T}{\partial y}\right]}^{\text{Horizontal mixing (diffusion)}} + \overbrace{\frac{\partial}{\partial z}\left[K_z\frac{\partial T}{\partial z}\right]}^{\text{Vertical mixing (diffusion)}} + \overbrace{Q_T}^{\text{Source/sink terms}}$$

$$\frac{\partial S}{\partial t} + u\frac{\partial S}{\partial x} + v\frac{\partial S}{\partial y} + w\frac{\partial S}{\partial z} = \underbrace{\frac{\partial}{\partial x}\left[K_x\frac{\partial S}{\partial x}\right] + \frac{\partial}{\partial y}\left[K_y\frac{\partial S}{\partial y}\right]}_{} + \underbrace{\frac{\partial}{\partial z}\left[K_z\frac{\partial S}{\partial z}\right]}_{} + Q_S$$

Summary of the Equations

- 7 unknowns: func(x,y,z,t) u, v, w, P, ρ, S, T
- 4 prognostic Equations:
  u, v, T, S

$$\frac{du}{dt} = forces; \quad \frac{dv}{dt} = forces$$

$$\frac{dT}{dt} = heat\ \ terms; \quad \frac{dS}{dt} = salt\ \ terms$$

- 3 diagnostic Equations:
  w, P, ρ

$$\frac{\partial w}{\partial z} = -\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right)$$

$$\frac{\partial p}{\partial z} = -\rho g$$

$$\rho = \rho(T, S, P)$$

We have 7 equations and 7 unknowns so we can solve this set of differential equations. But how?

Models that solve the complete set of equations are called Primitive Equations Models

Momentum equations

$$\frac{\partial u}{\partial t}+u\frac{\partial u}{\partial x}+v\frac{\partial u}{\partial y}+w\frac{\partial u}{\partial z}=-\frac{1}{\rho}\frac{\partial p}{\partial x}+fv+\frac{\partial}{\partial x}\left[A_x\frac{\partial u}{\partial x}\right]+\frac{\partial}{\partial y}\left[A_y\frac{\partial u}{\partial y}\right]+\frac{\partial}{\partial z}\left[A_z\frac{\partial u}{\partial z}\right]$$

$$\frac{\partial v}{\partial t}+u\frac{\partial v}{\partial x}+v\frac{\partial v}{\partial y}+w\frac{\partial v}{\partial z}=-\frac{1}{\rho}\frac{\partial p}{\partial y}-fu+\frac{\partial}{\partial x}\left[A_x\frac{\partial u}{\partial x}\right]+\frac{\partial}{\partial y}\left[A_y\frac{\partial u}{\partial y}\right]+\frac{\partial}{\partial z}\left[A_z\frac{\partial u}{\partial z}\right]$$

Hydrostatic equation

$$\frac{\partial p}{\partial z}=-\rho g$$

Equation of State

$$\rho=\rho(T,S,P)$$

Continuity equation

$$\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}+\frac{\partial w}{\partial z}=0$$

Heat and salt equations

$$\frac{\partial T}{\partial t}+u\frac{\partial T}{\partial x}+v\frac{\partial T}{\partial y}+w\frac{\partial T}{\partial z}=\frac{\partial}{\partial x}\left[K_x\frac{\partial T}{\partial x}\right]+\frac{\partial}{\partial y}\left[K_y\frac{\partial T}{\partial y}\right]+\frac{\partial}{\partial z}\left[K_z\frac{\partial T}{\partial z}\right]+Q_T$$

$$\frac{\partial S}{\partial t}+u\frac{\partial S}{\partial x}+v\frac{\partial S}{\partial y}+w\frac{\partial S}{\partial z}=\frac{\partial}{\partial x}\left[K_x\frac{\partial S}{\partial x}\right]+\frac{\partial}{\partial y}\left[K_y\frac{\partial S}{\partial y}\right]+\frac{\partial}{\partial z}\left[K_z\frac{\partial S}{\partial z}\right]+Q_S$$

May also need additional equations for mixing coefficients A_x, A_y, A_z, K_x, K_y, K_z

Notes:

- **One of the biggest challenges in modeling is how to parameterize or calculate the turbulent mixing coefficients.**

Finite Difference Schemes, Numerical Models and Grids

- The basic idea:
  - The real ocean is continuous (from molecular to global scales)
  - Computers are discrete (and they can not describe every molecule of the ocean…)
  - Thus we have to convert the equations describing the physical laws of the ocean into a discrete form that the computers can understand…

How do Numerical Models Work?

- Choose a numerical grid (i.e., divide the ocean into “boxes”)
- Choose numerical schemes (many choices; stable and efficient)
- Make a computer program (“numerical code”)
- Run the model (given data for initialization, forcing, boundary conditions, etc…)

Basic Finite Difference Schemes
(other methods: finite element, finite volume, spectral, etc. not discussed here)

Rectangular even-size grid

Total cells: (IM x JM x KM)

i=1,2,3,…IM, etc.

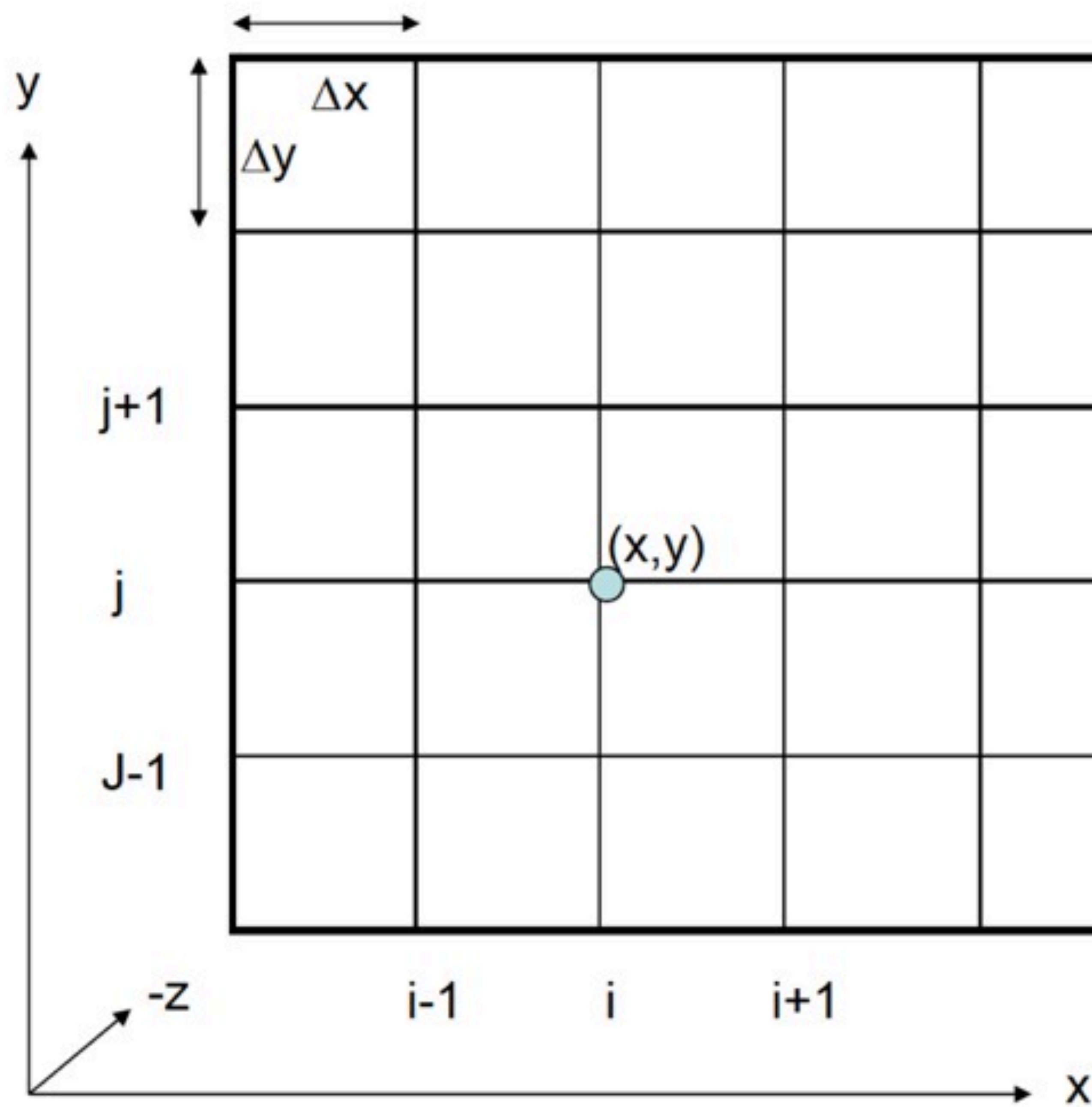
$x=i\Delta x$, $y=j\Delta y$, $z=k\Delta z$, $t=n\Delta t$

Grid size: $\Delta x, \Delta y, \Delta z$

Time step: Δt

- finer grid: more accurate solution, but requires smaller time step to solve and more computations

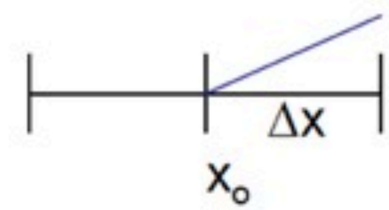
- remember that the equations of motion are solved in every grid cell at every time step!



Note: in non Cartesian grids need to track Lon/Lat of each cell

Different ways to represent the derivatives

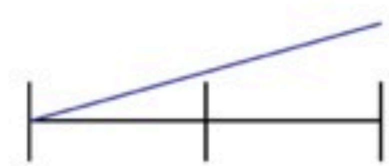
1. Forward, first order accurate



$$\frac{\partial f(x_o)}{\partial x} = \frac{f(x_o + \Delta x) - f(x_o)}{\Delta x} + O(\Delta x f_{xx})$$

2. Backward, first order accurate

$$\frac{\partial f(x_o)}{\partial x} = \frac{f(x_o) - f(x_o - \Delta x)}{\Delta x} + O(\Delta x f_{xx})$$



3. (1)+(2) → Forward, second order accurate

$$\frac{\partial f(x_o)}{\partial x} = \frac{f(x_o + \Delta x) - f(x_o - \Delta x)}{2\Delta x} + O(\Delta x^2 f_{xxx})$$

What do we need from a good numerical scheme?

- Convergence: computational error->0 when Δx,Δt->0
  - meaning: when using finer grid and smaller time step, the numerical error is getting smaller (i.e., more realistic simulation)
- Stability: finite solution, C ≠ ∞ when n->∞
  - meaning: when running a model for very long time, the numerical error does not grow (i.e., model does not “blow up”)
- Schemes can be:
  - unconditionally stable (best, but rare)
  - conditionally stable (for particular choice of parameters)
  - unconditionally unstable (usually a bad choice)

Example of conditions to make a scheme stable

The advection equation in finite difference form:

$$\frac{C_i^{n+1} - C_i^{n-1}}{2\Delta t} + u_i^n \frac{C_{i+1}^n - C_{i-1}^n}{2\Delta x} = 0$$

Can be written as:

$$\overbrace{C_i^{n+1}}^{\text{field in the "future"}} = \overbrace{C_i^{n-1} - \underbrace{u_i^n \frac{\Delta t}{\Delta x}}_{\alpha} \left(C_{i+1}^n - C_{i-1}^n\right)}^{\text{depends on fields in the "past" and "present"}}$$

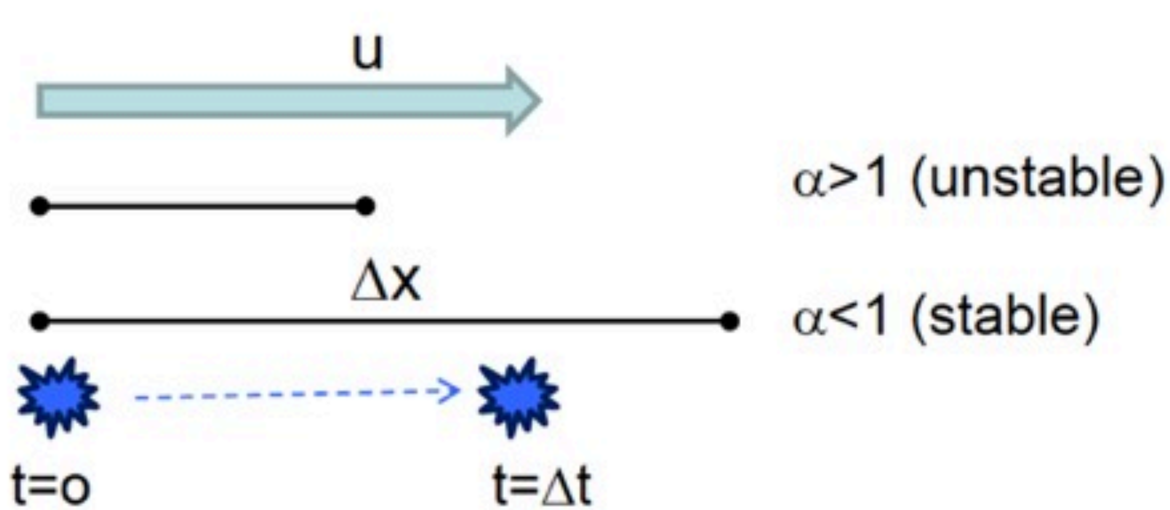
The condition needed for the scheme to be stable:

$\alpha < 1$

$$\alpha = u \frac{\Delta t}{\Delta x} \quad \text{is called the "Courant Number"}$$

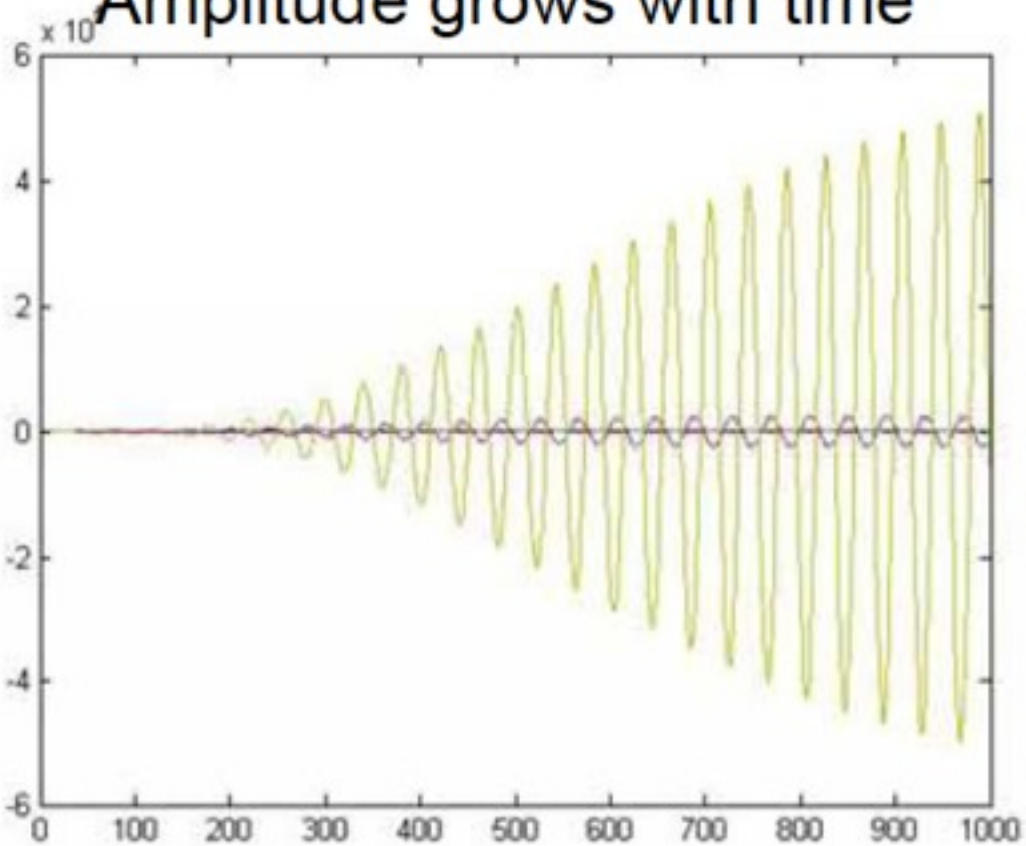
Thus, $(\Delta x / \Delta t) > u$

Smaller grid needs smaller time step for the model to be stable…



How does numerical instability look like?

Amplitude grows with time

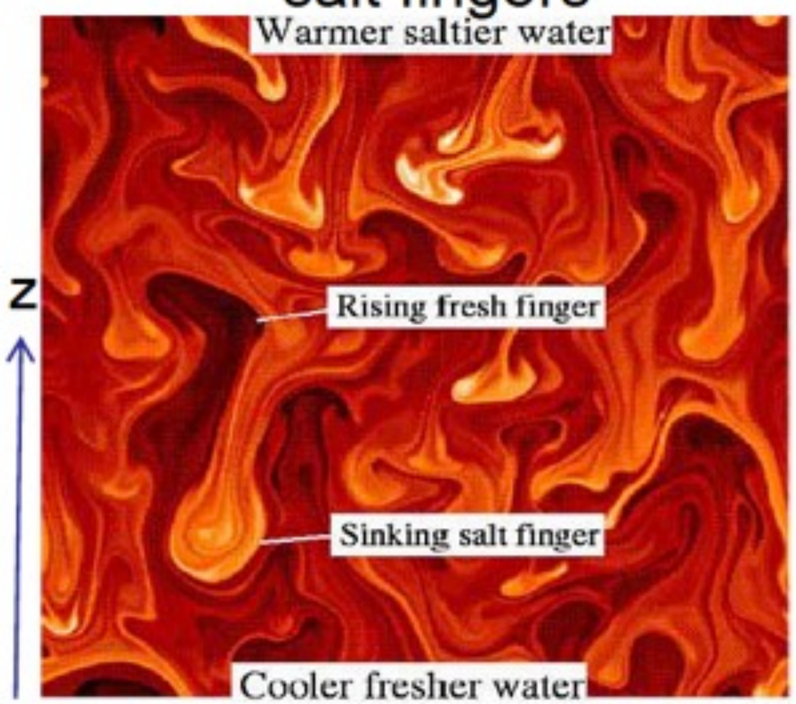


time

… compared with real ocean instability:

“salt fingers”

z



Computational Efficiency

- We usually want a model with as fine resolution as possible, but the size of the grid is often affected by the computational resources
- Time step constraints:
  CFL (Courant-Friedrichs-Lewy) criteria: $\Delta t < \Delta x / C$, $C = (gH)^{1/2}$ (in 1D) time step function of grid size and water depth (i.e., time step < time it takes gravity wave/current to move between two model grid points)

CFL in 2D:

$$\Delta t \leq \frac{1}{C_t}\left(\frac{1}{\Delta x^2} + \frac{1}{\Delta y^2}\right)^{-1/2}$$

where

$$C_t = 2(gH)^{1/2} + \overline{U}_{max}$$

Doubling resolution in a model → 2x2=4 times more grid pts
2 times smaller timestep → 8 times more calculations! (at least)

Some solutions for ocean model

- Oceans are deep, so gravity waves propagate very fast, $C=(gH)^{½}$ (for H=5000m, C~200m/s) requiring small time step: computational difficulties (climate models running for 100s of years will need a time step of a few minutes…)
    - Rigid Lid approximation:
        - Eliminates free-surface waves (i.e., no tides, tsunamis, etc.) and allows longer time step
    - Time Splitting techniques:
        - Separate the fast moving (barotropic, 2D) waves from the slower (baroclinic, 3D) motions
        - Baroclinic (internal waves) $C=(g'H)^{½}$ ~1-10m/s $g'=g(\delta\rho/\rho)$

Time Splitting: solve 2 sets of equations

Vertically integrated 2D equations

(Barotropic/ "External mode")

Solve for surface elevation η(x,y,t) and vertically averaged velocity

$\bar{u}(x,y,t)$, $\bar{v}(x,y,t)$

D=H+η is total water depth

$$\begin{cases} \frac{\partial \eta}{\partial t} + \frac{\partial \bar{u} D}{\partial x} + \frac{\partial \bar{v} D}{\partial y} = 0 \\ \frac{\partial \bar{u} D}{\partial t} - f\bar{v}D + gD\frac{\partial \eta}{\partial x} = F_x \\ \frac{\partial \bar{v} D}{\partial t} + f\bar{u}D + gD\frac{\partial \eta}{\partial y} = F_y \end{cases}$$

3D equations
(Baroclinic/ "interrnal mode")

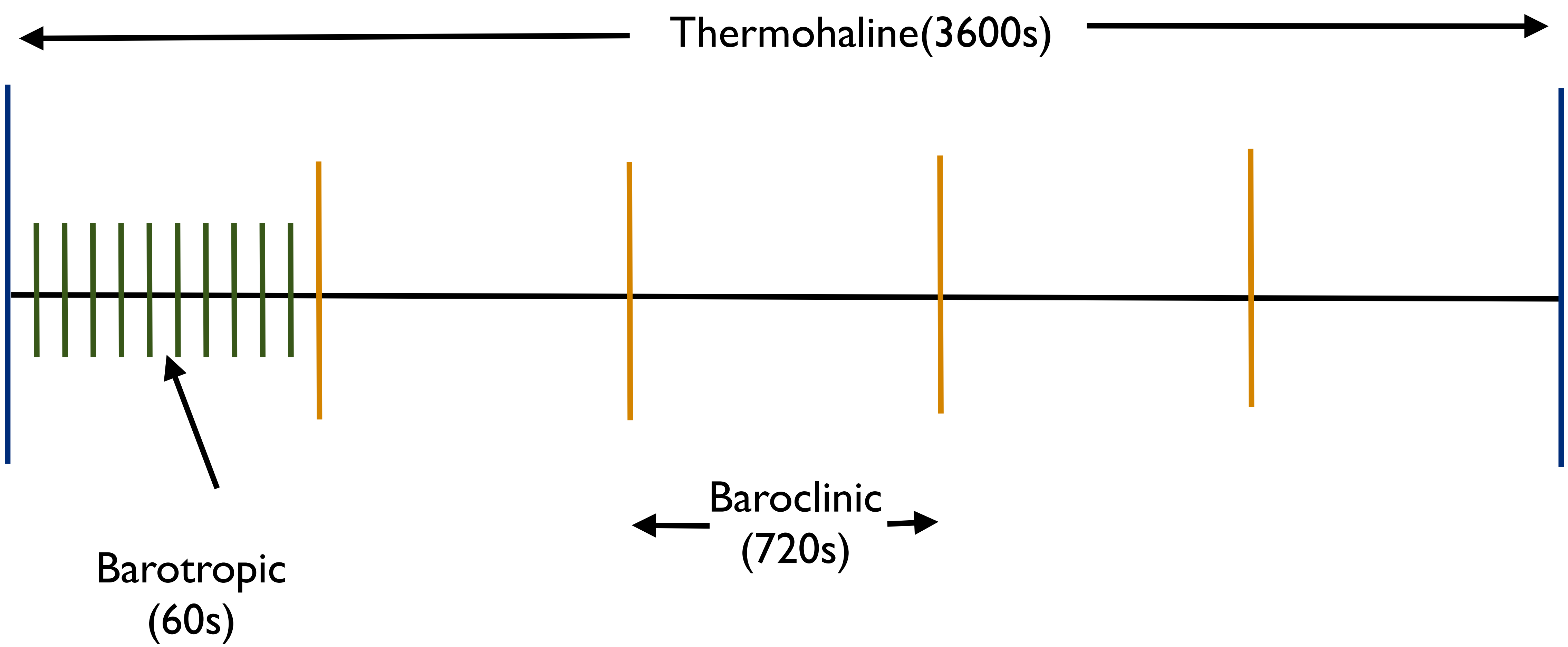
Solve for u(x,y,z,t), v(x,y,z,t), w(x,y,z,t)

$$\begin{cases} \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = 0 \\ \frac{\partial u}{\partial t} + F(u,v) = g\frac{\partial \eta}{\partial x} \\ \frac{\partial v}{\partial t} + F(u,v) = g\frac{\partial \eta}{\partial y} \end{cases}$$

Note: need to match the two solutions!

There are various ways to couple the two modes

Time splitting algorithm



Summary of scales resolved in different ocean models and typical mixing used

computational cost increases exponentially ↓

scales	model type	resolved (unresolved)	sub-grid mixing
100-1000s km	non-eddy resolving models (example: climate)	large-scale circulation (meso-scale eddies & smaller)	constant K, K-profile **(KPP)**, others
5-20 km	eddy resolving models (ex: Gulf Stream)	meso-scale eddies (small eddies and turbulence)	Reynolds-Averaged Navier-Stokes **(RAN)** models: Mellor-Yamada (MY), KPP, other
10cm-10m	Large Eddy Simulation **(LES)** (ex: wave-induced turbulence)	eddies important for turbulence (filter smallest eddies)	Sub-Filter Scale **(SFS)** or Sub-Grid Scale **(SGS)** models
1mm-1m	Direct Numerical Simulation **(DNS)**	smallest scales of fluid dynamics and complete turbulence	None

Great challenges for ocean modeling

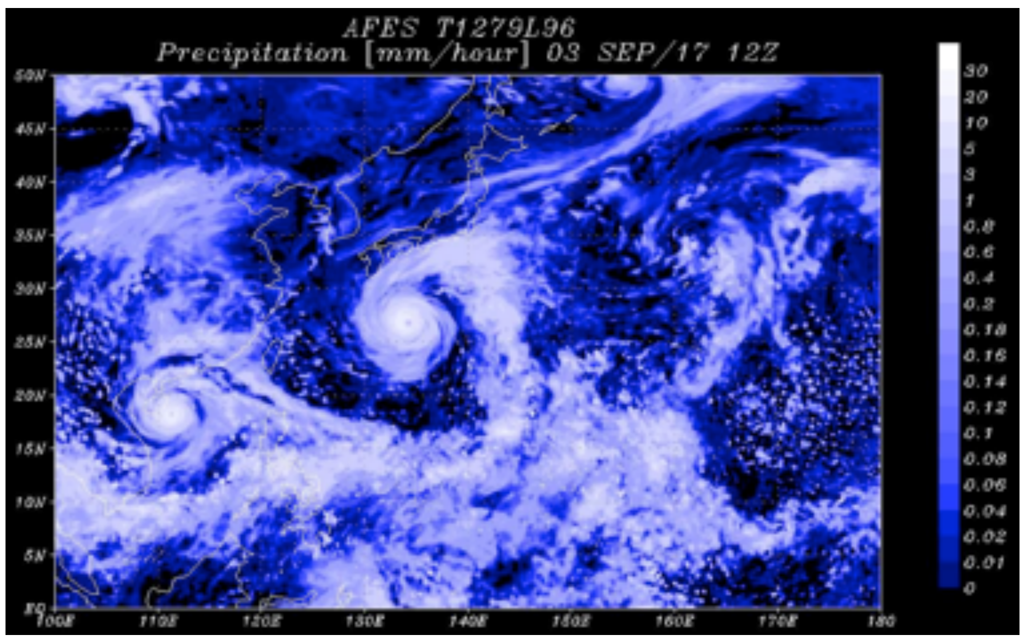
- Challenge 1: Internal Wave Mixing
  - Needs global or basin-scale ocean model (physics only) of order 1 km resolution, run for periods of order 30-40 years.
- Challenge 2: Global Carbon Cycle
  - Needs global physics and biology ocean model, at highest achievable resolution (1/4 or 1/12°) run for periods of order 1,000 years or more.
- Challenge 3: Climate Uncertainty
  - Needs coupled climate models at best possible resolution (e.g. 1 or 0.5°) run for periods of order 100-1000 years, but up to 1,000 times (or more).

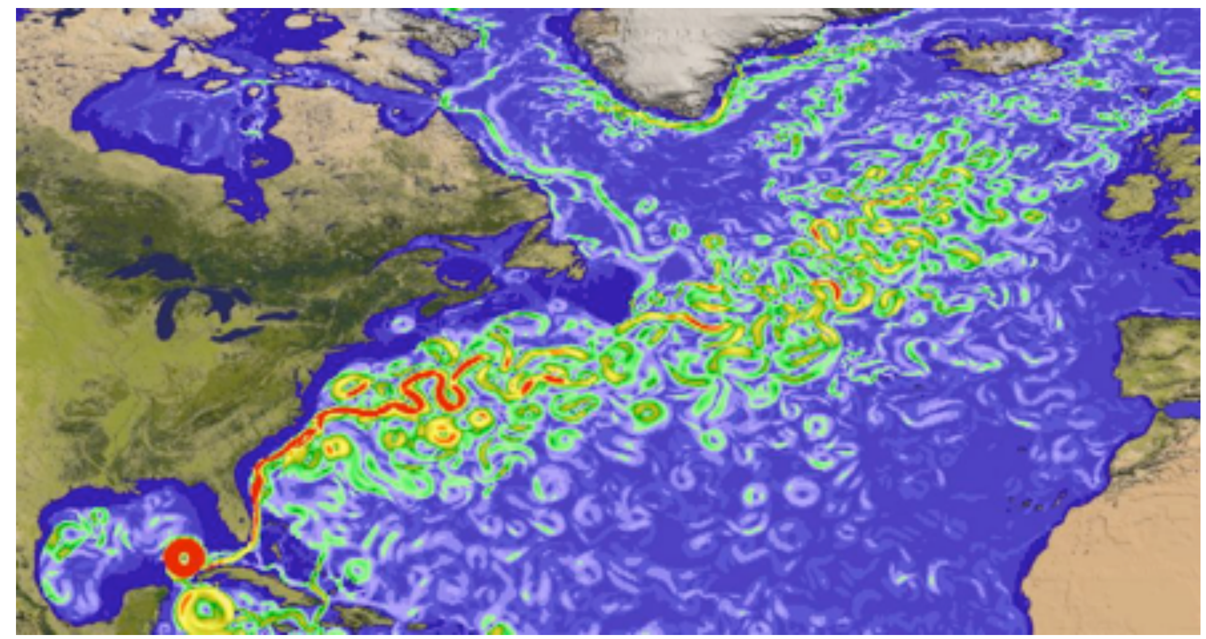
模式的发展趋势

- 趋势之一：越来越多的地球系统分量加入到模式中

模式的发展趋势

- 趋势之二：走向“高时空分辨率”





模式分辨率的提高

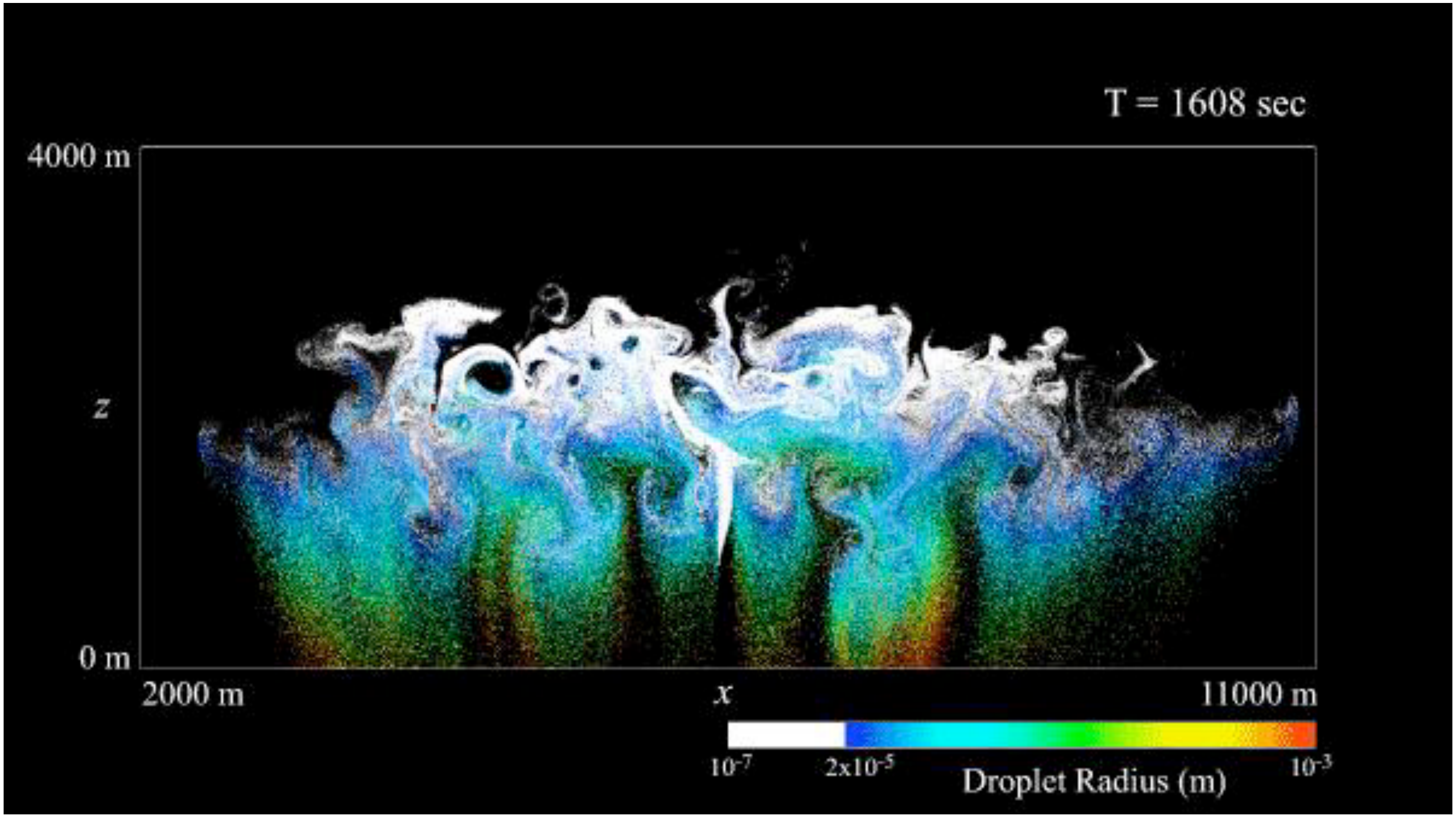
当前“100公里”的全球模拟

提升到“10公里”的全球模拟

提升到“<5公里”的全球模拟

模式的发展趋势

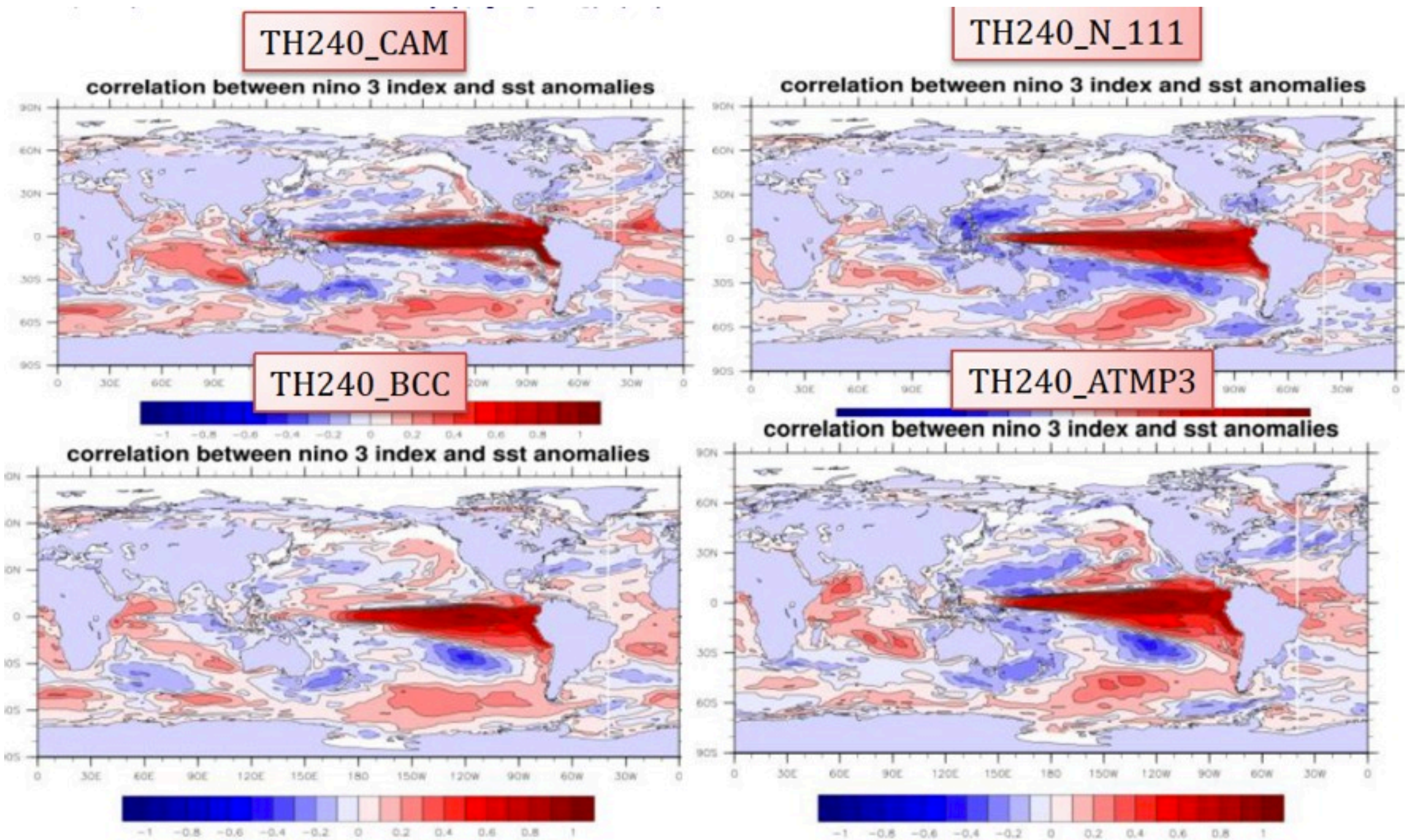
- 趋势之三：描述更多的精细过程



例：云－辐射相互作用的计算最重要的部分之一

模式的发展趋势

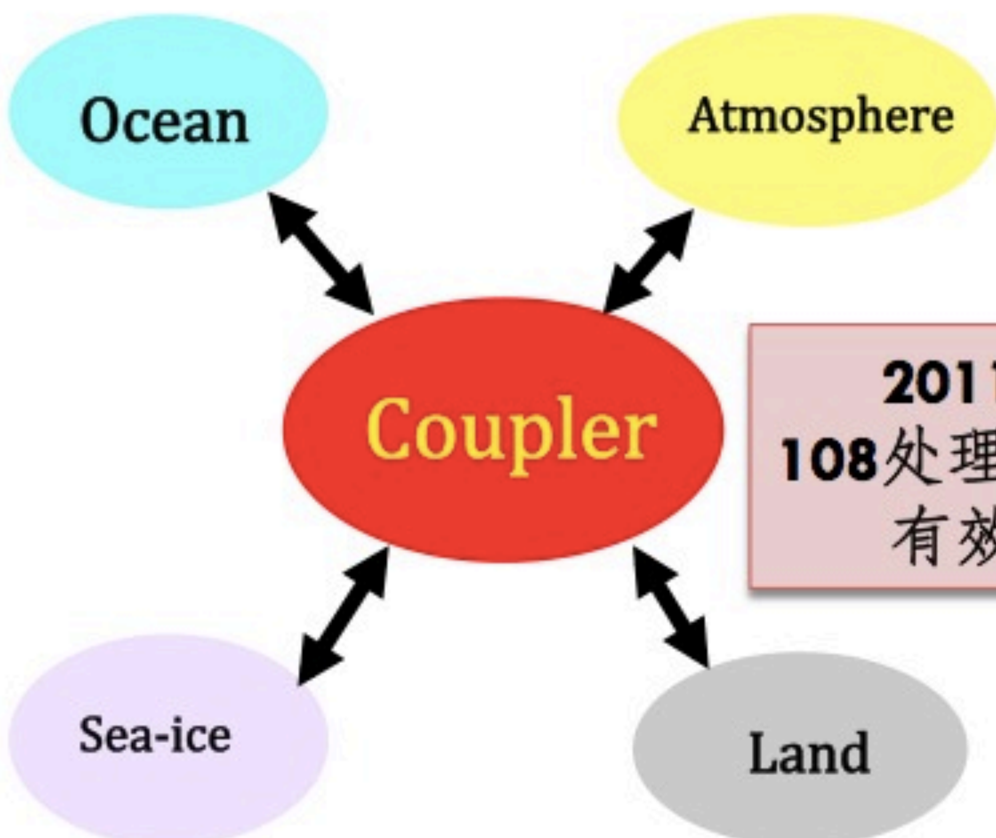
- 趋势之四：超级集合模拟



模式研究对高性能计算能力的需求

- IPCC AR5 计算量巨大
  - 完成年代际预测的试验至少需运行2140模式年
  - 完成气候变化试验至少需运行5365模式年
  - 完成整个IPCC AR5所要求的评估试验，整个模式系统需积分7500个模式年以上

FGOALS_g2.0



2011.4-2011.9 IPCC试验
108处理器核，每天积分约20年
有效数据总量超过40TB

清华探索100(740N,104TF,1PB)

模式研究对高性能计算能力的需求

- 提高分辨率对计算能力的一种形象说明
  - “100公里”的全球模拟－－>十万亿次（10TFlops）
  - “10公里”的全球模拟－－>万万亿次（10PFlops）
  - “5公里”的全球模拟－－>十万万亿次（100PFlops）
- 超级集合预报可利用任意大的计算能力

2011世界高性能计算Top 10

Rank	Site	Computer	Country	Total Cores	Rmax	E (%)	Power	Mflops/Watt
1	RIKEN AICS	K computer, Fujitsu, SPARC64 VIIIfx + custom	Japan	705024	10510000	93.17	12659.89	830.18
2	National Supercomputing Center in Tianjin	Tianhe-1A, NUDT, Intel + NVIDIA GPU + custom	China	186368	2566000	54.58	4040	635.15
3	DOE/SC/Oak Ridge Nat Lab	Jaguar, Cray AMD + custom	USA	224162	1759000	75.46	6950	253.09
4	National Supercomputing Centre in Shenzhen (NSCS)	Nebulea, Dawning Intel + NVIDIA GPU + IB	China	120640	1271000	42.59	2580	492.64
5	GSIC Center, Tokyo Institute of Technology	Tusbame 2.0, HP Intel + Nvidia GPU + IB	Japan	73278	1192000	52.11	1398.61	852.27
6	DOE/NNSA/LANL/SNL	Cielo, Cray AMD + Custom	USA	142272	1110000	81.27	3980	278.89
7	NASA/Ames Research Center/NAS	Plelades, SGI Altix ICE 8200EX/ 8400EX + IB	USA	111104	1088000	82.72	4102	265.24
8	DOE/SC/LBNL/NERSC	Hopper, Cray AMD + Custom	USA	153408	1054000	81.79	2910	362.2
9	Commissariat a l'Energie Atomique (CEA)	Tera-10, Bull Intel + IB	France	138368	1050000	83.7	4590	228.76
10	DOE/NNSA/LANL	Roadrunner, IBM AMD + PowerXCell + IB	USA	122400	1042000	75.74	2345	444.35

2012世界高性能计算Top 10

Rank	Site	System	Cores	Rmax (TFlop/s)	Rpeak (TFlop/s)	Power (kW)
1	DOE/SC/Oak Ridge National Laboratory United States	**Titan** - Cray XK7 , Opteron 6274 16C 2.200GHz, Cray Gemini interconnect, NVIDIA K20x Cray Inc.	560640	17590.0	27112.5	8209
2	DOE/NNSA/LLNL United States	**Sequoia** - BlueGene/Q, Power BQC 16C 1.60 GHz, Custom IBM	1572864	16324.8	20132.7	7890
3	RIKEN Advanced Institute for Computational Science (AICS) Japan	K computer, SPARC64 VIIIfx 2.0GHz, Tofu interconnect Fujitsu	705024	10510.0	11280.4	12660
4	DOE/SC/Argonne National Laboratory United States	**Mira** - BlueGene/Q, Power BQC 16C 1.60GHz, Custom IBM	786432	8162.4	10066.3	3945
5	Forschungszentrum Juelich (FZJ) Germany	**JUQUEEN** - BlueGene/Q, Power BQC 16C 1.600GHz, Custom Interconnect IBM	393216	4141.2	5033.2	1970
6	Leibniz Rechenzentrum Germany	**SuperMUC** - iDataPlex DX360M4, Xeon E5-2680 8C 2.70GHz, Infiniband FDR IBM	147456	2897.0	3185.1	3423
7	Texas Advanced Computing Center/Univ. of Texas United States	**Stampede** - PowerEdge C8220, Xeon E5-2680 8C 2.700GHz, Infiniband FDR, Intel Xeon Phi Dell	204900	2660.3	3959.0	
8	National Supercomputing Center in Tianjin China	**Tianhe-1A** - NUDT YH MPP, Xeon X5670 6C 2.93 GHz, NVIDIA 2050 NUDT	186368	2566.0	4701.0	4040
9	CINECA Italy	**Fermi** - BlueGene/Q, Power BQC 16C 1.60GHz, Custom IBM	163840	1725.5	2097.2	822
10	IBM Development Engineering United States	**DARPA Trial Subset** - Power 775, POWER7 8C 3.836GHz, Custom Interconnect IBM	63360	1515.0	1944.4	3576

中国的高性能计算机

- 我国在“十一五”期间，863支持研制成功了天河1A、曙光星云、神威蓝光三台千万亿次高性能计算机，在国内外有重要影响。

天河1A

- 峰值性能：4700TFlops，
- 持续性能：2566TFlops（LINPACK实测值）
- 全系统共有：23552个微处理器，其中14336 个Intel X5670 CPU 、2048个自主FT－1000 CPU 、7168个Nvidea M2050 GPU
- 内存总容量262TB
- 存储总容量2PB
- 满负荷运行最大功耗为4.04MW
- 全系统包含140个机柜
- 占地总面积700平方米
- 总重量160吨
- 环境温度10℃～35℃
- 湿度10％～90％

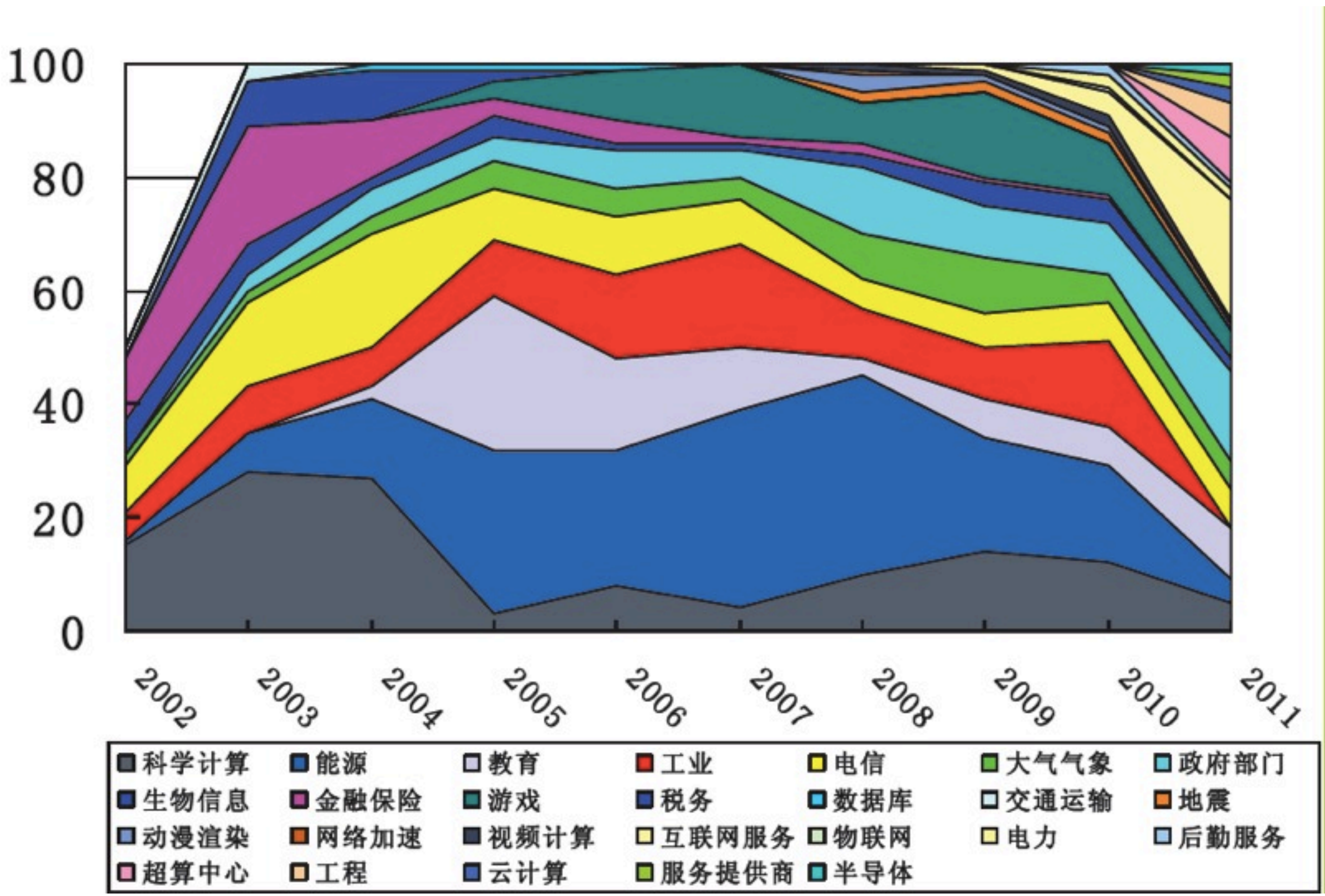
神威蓝光

- 于2011年9月安装于国家超算济南中心，全部采用自主设计生产的CPU(ShenWei processor SW1600)，系统共8704个CPU，峰值1.07016PFlops，持续性能795.9TFlops，Linpack效率74.37%，总功耗1074KW。

中国TOP100制造商分析

	厂商	系统	份额	Rmax [TF/s]	Rpeak [TF/s]	平均效率（%）	处理器核
国产机器	曙光	35	35%	2848.18	4544.56	61.40%	363864
	浪潮	7	7%	306.93	535.39	60.50%	55748
	神威	5	5%	1087.80	1404.71	84.34%	165512
	国防科大	2	2%	3337.70	6044.20	56.00%	256000
	中科院过程所	1	1%	496.50	1138.44	43.60%	33120
	联想	1	1%	102.80	145.29	70.80%	12160
国产小计		51	51%	8204.11	13812.59	62.90%	886404
引进机器	IBM	35	35%	3264.31	6020.59	57.60%	588524
	HP	13	13%	509.51	927.77	57.60%	98056
	Dell	1	1%	23.40	44.93	72.43%	6880
引进小计		49	49%	3797.22	6993.28	57.50%	690900
总计		100	100%	12001.33	20805.87	59.63%	1577304

中国TOP100应用领域趋势



全球变化研究对高性能计算人才的需求

- 地球系统模式的学科交叉性非常强，兼有科学研究、技术攻关和工程建设的特征，不仅涉及到地学的各个分支，还涉及到数学、物理、环境、计算机等众多学科。
- 目前一个很突出的制约我国地球系统模式发展的瓶颈，是严重缺乏学科的交叉，尤其缺乏与高性能计算领域的交叉，需要有众多的高性能计算专业人才了解全球变化，与全球变化的相关人员精诚合作，在支持全球变化的高性能计算技术方面取得重要突破。

我国地球系统模式发展的一些问题

- 1、具有典型MPMD特征的地球系统模式程序过于复杂
  - 海量的观测数据和再分析数据广泛分布且异构，难于处理
  - 程序模块化程度低，开发周期长、难于调试与优化
  - 前处理、后处理工具种类繁多，学习曲线高
- 2、高性能地球系统数值模拟水平较低
  - 我国模式分辨率低，具有自主知识产权与高科学置信度的模式程序远少于发达国家
  - 可扩展性差，现有的国内模式难以同时使用上千处理器核

我国地球系统模式发展的一些问题

- 3、缺乏支持地球系统模式研究与发展的软件支撑框架与系统平台
  - 美国有ESMF、FMS、WRF、ESG
  - 欧盟有PRISM
  - 日本有Earth Simulator
- 4、交叉人才培养不足
  - 系统地针对地球系统模式发展，从事计算方法、并行算法与程序研制的人员少


地球系统模式
开发流程
开发向导与
编辑器
源代码
编译器/
调试器/
优化器
可执行代码
I'm 01100110 01100001 01100101
01101100 01101001 01101110 01100111
00100000 01101100 01110101 01100011
01101011 01111001 00001010
运行时环境
初始场与
边界数据
计算结果
子系统
标准数据集
可视化与
数据比对
分析工具
可视化结果
比对结果
地球系统
模式
算法（并行）


开发向导与
编辑器
源代码
编译器/
调试器/
优化器
可执行代码
初始场与
边界数据
子系统
标准数据集
可视化与
数据比对
分析工具
计算结果
比对结果
可视化结果
地球系统
模式
算法（并行）
地
工欲善其事，必先利其器

“面向地球系统模式研究的高性能计算支撑软件系统”项目总体目标

- 项目目标:
针对我国地球系统模式研究的迫切需要，依托国产高性能计算机，研究支持地球系统模式开发的高性能计算支撑软件关键技术，研制基于“模式模块库、模板库、工具库和插件式平台”（三库一平台）架构的一体化地球系统模式集成开发环境，支持全球变化研究，填补我国在地球系统模式软件支撑平台方面的空白；培养一支具有国际水平的高性能地球系统模式计算研究队伍。

- 总体原则:
致力于减轻地球科学研究者的软件开发负担与高性能计算机的使用难度，提高效率，使他们腾出精力专注于模式研究

项目承担单位

清華大學
Tsinghua University

无锡江南计算技术研究所
JiangNan Institute of Computing Technology

中国科学院大气物理研究所
Institute of Atmospheric Physics, Chinese Academy of Sciences

北京航空航天大學
BEIHANG UNIVERSITY

中國科學院研究生院
GRADUATE UNIVERSITY OF CHINESE ACADEMY OF SCIENCES

国家气候中心
National Climate Center

北京应用物理与计算数学研究所
Institute of Applied Physics and Computational Mathematics

北京大學

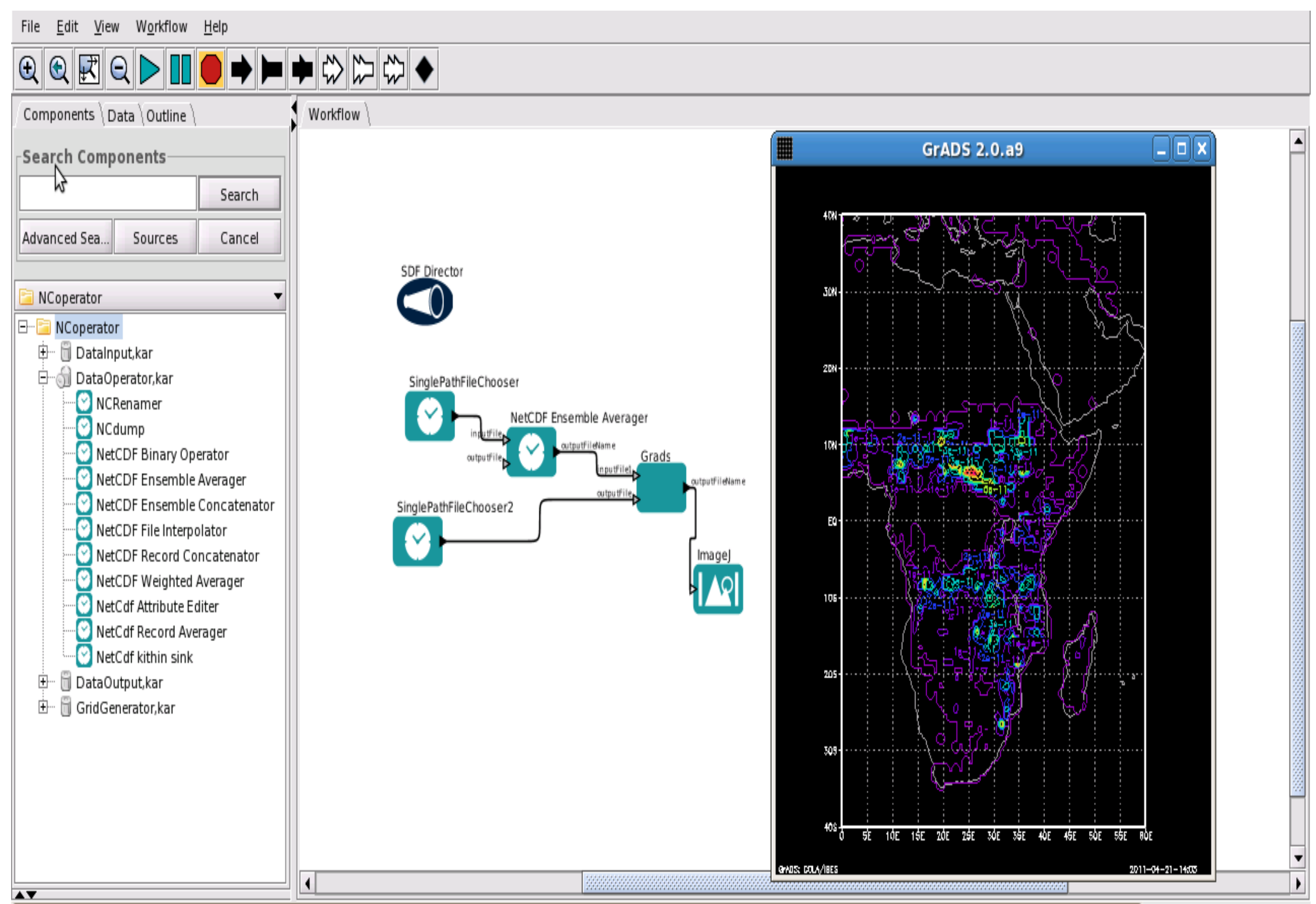
國防科學技術大學
National University of Defense Technology

中国科学院计算技术研究所
Institute Of Computing Technology Chinese Academy Of Sciences


应用层
用户层
命令行
客户端
Web
课题四
集成开发平台
Fortran代码辅助
项目管理视图
组件化的模式模块/模板库
运行时支持
支撑层
数据支撑
数据门户
数据处理
高速IO
MPMD调试分析
并行调试
性能分析
高可用
可视化诊断分析
可视化诊断分析工具包
脚本模板库
远程调用接口库
课题一
课题三
基础工具集
课题二
ESMF框架
中间件
编译器
调试器
作业管理器
版本控制器
资源层

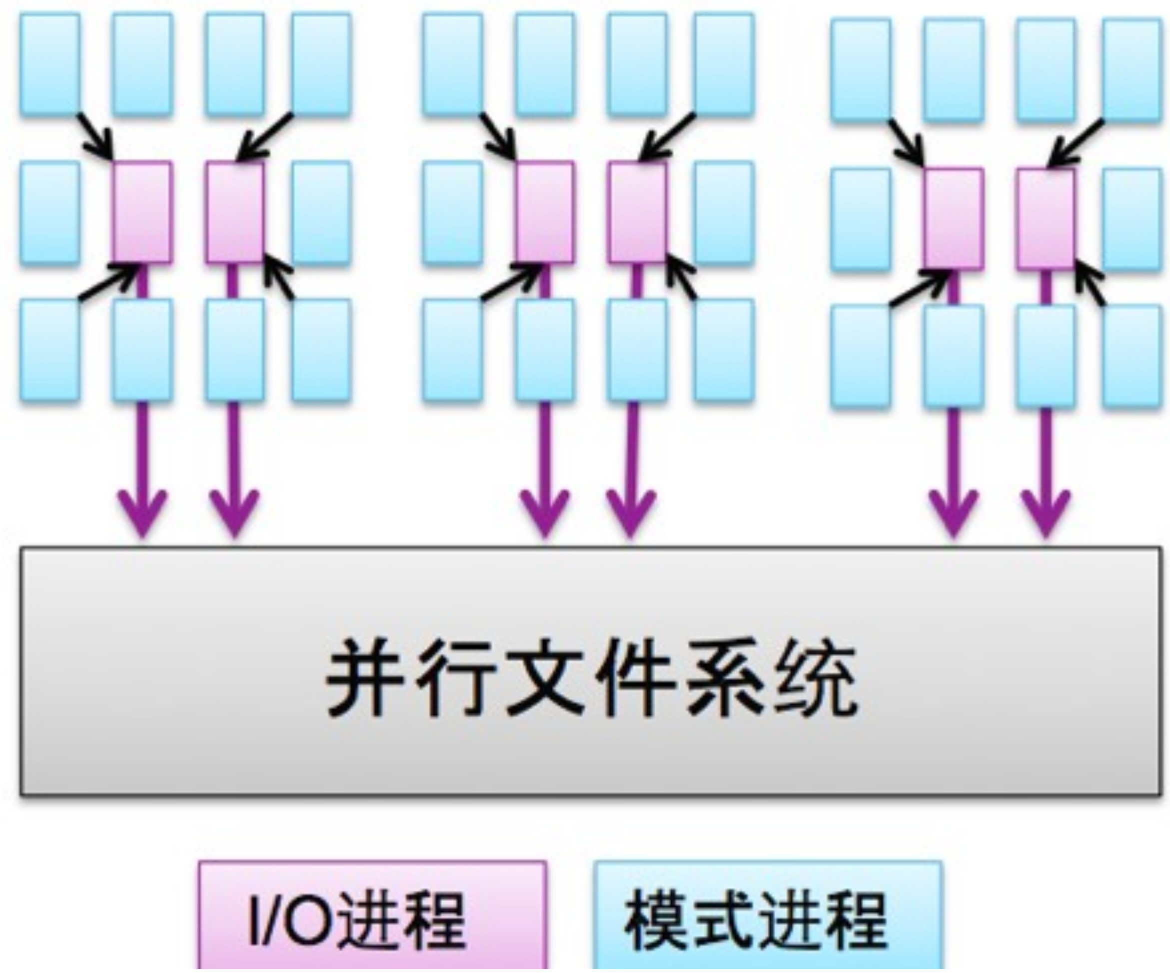
课题一亮点：数据处理工作流软件

- 软件功能：
  - 将一系列数据前处理工具和后处理工具封装为工作流模块
  - NetCDF处理工具并行化
  - 支持脚本辅助生成
  - 支持外部命令调用
  - 支持数据统计
  - 支持数据可视化
  - 一键安装

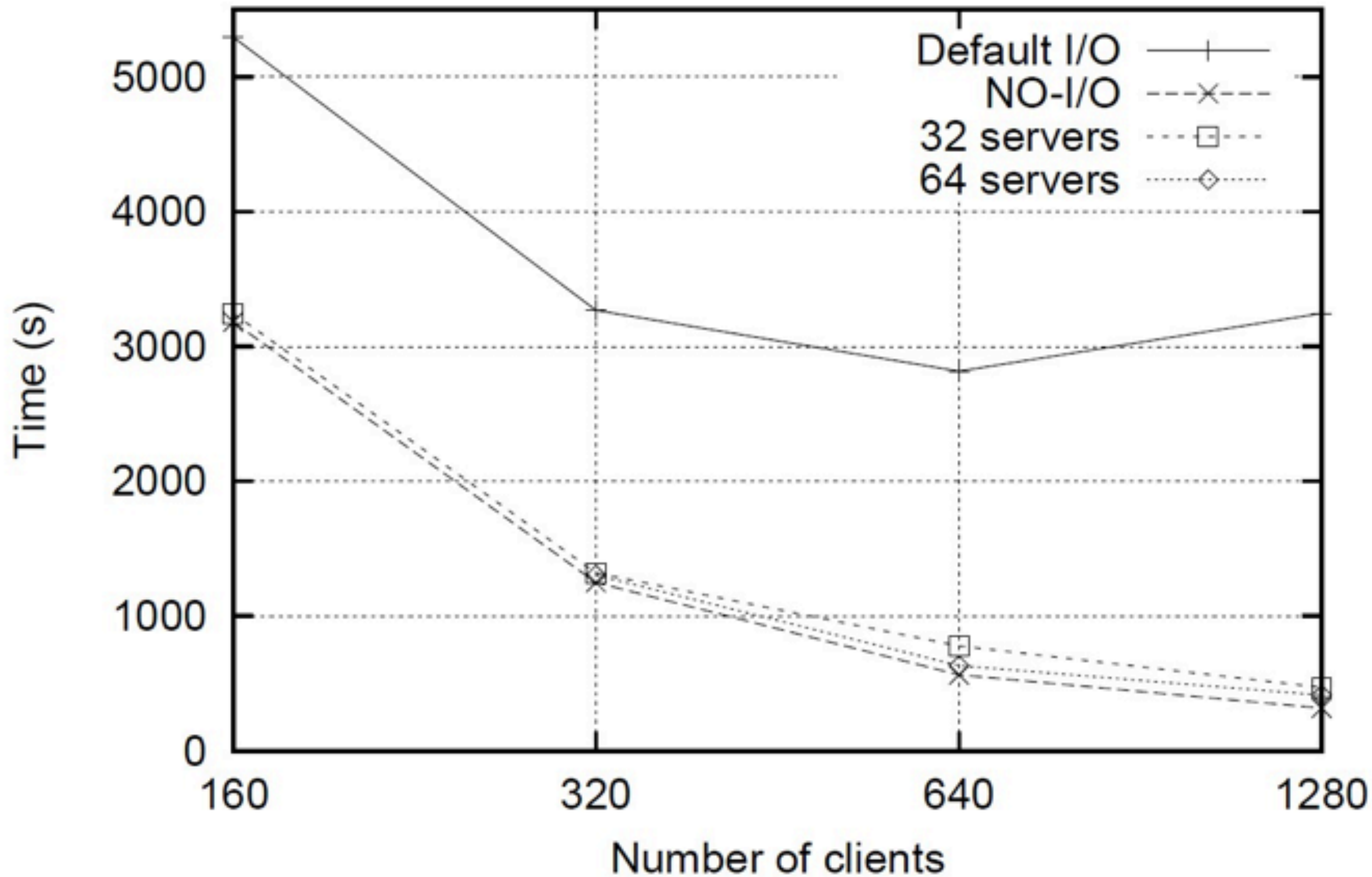


课题一亮点：高速I/O库

- 问题：
  - 高分辨率地球系统模式海量的输出数据使得I/O成为模式计算的瓶颈
- 方法：
  - 实现计算和I/O的重叠
  - 利用通信和内存实现二阶段异步IO
  - 符合NetCDF文件接口
- 结果：
  - 本研究工作正在代码实现阶段，初步得到的结果表明我们方案的性能5倍优于传统I/O



CFIO：A Fast I/O Library for Climate Models



POP海洋模式性能优化

A Scalable Barotropic Mode Solver for the Parallel Ocean Program

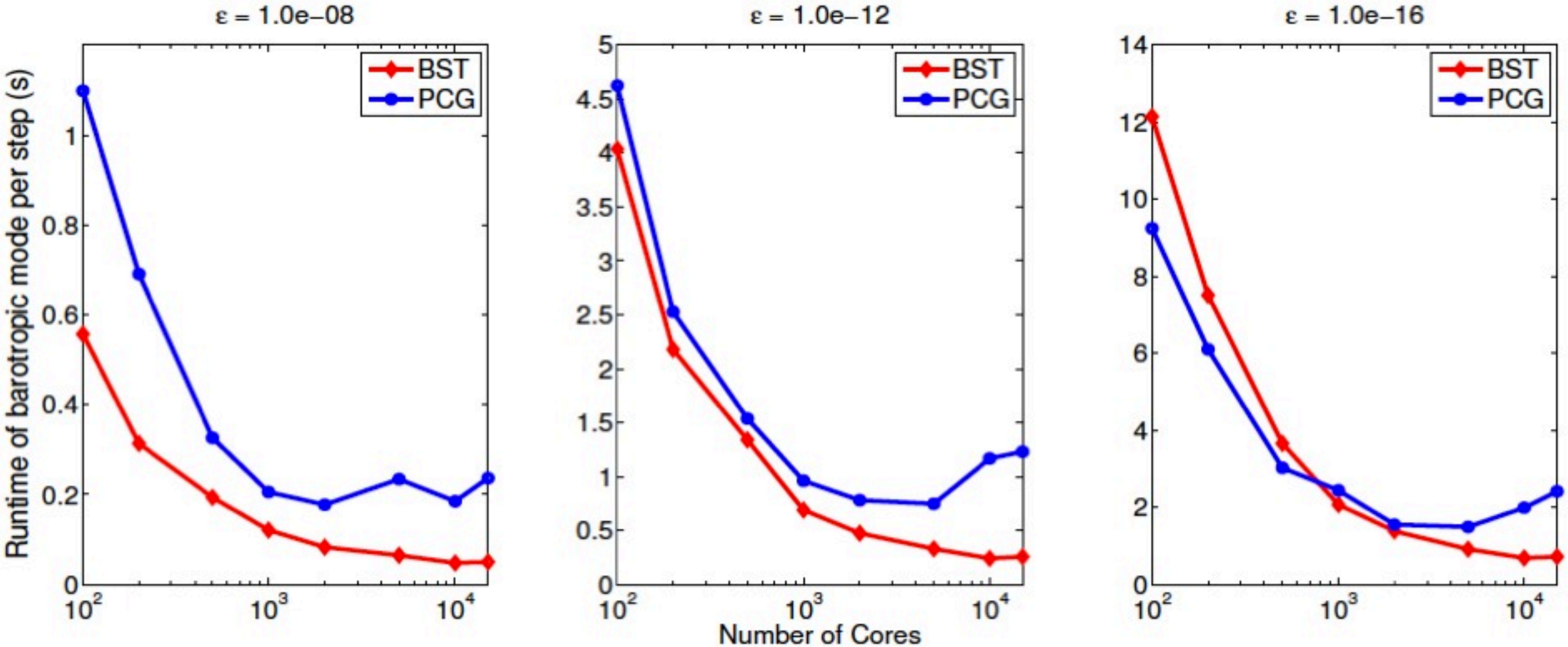
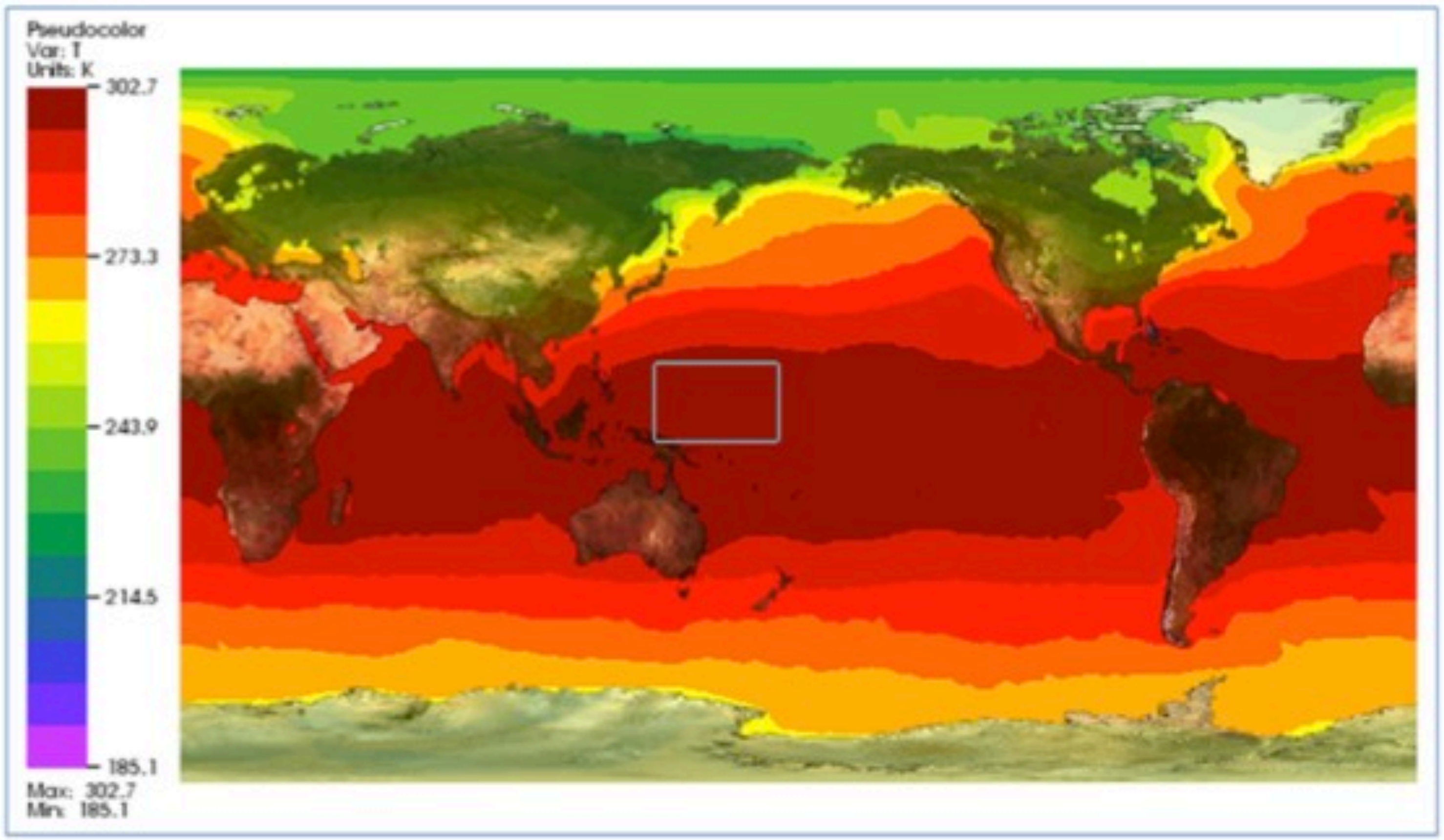
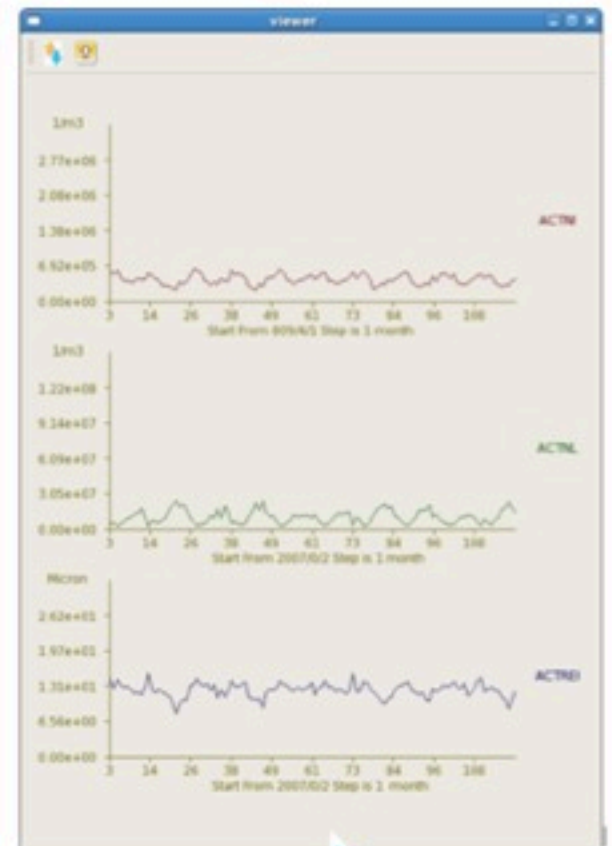


Fig. 5. Scalability of PCG and CSI in the 0.1 degree POP

课题二亮点：多维查询

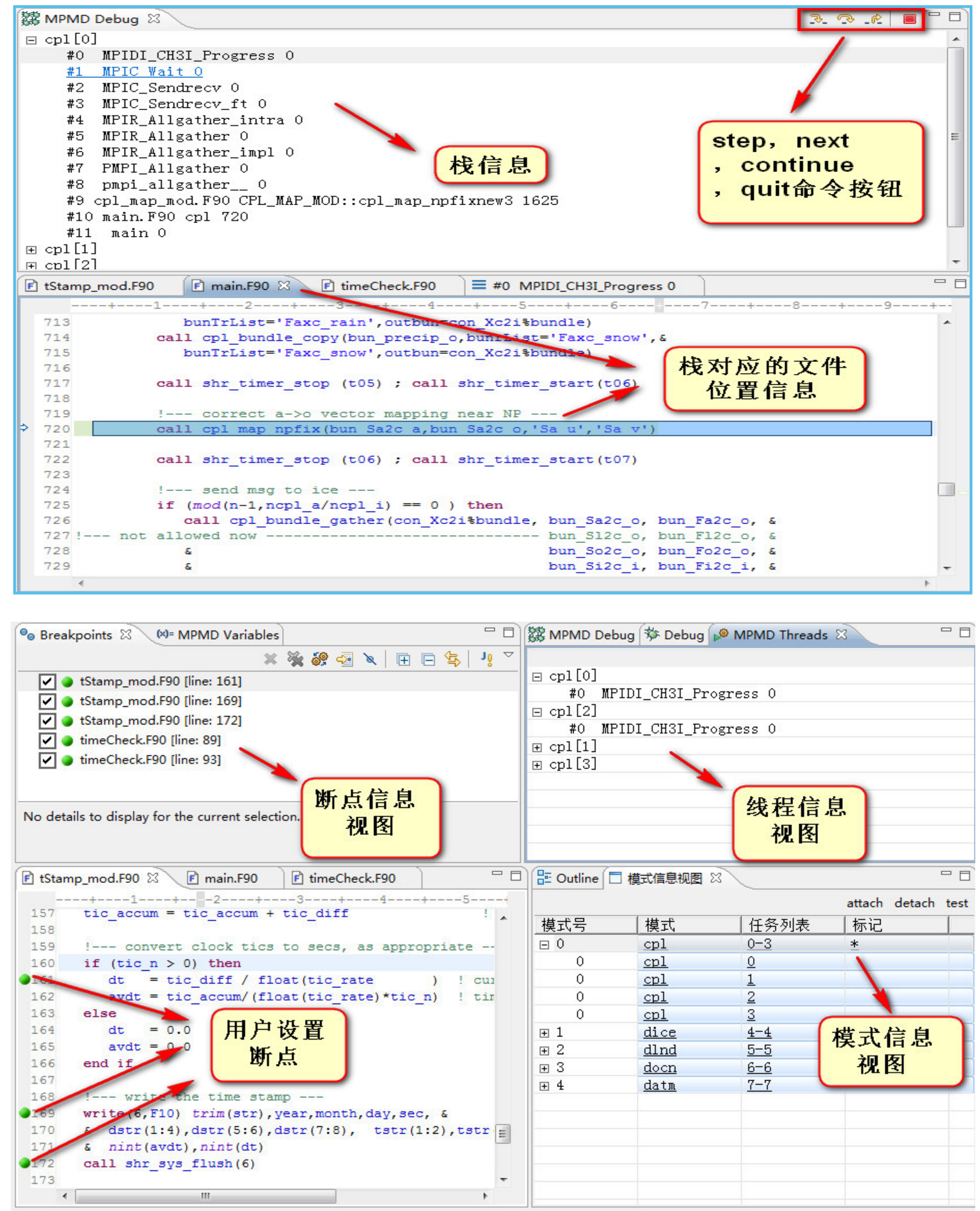




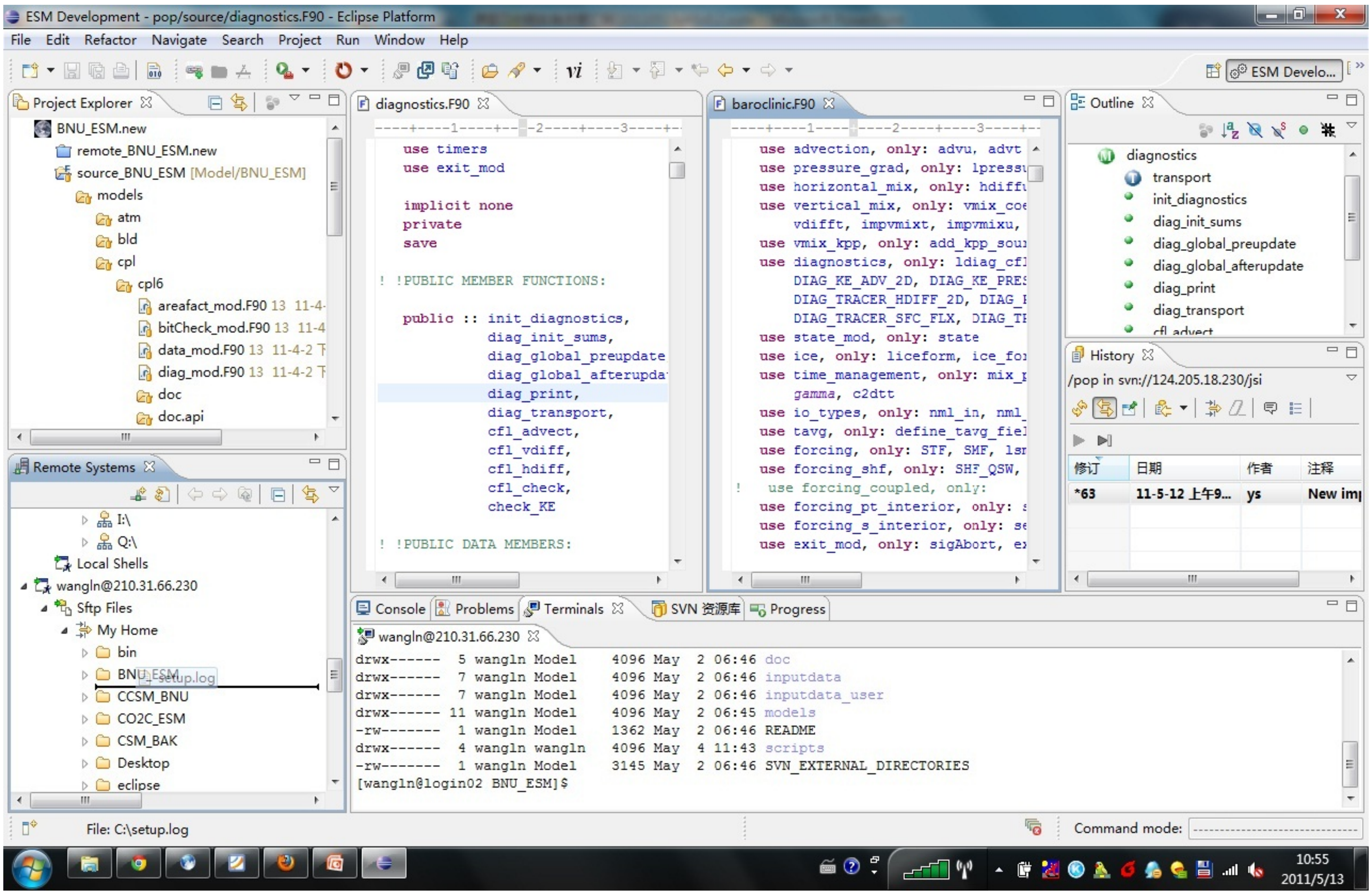
- 任意时间跨度和空间范围
- 既可直接选取，亦能精确设定
- 多口径统计
- 多变量对比
- 多窗口并列

课题三亮点：MPMD并行调试

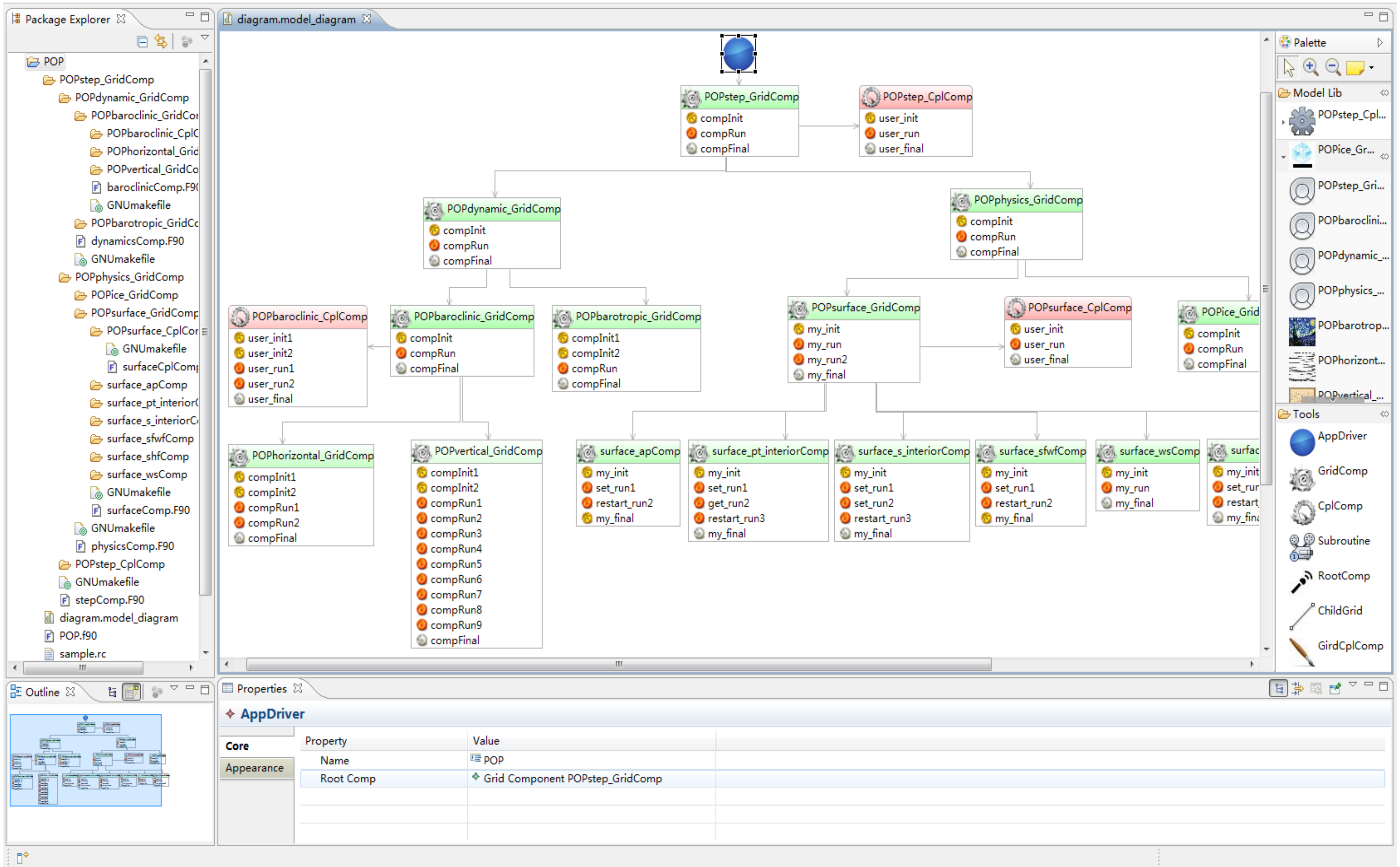
- 面向模式的MPMD并行调试
  - ✦ 从并行程序调试向耦合模式调试过渡
  - ✦ 面向多种并行粒度的混合编程模型调试
- 面向模式的MPMD性能分析
  - ✦ 以耦合器为中心进行分析，尝试将多个模式解耦合为模式间关系独立的SPMD程序
- 将北师大的GCCESM模式性能由25模式年/天提高到88模式年/天



课题四亮点：一体化集成开发平台



课题四亮点：模式模块库/模板库的构建



在实际模式中的应用

- “零编程”模式数据处理工作流软件
  - 实例：观测资料处理、CoLM陆面模式输入数据
- 模式输出数据多变量实时范围查询
  - 实例：GAMIL大气模式输出数据
- MPMD程序的模式级调试
  - 实例：CCSM3地球系统模式
- 模式模块的图形化拖拽和代码自动生成
  - 实例：POP海洋模式

上述用例覆盖国内外典型地球系统模式：NCAR的CESM、大气所的FGOALS_g、北师大的GCCESM

Thanks！
Q&A