

可変性と可読性を考慮した 大気大循環モデルの 設計と実装実験： 物理過程モジュールでの試み

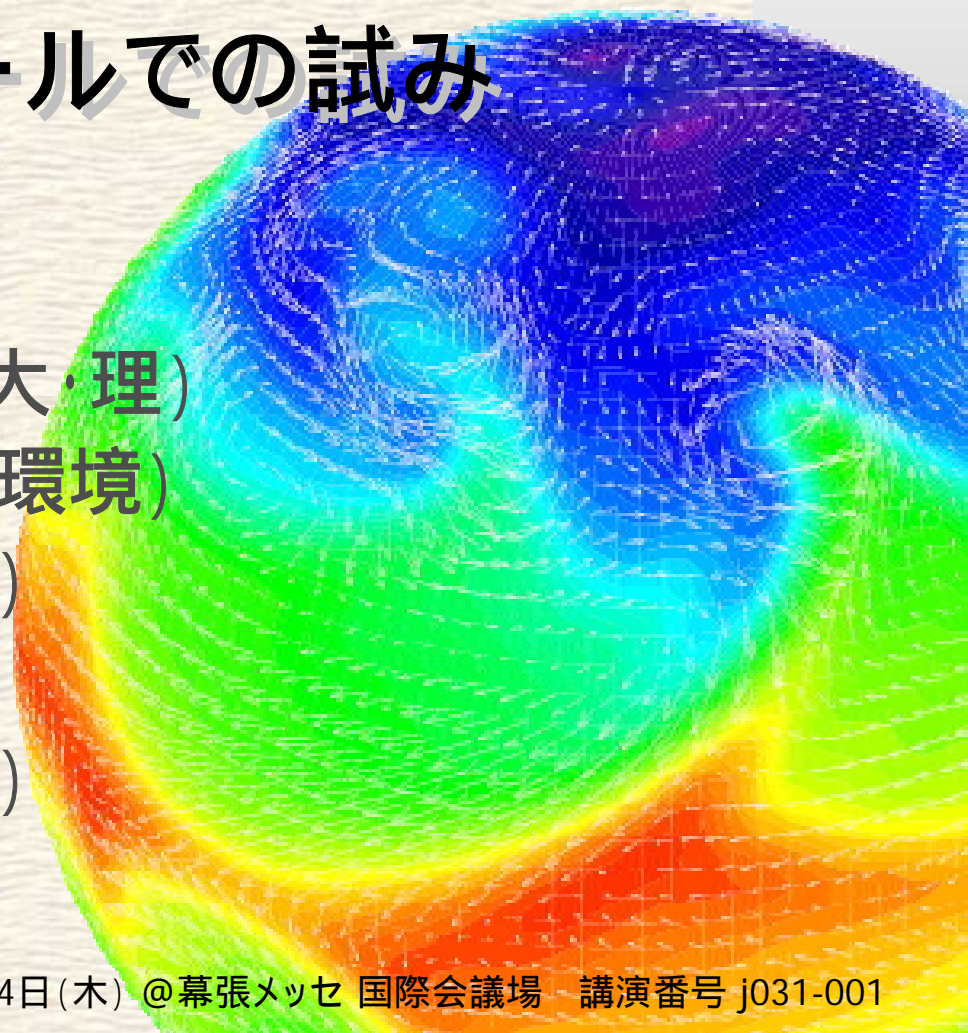
森川 靖大 (神戸大/北大・理)

石渡 正樹 (北大・地球環境)

高橋 芳幸 (神戸大・理)

小高 正嗣 (北大・理)

林 祥介 (神戸大・理)



なぜ可変性・可読性が重要か？

■ 仮想的な惑星大気の計算

- 計算条件を手軽に変更
 - ◆ 大気組成
 - ◆ 入射太陽放射量
 - ◆ 重力加速度
 - ◆ 自転周期
 - ◆ 大気圧
 - etc. ...



可変性・可読性を重視した 大気大循環モデルの試み

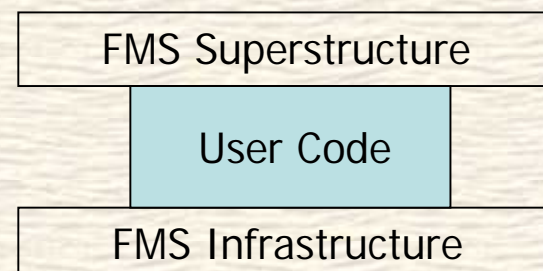
3/13

■ AGCM5 (沼口, 1992; SWAMP Project, 1998; <http://www.gfd-dennou.org/arch/agcm5>)

- 変数命名規則・プログラム書法の工夫
- FORTRAN77 の制約大

■ FMS (Flexible Modeling System; Geophysical Fluid Dynamics Laboratory, 2005)

- 基盤部分 (I/O, 並列化等)、モデル (大気, 海洋等) 結合部分の隠蔽
- 放射スキームなどの素過程の交換や分離に関して工夫の余地有



■ DCPAM (2005 年合同大会)

- 支配方程式から容易に想像できるソースコード
(dcmode1 コーディングルール <<http://www.gfd-dennou.org/library/dcmode1>>)
 - ◆ 配列演算関数の利用 (スペクトル演算には spmodel ライブラリ)
 - ◆ 一目で次元や物理的意味の分かる変数命名規則

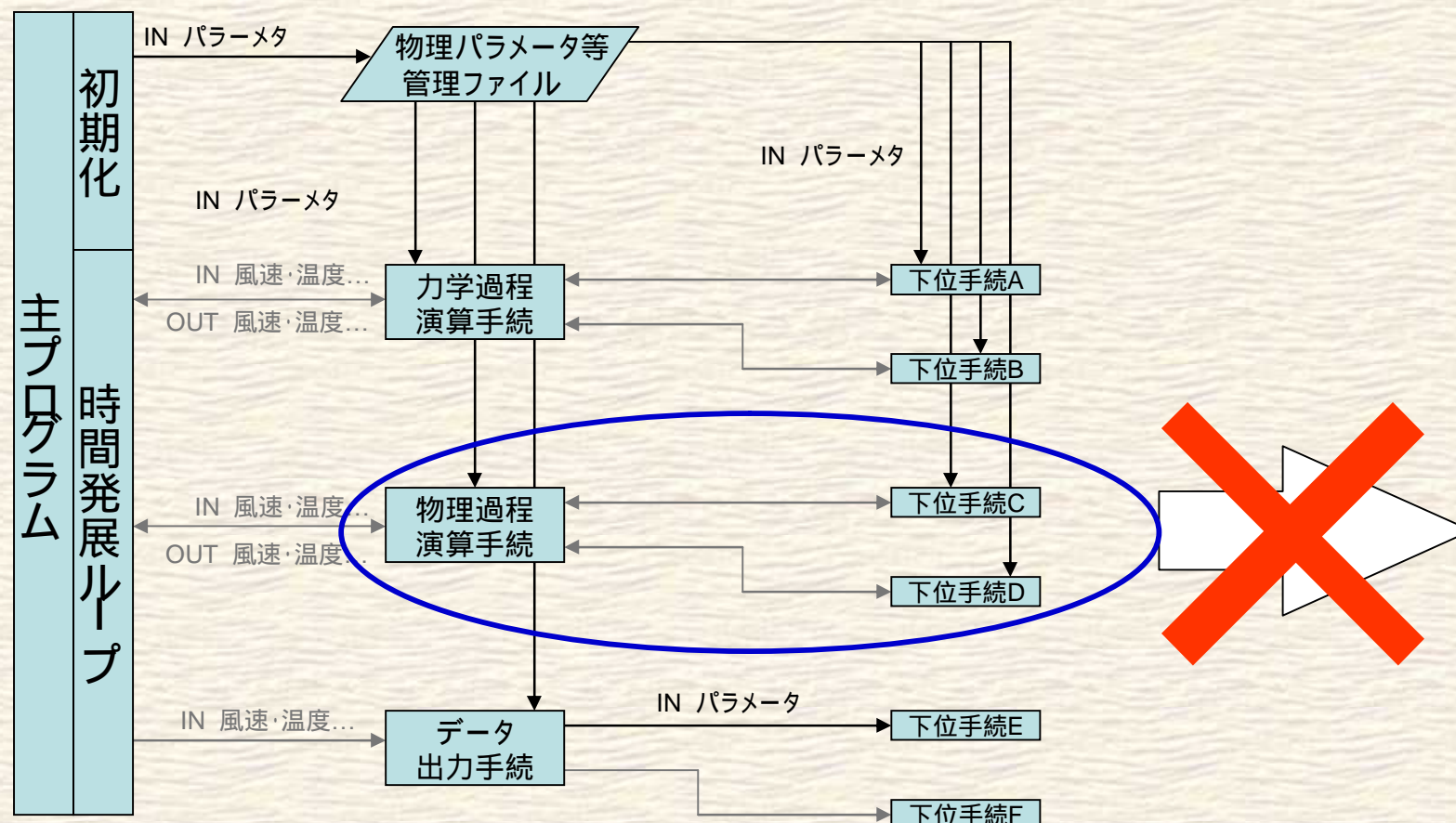
現状のモデルの問題点

- 物理過程モジュールの交換が面倒
 - モジュール間の依存性の整理が不十分
- デバッグが面倒
 - ソースコードが膨大
 - デバッグのたびにコード全体をコンパイルして実行
- ドキュメント作成が面倒
 - ソースコード解説文書とソースコード本体とを別々に作成する必要あり

物理過程交換のためのモジュール設計(1)

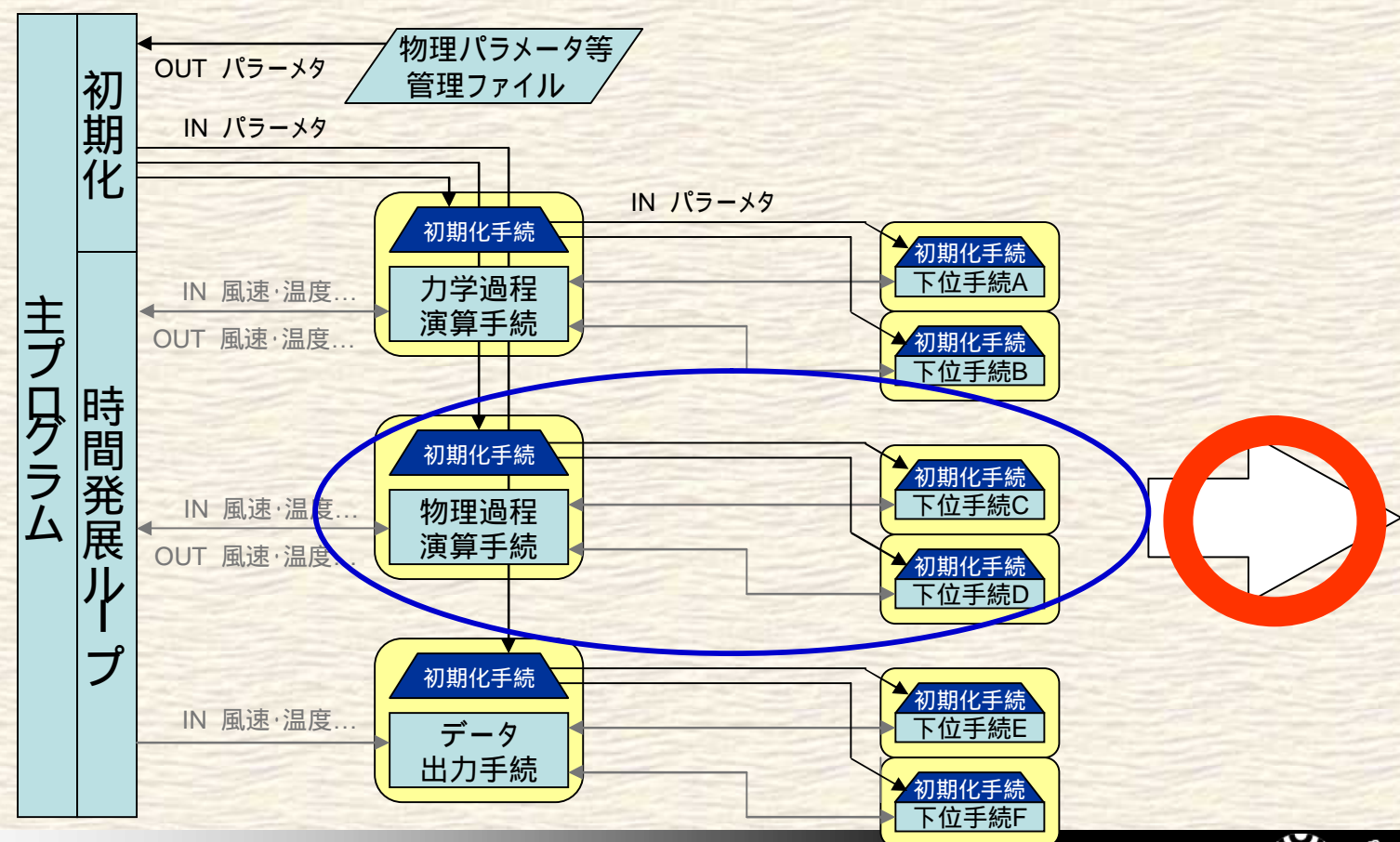
■ モデル設定パラメータの集中管理

- モデルを構成する個々のサブセットを切り離しが面倒



物理過程交換のためのモジュール設計(2)

- 個々の演算に必要なパラメータは各モジュールごとに保持
 - 各モジュールで初期化手続を用意し、その手続でパラメータを設定
 - モジュールをサブセットとしてモデルから切り離す作業コストが低減

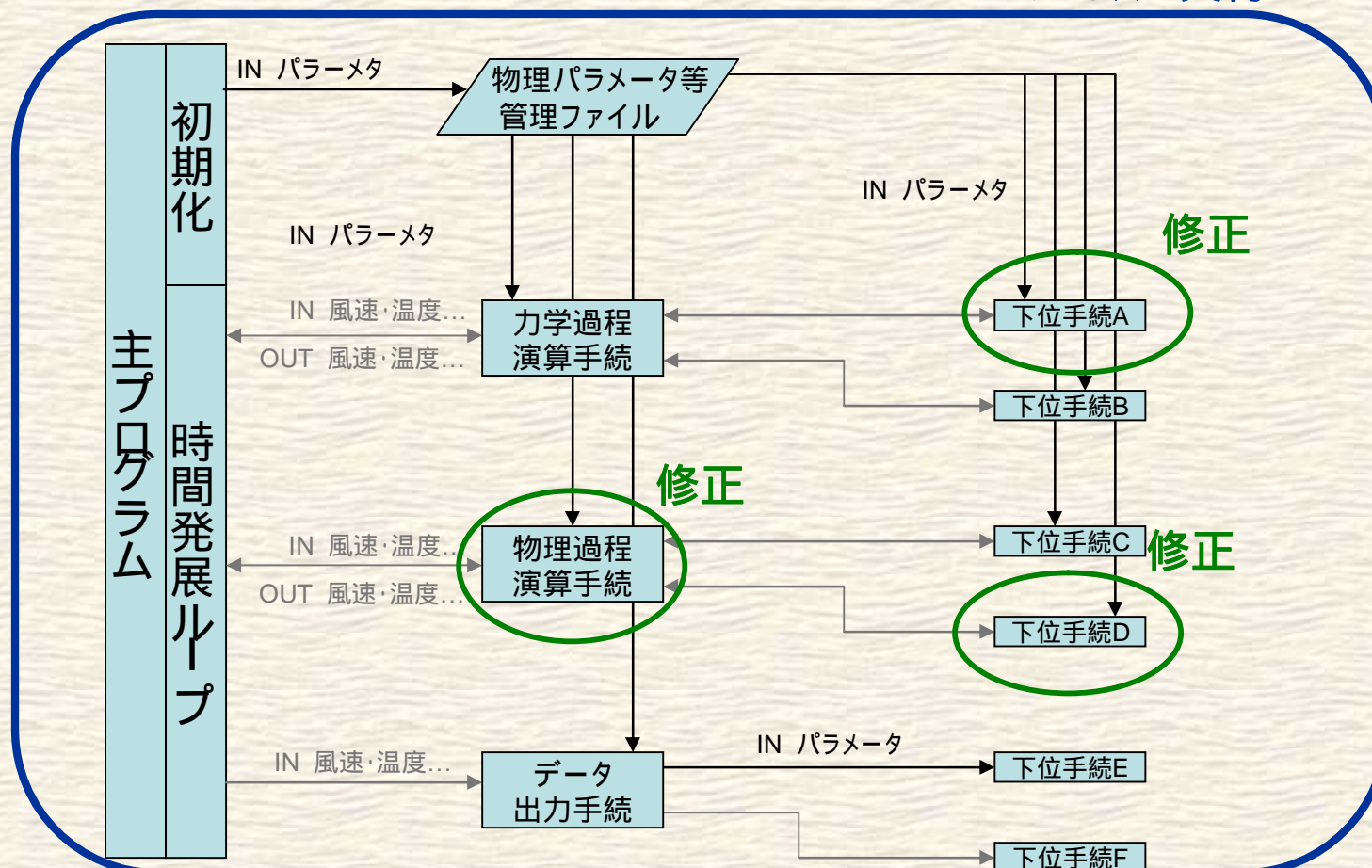


単体テスト (1)

■ 常に全体テスト

- 細かい修正のチェックのたびに全体でコンパイルと実行の必要あり
- デバッグの手順が煩雑に

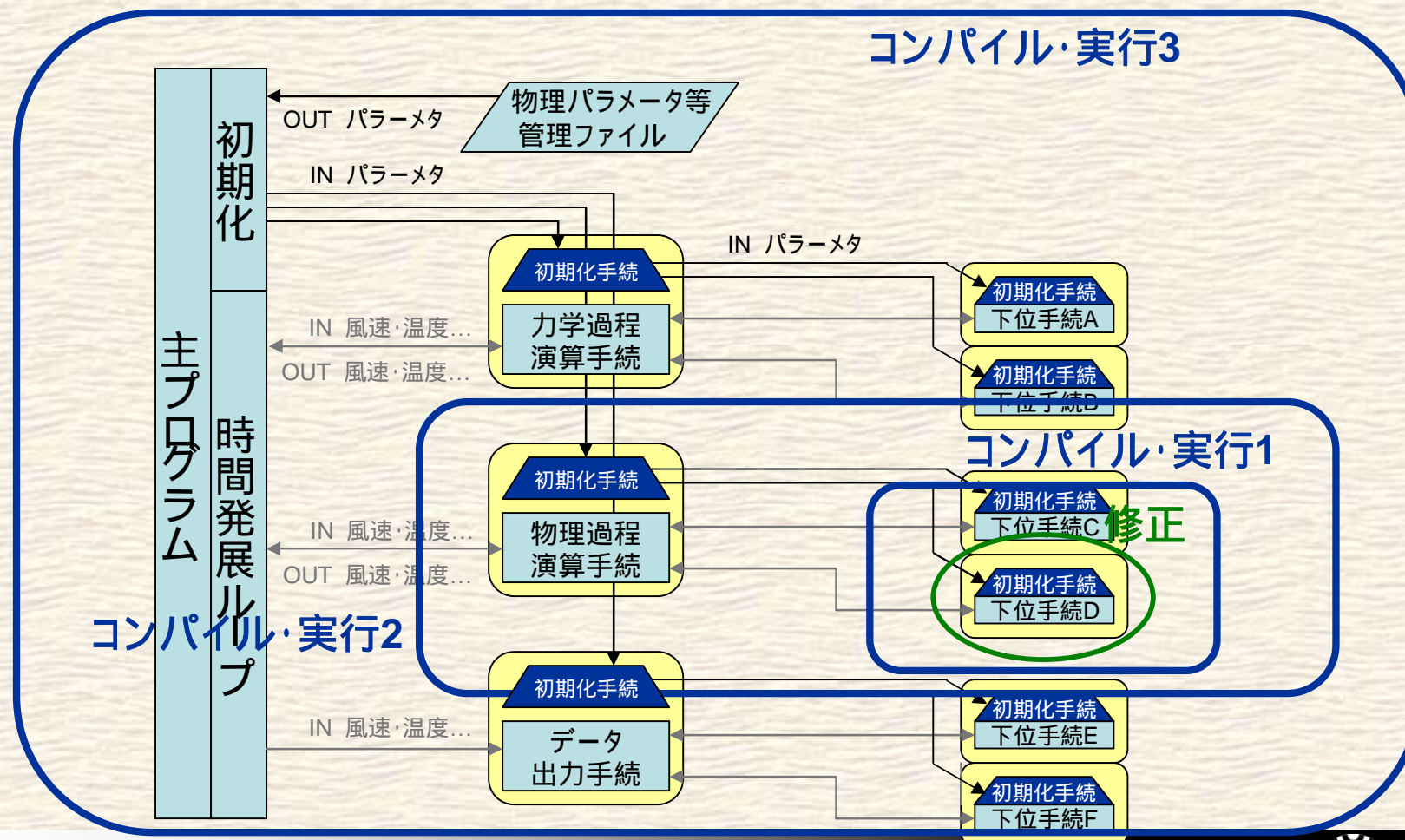
コンパイル・実行



単体テスト (2)

■ モジュール毎に個別テスト

- 小さいプログラム単位でチェックした後大きなプログラム単位へ



単体テスト (3)

■ 簡単テストプログラムによるテストの自動化

- 簡単な動作テストプログラム
 - ◆ ここで言う「テスト」= 答えが分かっている場合について、ちゃんと計算できているか確認すること
- デバッグの際はまともに計算・可視化の必要あり。しかし、まずは誤りが存在することを自動的に検出できることが大事
- 自動化してコマンド1つでチェックできることが大事

■ gt4f90io ライブラリ (2004 年 合同大会) のサブセットとして dc_test モジュール作成

- 組み込み型変数・配列 (1 ~ 7次元) に関して与える 2 つの引数を比較 (等しいかどうかのチェック)

単体テスト (4)

■ dyn_spectral_as83_test.f90

```
program dyn_spectral_as83
  use dc_test, only: AssertEqual
  use dyn_spectral_as83, only: Calculation
  integer :: nmax, imax, kmax, ...
  :
  call Calculation( &
    & xyz_VorB, xyz_DivB, xyz_TempB, xyz_QVapB, xy_PsB, & ! (in)
    & xyz_VorA, xyz_DivA, xyz_TempA, xyz_QVapA, xy_PsA ) ! (out)
  :
  call AssertEqual( message='Vorticity test' &
    & answer = (/ -1.208654e-08, -1.235772e-08, ... &
    & :
    & ... /), &
    & check = xyz_VorA )
  :
  call AssertEqual( message='Temperature test' &
    & answer = (/ 299.9977, 299.9978, ... &
    & :
    & ... /), &
    & check = xyz_TempA )
  :
end program dyn_spectral_as83
```

演算ルーチン

チェックルーチン

Make コマンド

自動的にコンパイルと実行

正答

正答

チェックメッセージ
誤りの検出

```
% make test
f90 dyn_spectral_as83_test.f90 dyn_spectral_as83.o ... -L/usr/lib
-o dyn_spectral_as83_test
./dyn_spectral_as83_test
*** MESSAGE [DCAssertEqual] *** Checking Vorticity test OK
:
*** MESSAGE [DCAssertEqual] *** Checking Temperature test FAILURE

check(22,16,3) = 328.2
is INCORRECT
Correct answer is answer(22,16,3) = 238.5
```

■ 解説文書の維持・更新のコスト高

- モデルを構成する個々のモジュールが頻繁に交換・変更されることを想定
- セットとなる解説文書も同時に手動で維持・更新するのは大変
 - ◆ ソースコードの書き換えだけでも大変
- 解説文書がどんどん廃れる
 - ◆ ソースコードを書いた本人も利用法を忘れる
 - ◆ 第三者の利用はほぼ不可能

RDoc による解説文書自動生成 (2)

12/13

- RDoc Fortran 90/95 強化版 (森川 他、2007、天気 Vol 54 No 2、2006 年 連合大会) によって解説文書を自動生成
 - 解説文書が低コストで定常的に更新

```
module dyn_spectral_as83
  != 力学過程
  ! 力学過程を演算するモジュールです。
  ! 水平離散化にスペクトル法を、鉛直離散化には
  ! Arakawa and Suarez (1983) を用いています。
  != List
  ! Create      :: 初期化手続
  ! Calculation  :: 演算手続
  !
  :
  interface Calculation
    module procedure DynSpectralAS83Calculation
  end interface
  :
contains
  subroutine DynSpectralAS83Calculation( &
    & xyz_VorB, xyz_DivB, xyz_TempB, xyz_QVapB, xy_PsB, &
    & xyz_VorN, xyz_DivN, xyz_TempN, xyz_QVapN, xy_PsN, &
    & xyz_VorA, xyz_DivA, xyz_TempA, xyz_QVapA, xy_PsA )
    !
    ! 与えられた $ t %Delta t $ および $ t $ の
    ! 渦度、発散、温度、比湿、地表面気圧から、
    ! $ t+%Delta t $ の渦度、発散、温度、比湿、地表面気圧 を
    ! 返します。
    !
    ! 時間積分法にはリーブログスキームを用いています。
  end subroutine DynSpectralAS83Calculation
end module dyn_spectral_as83
```



Files	Classes	Methods
dyn_spectral_as83.f90 phys_radiation.f90	dyn_spectral_as83 phys_radiation	Calculation (phys_radiation) Calculation (dyn_spectral_as83) Create (phys_radiation) Create (dyn_spectral_as83) DelTemp (phys_radiation)

Class **dyn_spectral_as83**

In: dyn_spectral_as83.f90

力学過程

力学過程を演算するモジュールです。水平離散化にスペクトル法を、鉛直離散化には Arakawa and Suarez (1983) を用いています。

List

Create : 初期化手続
Calculation : 演算手続
Put_Line : 設定の表示

Methods

Calculation Create Put_Line

Included Modules

dc_types wa_module w_module dc_message dc_trace dcpam_error

まとめ

- **DCPAM (Dennou Club Planetary Atmospheric Model)**
 - <http://www.gfd-dennou.org/library/dcpam>
- **今回の試み**
 - 物理過程交換のためのモジュール設計
 - 単体テストのためのモジュール設計とライブラリ整備
 - RDoc による解説文書自動生成
- ***to be continued ...***
 - 基礎実験を通じた検証
 - ◆ 地球条件における動作の検証
 - ◆ 火星・金星・木星表層大気のための物理過程交換の簡便さの検証
 - ◆ 仮想惑星のパラメタスタディへ
 - 実行速度の改善
 - ◆ 関数利用を原因とするパフォーマンス低下の定量的検証
 - ◆ 大規模計算のための並列化実装

参考文献

- Balaji, V.: The FMS Manual: A developer's guide to the GFDL Flexible Modeling System.
<http://www.gfdl.noaa.gov/~vb/FMSManual/FMSManual.html>
- The flexible modeling system (FMS). <http://www.gfdl.noaa.gov/~fms/>, GFDL
- 森川 靖大, 小高正嗣, 石渡 正樹, 林 祥介, gtool4 開発グループ, 2006: gt490io ライブラリ, <http://www.gfd-dennou.org/library/gtool4/>, 地球流体電脳倶楽部.
- 森川靖大, 石渡正樹, 堀之内武, 小高正嗣, 林祥介, 2007: RDoc を用いた数値モデルのドキュメント生成. 天気, 54, 185--190.
- 沼口 敦, 1992: 博士論文.
- RDoc: <http://www.ruby-doc.org/stdlib/libdoc/rdoc/rdoc/>
- Ruby: <http://www.ruby-lang.org/>
- SWAMP Project, 1998: AGCM5. <http://www.gfd-dennou.org/arch/agcm5/>. 地球流体電脳倶楽部
- 竹広 真一, 小高 正嗣, 石岡 圭一, 石渡 正樹, 林 祥介, 2006: 階層的地球流体スペクトルモデル集 SPMODEL. ながれマルチメディア 2006.
- 竹広真一, 石岡圭一, 森川靖大, 小高正嗣, 石渡正樹, 林祥介, SPMODEL 開発グループ, 2004: 階層的地球流体力学スペクトルモデル集 (SPMODEL), <http://www.gfd-dennou.org/library/spmodel/>, 地球流体電脳倶楽部.