

# RDoc を用いた数値モデルのドキュメント生成

\*森川靖大 (北大理), 石渡正樹 (北大地球環境),  
堀之内武 (京大生存研), 小高正嗣, 林祥介 (北大理)

## 1. はじめに

数値モデルの開発, カスタマイズ, 保守を効率的に行うためには, モデルの仕様を解説したドキュメントの整備がなされていることが肝要である. また, 数値モデルを利用する第三者にとってもドキュメントが提供されていることは必須である.

しかし, 例えば, ドキュメントのファイルとソースコードのファイルを別々に作成し管理するような, 開発者にとって手間のかかる形態を取っていると, ドキュメント整備が放置されがちとなる. 小規模開発においては, ソースコードにおける編集作業と似て非なる編集作業をドキュメントにおいても開発者自身が別途行う必要があるため, 大きな負担感となってしまうからである. ドキュメント作成の手間を軽減し, 小規模開発でもドキュメント整備を促進させるためには, ソースコードにドキュメント用の記述を埋め込んでおいて一つのファイルとして作成管理できることが望ましい.

## 2. RDoc によるドキュメント自動生成

Fortran90/95 で書かれた数値モデルのドキュメント, 特に個別のサブルーチンやモジュールの機能を記述したリファレンスマニュアルの管理を容易にすることに關しては, これまでに数多の試みがある. 例えば GFDL の Flexible Modeling System (FMS) の開発者向け手引書 The FMS Manual (Balaji, 2002) では, ソースコードのコメント文に XML 書式の記述を埋め込み, それらをスクリプトで抽出するという方法をとっている. しかし, この方法では, モジュール間の依存関係は明示的に書き込まねばならず, また, XML 埋め込みはソースコードの可読性を下げてしまう.

ソースコードの解析自動化を進めた方法としては, オブジェクト指向スクリプト言語 Ruby の標準ライブラリである RDoc の Fortran90/95 解析機能を使うというものがある. RDoc 付属のソースコード解析プログラムによってソースコード自体を解析し, HTML 生成プログラムによってコメント文に記載された文書と合わせて HTML 形式のリファレンスマニュアルを作成する. しかし, 現在の RDoc の Fortran90/95 解析機構では, ファイル, モジュール, サブルーチン以外の要素, すなわち関数や変数などを抽出できないという問題や, コメント文を解釈してリファレンスマニュアルに取り込む機能もあまり充実していないという問題がある.

## 3. Fortran90/95 解析機構の改良

我々は RDoc の Fortran90/95 ソースコード解析機構に改良を施し, より充実した情報量を持つリファレンスマニュアルの生成を可能とする機能を持たせた. 改良した RDoc は, Fortran90/95 ソースコードからモジュール, サブルーチン, 関数, 変数, 定数, 構造型定義, 利用者定義演算, 利用者定義代入を抽出し, それらの直後 (行末もしくは直後に続く行) に書かれたコメント文と併合して HTML 形式のリファレンスマニュアルを自動生成する (図 1). モジュール間の依存関係も自動的に解析し, ハイパーリンクとして反映する. サブルーチンや関数に關しては, その引数の型や引数の直後に記述されるコメント文も解析し, 自動的にリファレンスマニュアルに反映する.

我々はこの RDoc Fortran90/95 ソースコード解析機能強化版を <http://www.gfd-dennou.org/library/dcmodel/> にて公開している.

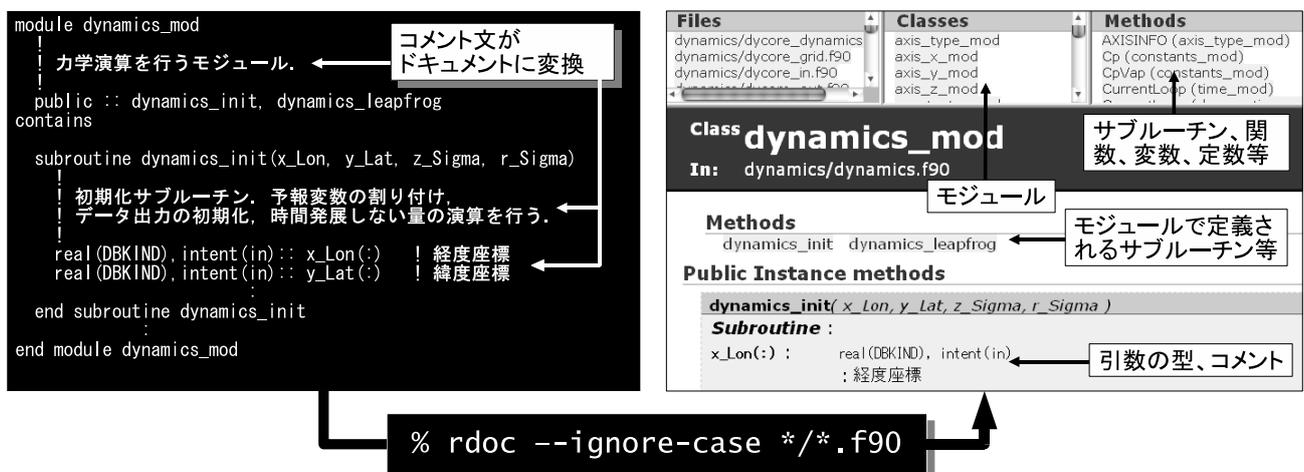


図 1: Fortran90/95 ソースコード (左) から自動生成される HTML 形式のリファレンスマニュアル (右). 一覧表示されているモジュール, サブルーチン, 関数, 変数, 定数などは, それぞれの解説文書へのハイパーリンクとなる.