

GT00L3 内部処理のオブジェクト指向的再解釈: gtool4 への実装のためのレビュー

Object-oriented reinterpretation of internal data processing in GT00L3: review for preparation of gtool4

豊田 英司[1], 石渡 正樹[2], 竹広 真一[3], 林 祥介[4], 地球流体電脳倶楽部 Davis Project
Eizi TOYODA[1], Masaki Ishiwatari[2], Shin-ichi Takehiro[3], Yoshi-Yuki Hayashi[4], GFD Dennou Club Davis Project

[1] 気象庁 数値, [2] 北大・地球環境, [3] 九大・理・地惑, [4] 北大・理・地球惑星

[1] JMA NPD, [2] Graduate School of Environmental Earth Science, Hokkaido University, [3] Earth and Planetary Sci., Kyushu Univ., [4] Earth and Planetary Sci., Hokkaido Univ.

<http://www.gfd-dennou.org/arch/gtool4/>

(序論)

本大会 (2000 年以降) で報告してきたように、著者らは大規模な地球科学数値データを念頭においた解析可視化ツール/ライブラリ 'gtool4' を開発してきた。試験的な実装において必要最低限の機能、すなわち時系列格子点データの作成、要素毎演算、そしてコンタリング・塗り分けによる可視化ができるようになり、利用も次第に広がっている。一方、データ処理ツールの開発しやすさ、gtool4

netCDF 規約への適合性などが課題となっている。今回はこのために GT00L3 ライブラリの再分析を行った結果と gtool4 での実装構想について報告する。

(背景)

「gtool4」は GT00L3 の後継たるべく名付けられた。GT00L3 は地球流体科学の 3 次元格子点データのためのファイル形式仕様、入出力ライブラリ、解析可視化ツールの総称である。もとは 1990 年代始めから大気大循環モデル AGCM5 のために開発されたものであり、地球流体電脳倶楽部の他のモデルでも利用されている(すべてのソフトウェアは地球流体電脳倶楽部のウェブサイトからダウンロード可能である)。データ解析ツールは簡単なデータ解析、たとえば部分配列の抽出や軸に沿った平均、DCL を利用した線画可視化などの機能を提供する。gtool4 は GT00L3 のデータ可搬性(ネットワーク透過性)、拡張性、プログラム可読性を改善するためのプロジェクトで、新しく netCDF ファイル形式をベースとし Fortran 90 を用いて全面的書き直しをしている。

(GT00L3 の内部処理)

GT00L3 ファイルは Fortran の順番探査書式なしファイルに基づいている。記録は先頭から交互にヘッダ、データと呼ばれ、ヘッダにメタデータが格納されている。データは最大 3 次元の配列であり、アプリケーションはファイルからヘッダとデータ配列の対を逐次読み取って処理する。データ処理プログラムにおいては内部処理でもヘッダ(文字型配列として保持される)とデータ配列の対応関係が維持されており、配列演算などのサブルーチンにはこの対が引数として渡され、ヘッダの情報を参照して配列寸法や空間ジオメトリなどの判定が行われる。これらのサブルーチンではデータ履歴などのヘッダ属性の管理も共通化されているため、データ処理ツールのメインロジックでは本質的な処理内容を記述するだけで多くのヘッダ属性が自動的に管理されるという仕掛けである。オブジェクト指向的な言い方では、ヘッダとデータ配列の対が「属性付き配列オブジェクト」であり、GT00L3 ライブラリのサブルーチン群の多くはこのクラスに属するメソッドと説明できる。実際、四則演算サブルーチンなどはオブジェクトの状態を変更する(破壊的)版と、新しいオブジェクトを作成する(非破壊的)版まで用意されていた。

(今日的再実装の考察)

gtool4 の試験的な実装において Fortran 90 の構造型を用いた抽象データ型によるコーディングスタイルが確立されているので、これによって GT00L3 で見られたオブジェクト構造を再現すれば同機能をより簡潔かつ明解に書くことができる。ただし、ファイルが順番探査でなくなったので、GT00L3 のクラス構成をそのまま写像することは適当ではない。汎用機の磁気テープに由来する順番探査ファイルでは前後の記録に依存せず記録が自己記述する必要があったが、netCDF では配列全体が自己記述の単位であるため、属性を保持する「変数」とその一部を読み取った「配列」は異なるオブジェクトとするのが適当だからである。現在の試験的な実装では「変数」だけが抽象データ型化されているが、これから「配列」も抽象データ型化することにより、データ処理ツールの開発を進めていきたい。

(参照)

DCL. <http://www.gfd-dennou.org/arch/dcl/>

GFD-Dennou Club. <http://www.gfd-dennou.org/>

GFD-Dennou Club Version of AGCM5. <http://www.gfd-dennou.org/arch/agcm5/>