New features in MAPL

History Parser, ExtData, and Running on a Desktop
GMAO Brown Bag - October 30, 2013

Outline

Discuss several items of that could be of interest to GEOS5 users and developers:

- Parser capability in History
- The ExtData component
- Running GEOS5 on a Desktop

MAPL Arithmetic Parser: What is it?

- The MAPL arithmetic parser is a MAPL component to evaluate an expression element by element for given set of input fields
- Currently can be used in History and ExtData to evaluate new fields from existing ones.
- In Ganymed-1_0 onward.
- Can easily be used elsewhere; simple subroutine that takes an esmf state and an expression and returns an esmf field. I can easily make more overloads if needed.

Example History Collection

```
geosgcm_prog.template:  '%y4%m2%d2_%h2%n2z.nc4',
geosgcm_prog.archive:   '%c/Y%y4',
geosgcm_prog.format:    'CFIO',
geosgcm_prog.frequency:  060000,
geosgcm_prog.resolution: 144 91,
geosgcm_prog.fields:    'U'              , 'DYN'            ,
                         'V'              , 'DYN'            ,
                         'T'              , 'DYN'            ,
                'sqrt(U*U+V*V)' , 'DYN'      'Wind_Magnitude'   ,
          '(T-273.15)*1.8+32.0' , 'DYN'    , 'TF' ,
                         'T^3'            , 'DYN',    'T3' ,
                ::
```

Note: Expressions are evaluated before temporal averaging if the collection is not instantaneous and before any spatial regridding.

What Can Be In An Expression?

- Parentheses and arithmetic operators: +-*/^()
- Field Names
- Single argument functions such as sqrt, log, etc. See documentation for full list, are case insensitive
- constants 2.0E3, 2.4e-3, -0.1d37, etc …
- Operations evaluated in the usual order (), ^, */, finally +-

Examples:
sqrt(A*B-C)
(A*B)^3.0
log(A)-(B+C)/2.0e1

Rules and Limitations for Expressions in a History Collection

- Fields in an expression must be a real model field
- If model field has an alias, use that in an expression
- Can't mix center and edge variables in an expression
- Can mix 2D and 3D. 2D gets promoted to 3D and replicated in each layer for operation.
- Fields in expression must be in the collection, or in the component name following the expression
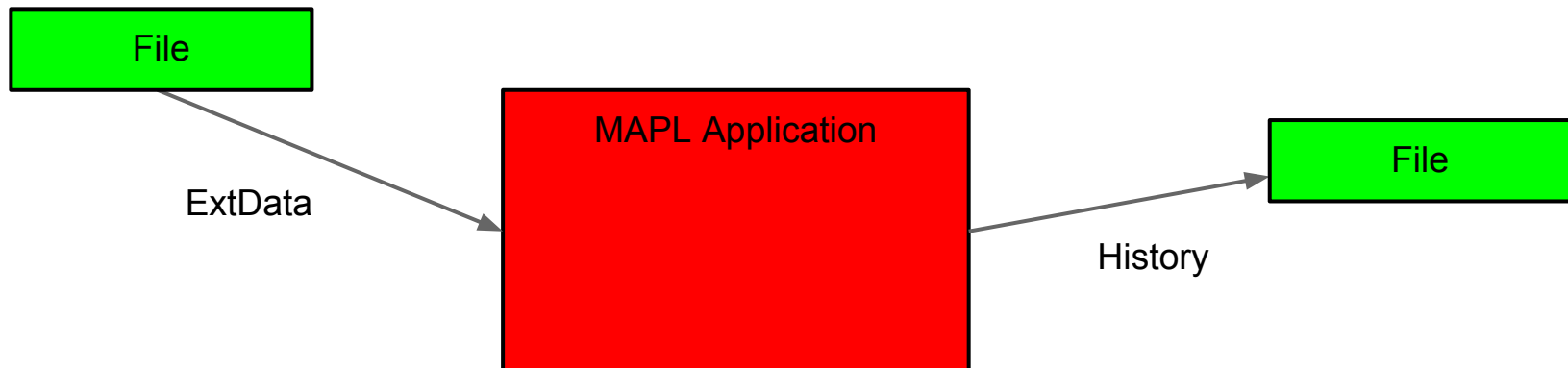- Doesn't work on collections going through a tile file

3D   2D

+
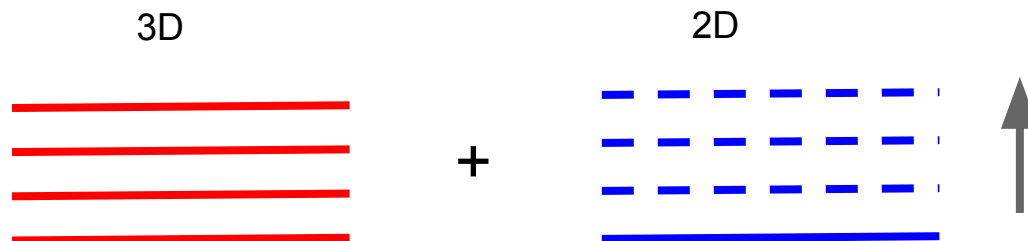
More on Scope Rules for Fields in Epxressions

```
geosgcm_prog.template:  '%y4%m2%d2_%h2%n2z.nc4',
geosgcm_prog.archive:   '%c/Y%y4',
geosgcm_prog.format:    'CFIO',
geosgcm_prog.frequency:  060000,
geosgcm_prog.resolution: 144 91,
geosgcm_prog.fields:    'U'             , 'DYN'           ,
                         'V'             , 'DYN'        , VV
                'sqrt(U*U+VV*VV)' , 'DYN'      'Wind_Magnitude'   ,
                '(T-273.15)*1.8+32.0' , 'DYN'    , 'TF' ,
                         'T^3'            , 'DYN',    'T3' ,
                ::
```
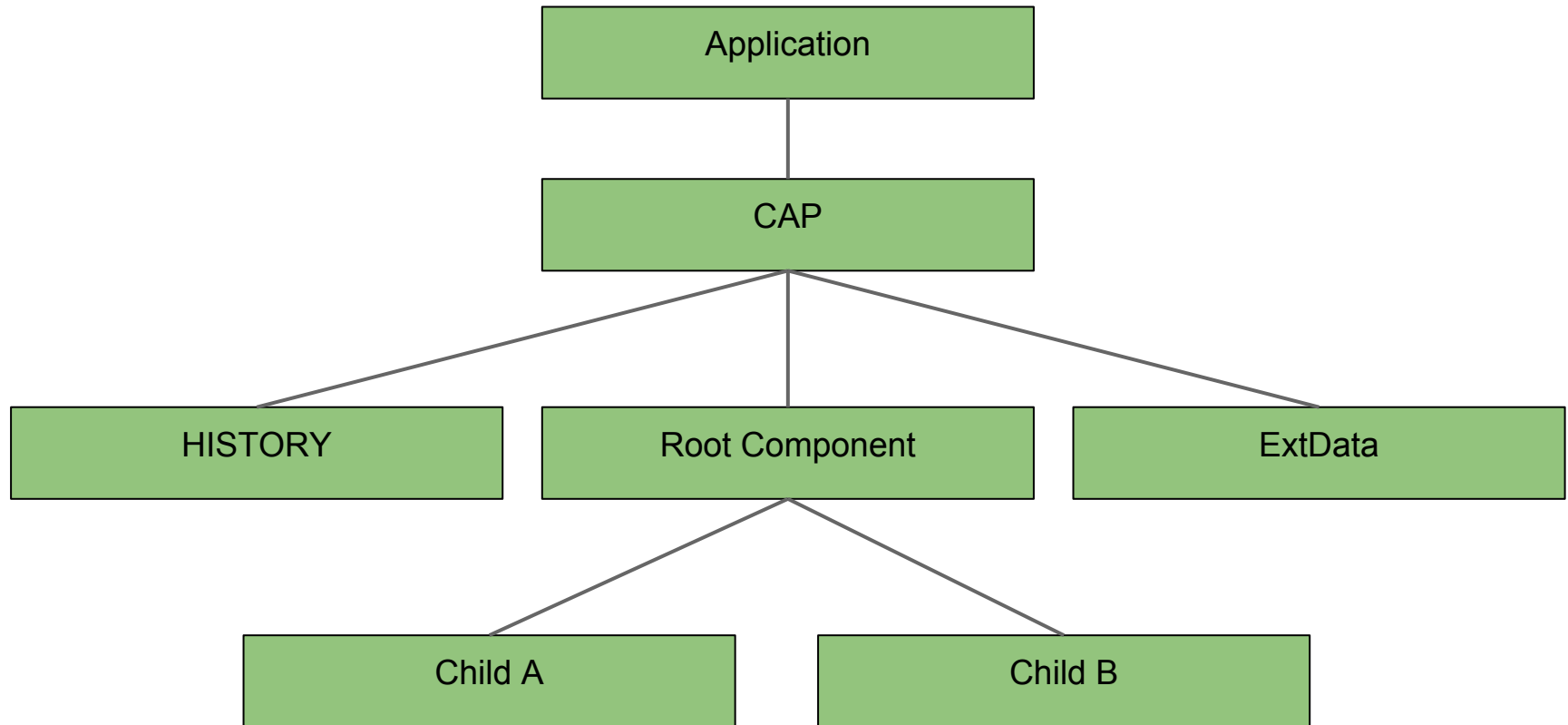
- The 2nd column provides another place to look for the fields in an expression if they are not in the collection.
- U and V are in the collection above so they can be used in any expression in it.
- I can use T in those expressions in the example collection because it is in the DYN component.
- Bottomline: a field in an expression must be in the collection or in the component following the expression.

ExtData Gridded Component

- The History gridded component connects export fields to files on disk in a MAPL application, running each step dumping out data to file based on the user input.
- ExtData is the converse, it provides a mechanism to connect import fields to an external file on disk in a MAPL application. It runs every step, ingesting data from files based on user input.
- Can create drivers for individual science components (RadApp tool for example)

Overview of a MAPL Application

How ExtData Works

CAP

At this point T has not been satisfied go to ExtData

GridComp A

Check for connectivity of T, assume no, check in parent

ExtData

Check if ExtData can fill a variable named T, if not abort

GridComp B

Check for connectivity of T, assume no, check in parent

GridComp C

Import T

Other components, maybe one of them fill fulfill T in GridComp C
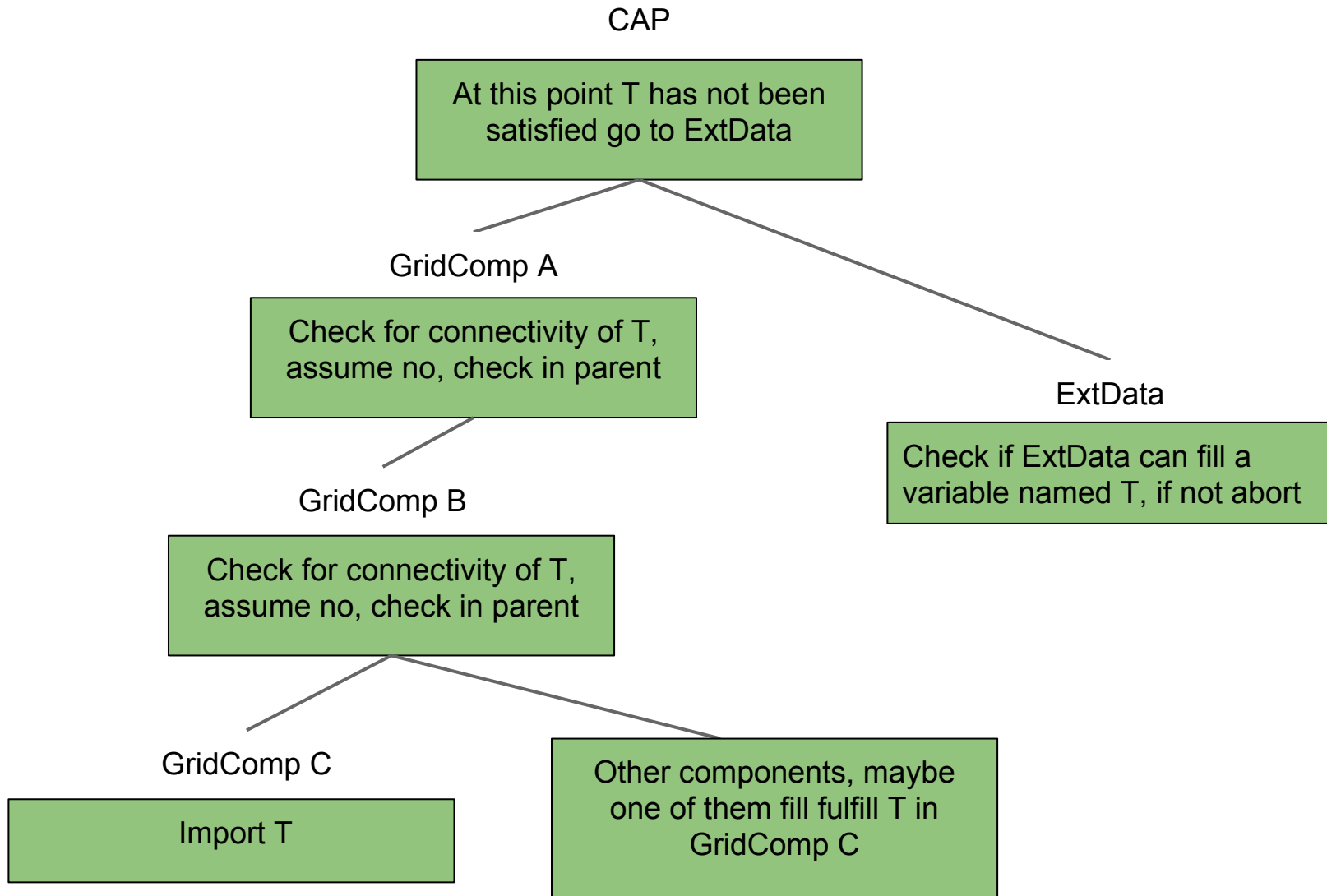
The ExtData RC file

- Think of the resource file as an announcement of what the file CAN provide.
- Must have a primary export section. Each entry connects a model field to a variable on file.
- Checks unfulfilled imports against RC file at initialization
- Can also specify: dimension of field, when to update it, whether it is a climatology, and how the data is distributed among files.
- Data need not be all on one file, very flexible.
- Data need not be on the same grid as the application. I/O library will interpolate

Primary Export Entry

- Import name - Name of the import to fill in the application.
- Units - Currently not used but must have something there
- Dimensional - xy for 2D, xyz for 3D
- Vertical Location - enter c for center, e for edge, if 2D enter c or e but will be ignored
- Climatology - enter Y if it is a climatology, otherwise enter N
- Refresh time - update once, every timestep, or at certain times.
- Offset factor - offset data by this number (0.0 for no offset)
- Scale factor - scale data by this number (1.0 for no scaling)
- Variable on File - name of variable on the file
- File template - grads style template for filename, can also enter /dev/null: constant to set the variable to a constant, if just /dev/null will be set to zero
- The next 3 entries are optional, but must specify all 3 if used. Provide advanced flexibility in describing how the data is distributed in the files for every step update, similar to reference time and frequency in history collection.

Refresh Options

- Update once: Enter '-', the field will only be updated the first time ExtData runs, applying current time to file template
- Updated a regular time, for example: %y4-%m2-%h2T%21:00:00 will update the field every day at 21z, applying current time to file template
- Enter 0 to update every step, then the field will be updated every step and ExtData will interplate to that time using the available data on file, can even interpolate between files.
- In this case ExtData will try to determine frequency of the files (single, monthly, daily, hourly, etc . . .) or user can use 3 optional arguments to specify this

Example of Continuous Updating With Hourly Files, 30 Minute Application Time

	20:00	20:30	21:00	21:30	22:00	22:30

Application TIme

Files: 20:00   21:00   22:00   22:30
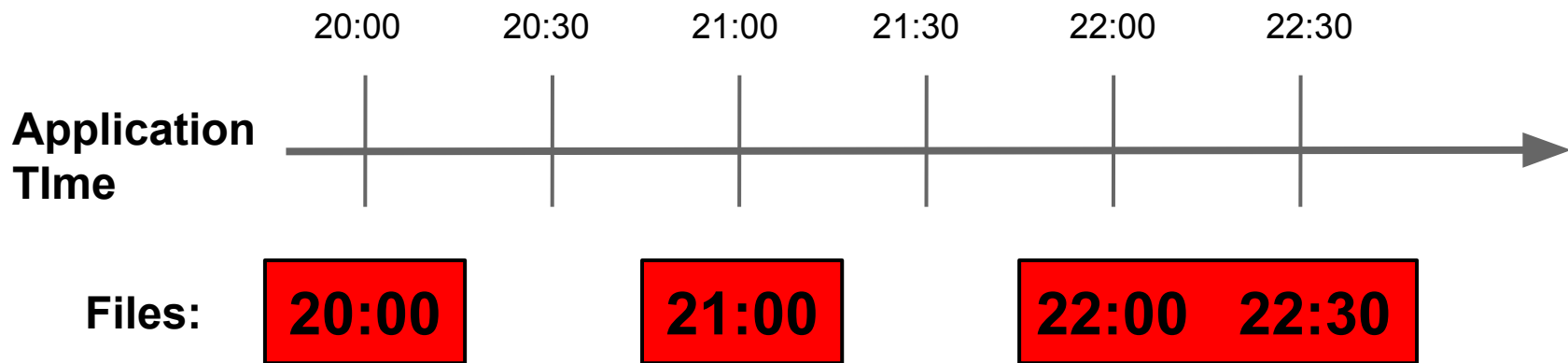
File Template: MyFile.%y4%m2%d2_%h2.nc4

Current Limitations

- Can't fulfill imports on a tile grid, only fields on the atmospheric grid or ocean for now.
- Uses MAPL_CFIO for I/O so limited to current capabilities in this module.
- No mechanism yet to break the resource file up for better maintainability.

Things not covered

- You can also specify derived fields based on primary fields, similar to History.

Running GEOS 5 on a GMAO desktop

What do you need?
- A desktop running a 64 bit OS
- /ford1/share and /ford1/local mounted
- Ask GMAO support if you don't have these

How do I get the model on my desktop?
- Set up a tunnel: https://progress.nccs.nasa.gov/trac/admin/wiki/QuickStart
- Modify your .ssh/config, upload your key, and ssh to cvsacl.nccs.nasa.gov to create the tunnel
- Check out from CVS as usual (make sure CVSROOT is set to :ext: user_name@ctunnel:/cvsroot/esma)

What you need to add to your .ssh/config:

```
Host cvsacl
        HostName cvsacl.nccs.nasa.gov
        LocalForward 51234 localhost:22223

Host ctunnel
        HostName localhost
        Port 51234
```

Desktop Continued . . .

Once you have checked out you might have to update a few files: g5_modules, gcm_setup, scm_setup, see wiki the right tag or contact SI team for the tag.

Build as normal:
- make sure ESMADIR is set
- source $ESMADIR/src/g5_modules to set compiler and baselibs (currently ifort 13.1.1.163 and openmpi 1.6.5)
- make -jn pinstall

   What is feasible to run?
   - single column on any desktop, use scm_setup as normal
   - full model at 144x91 or c48 with Reynolds ocean if you have newer 8 core, 8 GB memory desktop. Use gcm_setup, add exit to script after the GEOSgcm.x call, no post processing yet
   - One day at c48, standard history, no gocart, 1x6 layout took ~9 minutes and had a highwater memory mark of ~ 7 GB on my desktop
   - Currently have boundary conditions for c48 and 144x91 with Reynolds ocean on /ford1. Will try to get c24 and 72x46 boundary conditions

Additional Information

Running the model on a GMAO desktop:

http://geos5.org/wiki/index.php?title=Running_GEOS_5_on_a_GMAO_Desktop

Using the ExtData component:

http://geos5.org/wiki/index.php?title=Using_the_ExtData_component

Using the parser capability in History:

http://geos5.org/wiki/index.php?title=Ganymed_1.0_User%27s_Guide#Outputting_Derived_Fields