2024 Planetary Science Technology Symposium

Developments in GRAM for Planetary Missions

Soumyo Dutta, Kunio Sayanagi; NASA Langley Research Center
Hilary Justh, Paul Bremner; NASA Marshall Space Flight Center
Jim Hoffman, Justin Garland; Analytical Mechanics Associates
Lee Burns; Amentum Space Exploration Group

October 24, 2024

What is GRAM?

- Global Reference Atmospheric Model (GRAM)
  - Engineering-oriented atmospheric model
  - Based on data derived from planetary global circulation models and mission data
  - Not a forecast model
- Available for Venus, Earth, Mars, Jupiter, Titan, Uranus, and Neptune (currently missing Saturn)
- The GRAMs provide mean values and dispersions for any point in the atmosphere as well as seasonal, diurnal, geographic, and altitude variations
- Inputs include local solar time, latitude, longitude, and solar longitude (season)
- Outputs include atmospheric density, temperature, pressure, winds, chemical composition, and (for Mars-GRAM) radiative fluxes along a user-defined path
- First developed in 1974 and have been continually updated and expanded
- SMD has been supporting upgrades to the GRAM software in the last 5 years

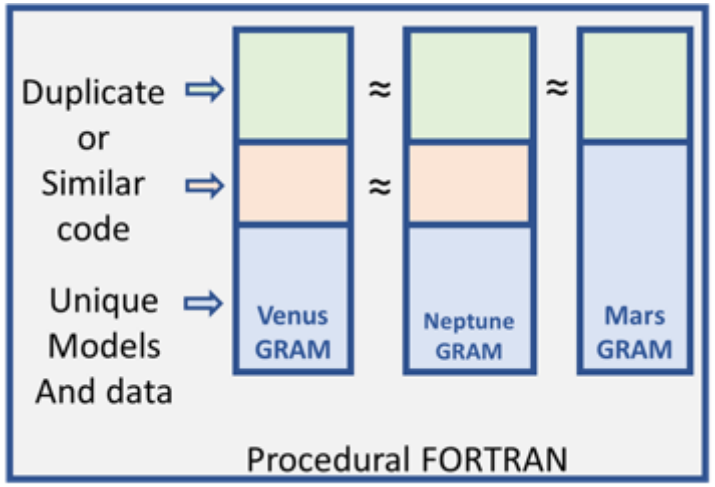
Outline

- GRAM Code
- APIs
- Upcoming GRAM Updates
  - Uranus
  - Neptune
  - Saturn
  - Venus
  - Mars
- Development Plans

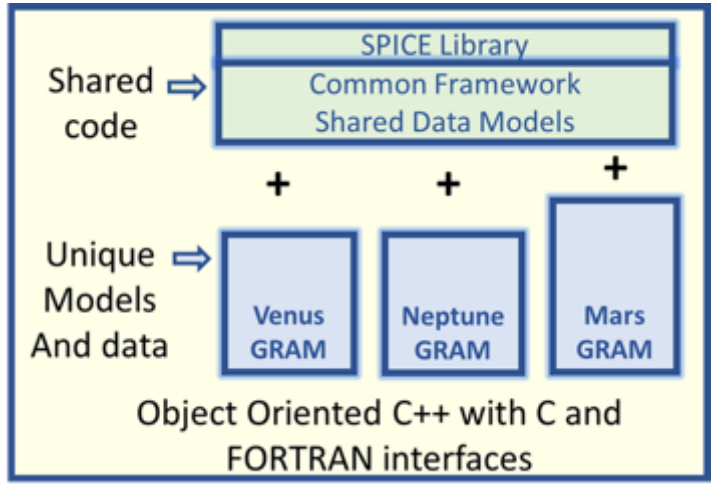
GRAM Code

- GRAM Suite: C++-based common code
  - Reduces duplicated code
  - Ensures consistent constants across all GRAMs
  - Provides consistent user interface
  - Simplifies bug fixes
  - Streamlines the interface with trajectory codes
- Incorporated connections to SPICE for ephemeris data
- All future GRAM releases will be in GRAM Suite
  - Current Version: GRAM 2.0 (2023)
  - New Release: GRAM 2.1 (October 2024)
- Software available via the NASA Software Catalog

Legacy GRAMs



GRAM Suite



APIs

- Includes C and Fortran interfaces which can be incorporated in a trajectory code
- Python API (available in version 2.0): GRAMpy
- MATLAB API (available in version 2.1): GRAMmi

First, import the gram module:

```
In [1]: from GRAMpy import gram
```

Import Module

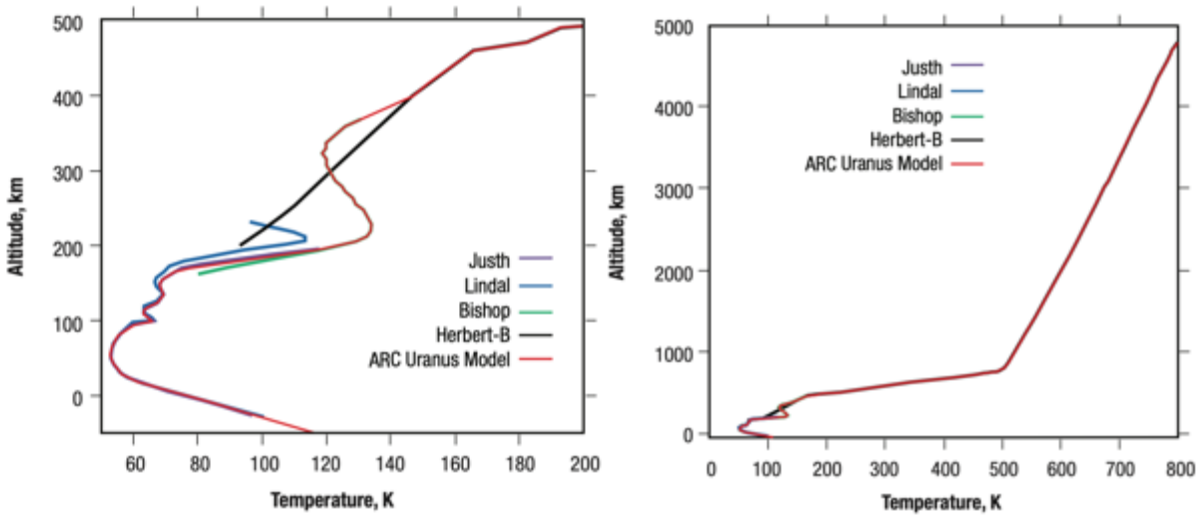
Next, we are going to get some data for a point in the Mars atmosphere on a specific day.

```
In [2]: inputParameters = gram.MarsInputParameters()

        # Set the path to the Mars data folder- this is important!
        inputParameters.dataPath = "../Mars/data/"

        # Load SPICE data
        gram.NamelistReader().tryGetSpicePath(inputParameters)
        gram.SpiceLoader().setInputParameters(inputParameters)

        # Create Mars atmosphere
        mars = gram.MarsAtmosphere()
        mars.setInputParameters(inputParameters)

        # Set model options
        mars.setMapYear(0)
        mars.setMGCMDustLevels(2.0, 0.0, 0.0)
        mars.setPerturbationScales(1.5)
        mars.setMinRelativeStepSize(0.5)
        mars.setSeed(1001)

        # Set simulation start time
        time = gram.GramTime()
        time.setStartTime(2020, 11, 2, 0, 0, 0, gram.UTC, gram.ERT)
        mars.setStartTime(time)

        # Set simulation position
        position = gram.Position()
        position.height = 2
        position.latitude = 22
        position.longitude = 48
        position.elapsedTime = 100
        position.isPlanetoCentric = True
        mars.setPosition(position)

        # Update the model based on the current time and position
        mars.update()
```

Set Inputs

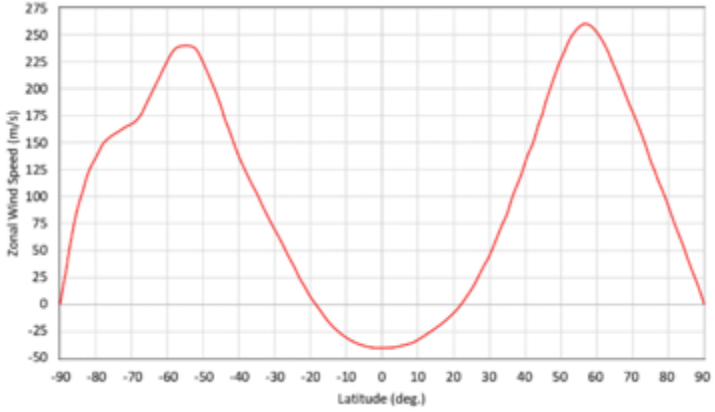
Call GRAM

Upcoming Updates: Uranus

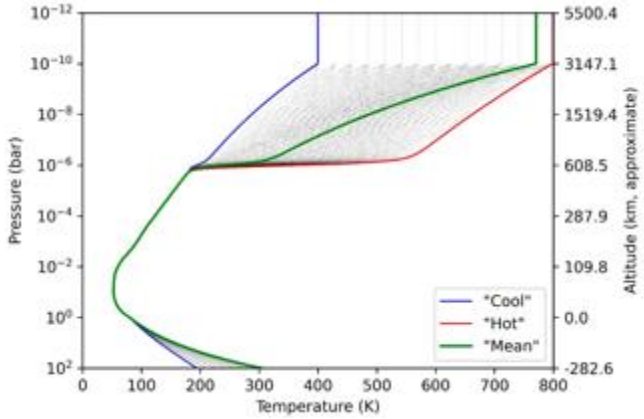


Previous Uranus GRAM

Credit: NASA/TM -20210017250



Cloud Top Zonal Winds vs. Latitude (Sromovsky et al. 2015)



Updated Uranus GRAM Atmosphere

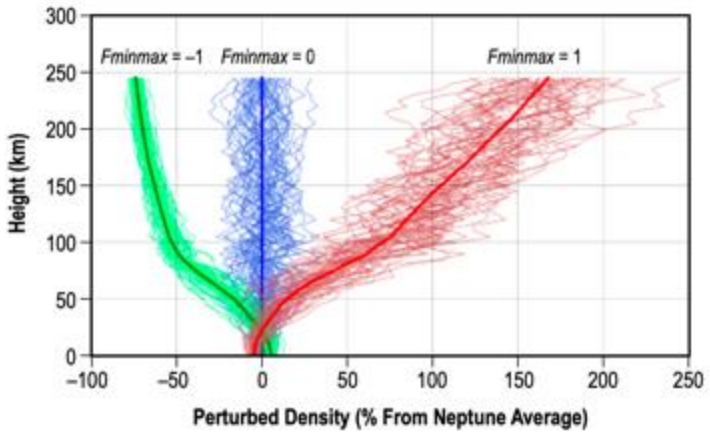
- Previous Uranus GRAM based on Voyager analysis papers contemporary with the 1986 flyby (Lindal et al. 1987; Herbert et al. 1987; Bishop et al.1990)
- New Uranus GRAM includes updated thermospheric temperature bounds (Teanby et al. 2022); ground-based observations (Melin et al. 2019); data from Spitzer Space Telescope (Orton et al. 2014)
- Includes zonal winds and estimated vertical wind shears
- Uranus observations from late-breaking missions (e.g. James Webb) will continuously be incorporated

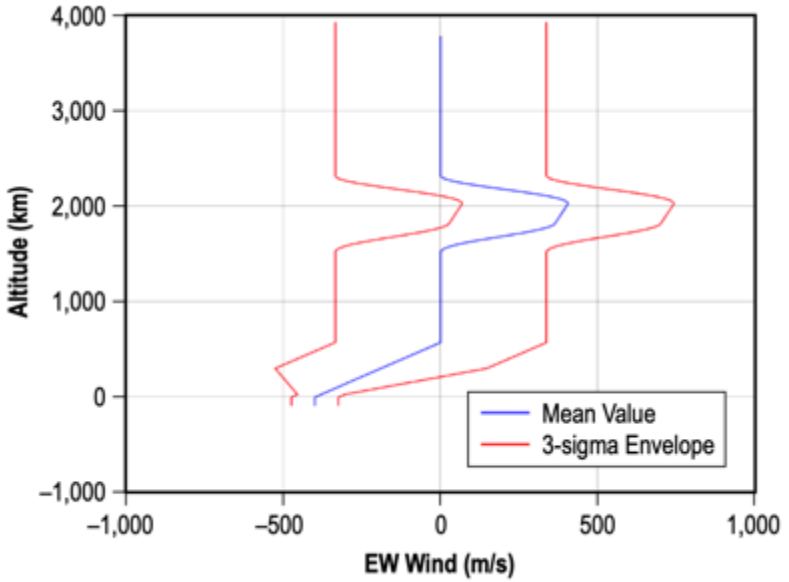
Upcoming Release

Upcoming Updates: Neptune

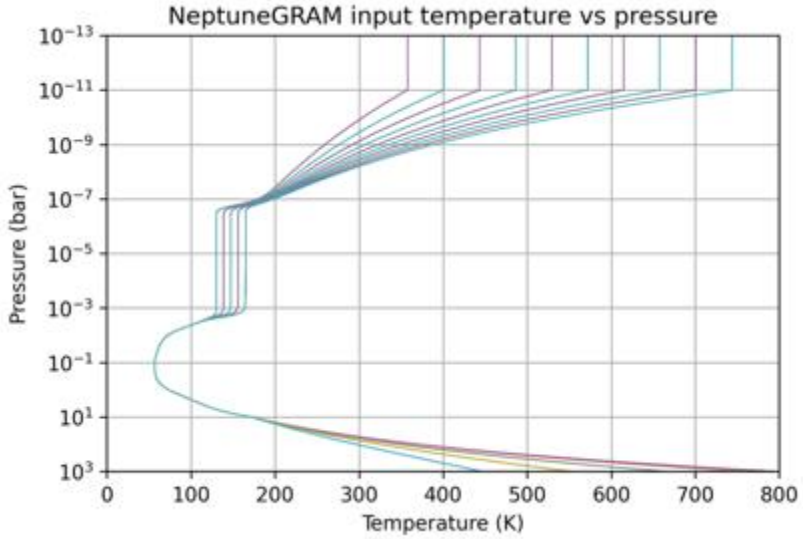
- Previous Neptune GRAM also based on analysis of data contemporary with the Voyager 2 flyby – 1989
- Neptune GRAM update still underway. Will incorporate newer reanalysis, update zonal and meridional winds estimates

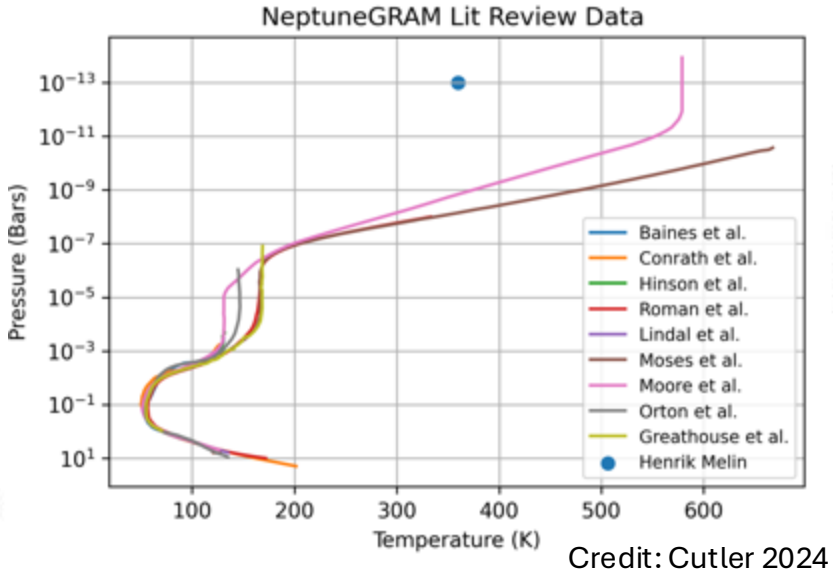
Current GRAM



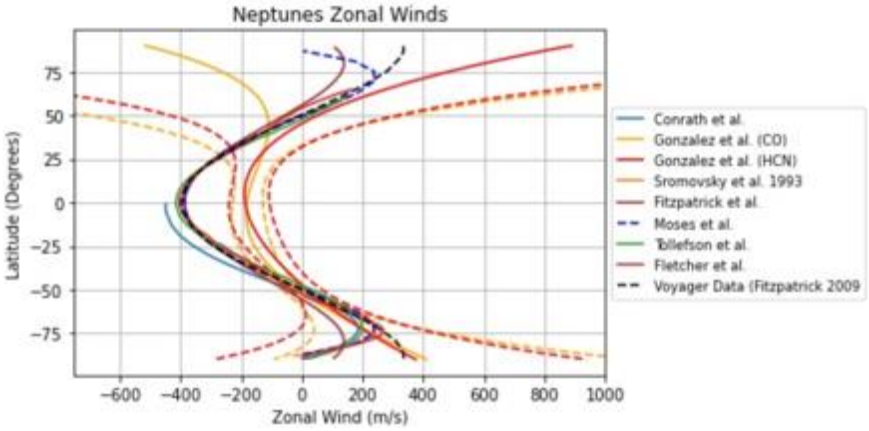


Potential Updates





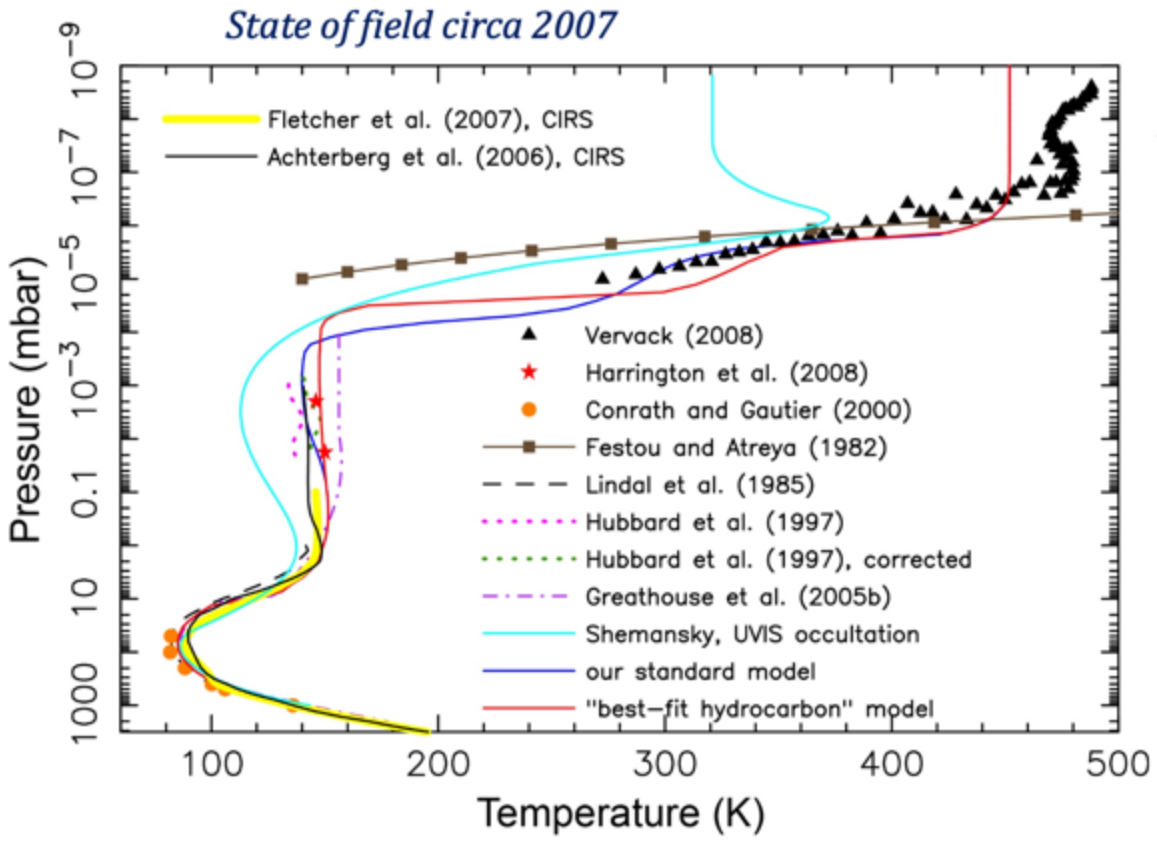
Credit: Cutler 2024



Credit: Cutler 2024

Upcoming Updates: Saturn

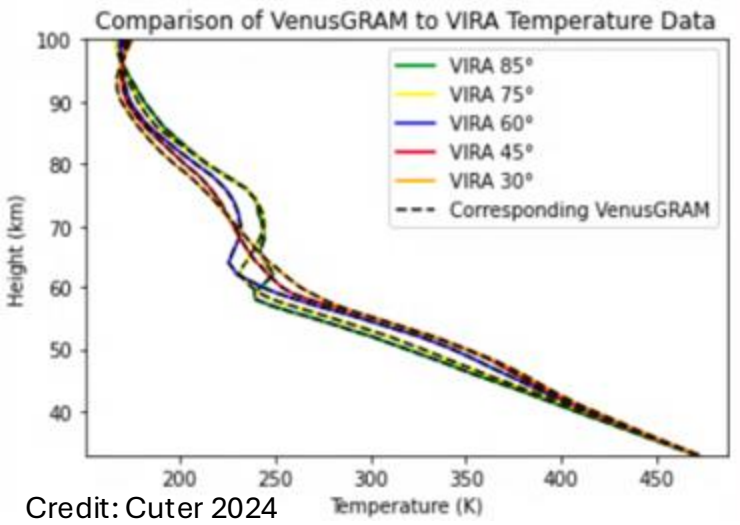
- No Saturn GRAM exists but is valuable for the next New Frontiers round (~2026)
- Goal: 1-D (in altitude) profile of atmospheric properties similar to v1 Uranus, Neptune, and Jupiter GRAM



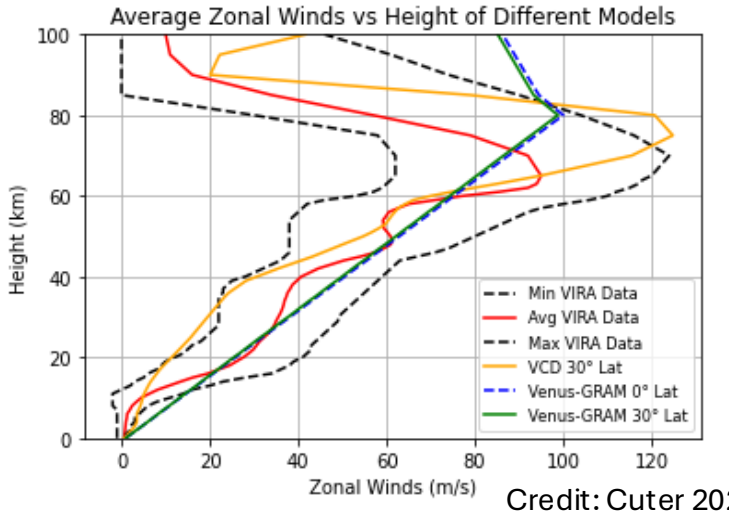
Credit: Moore

Upcoming Updates: Venus

Current GRAM

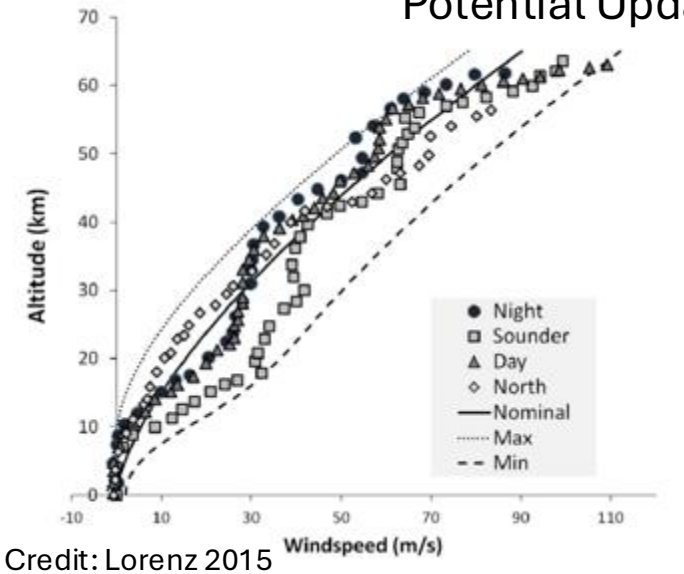


Credit: Cuter 2024

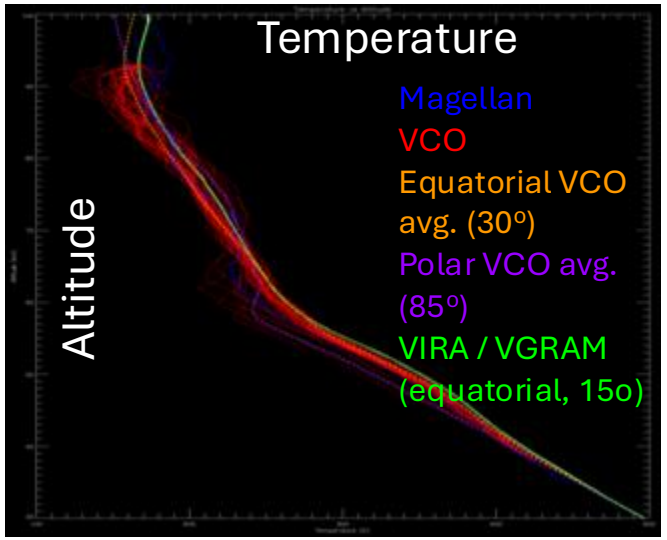


Credit: Cuter 2024

Potential Updates



Credit: Lorenz 2015



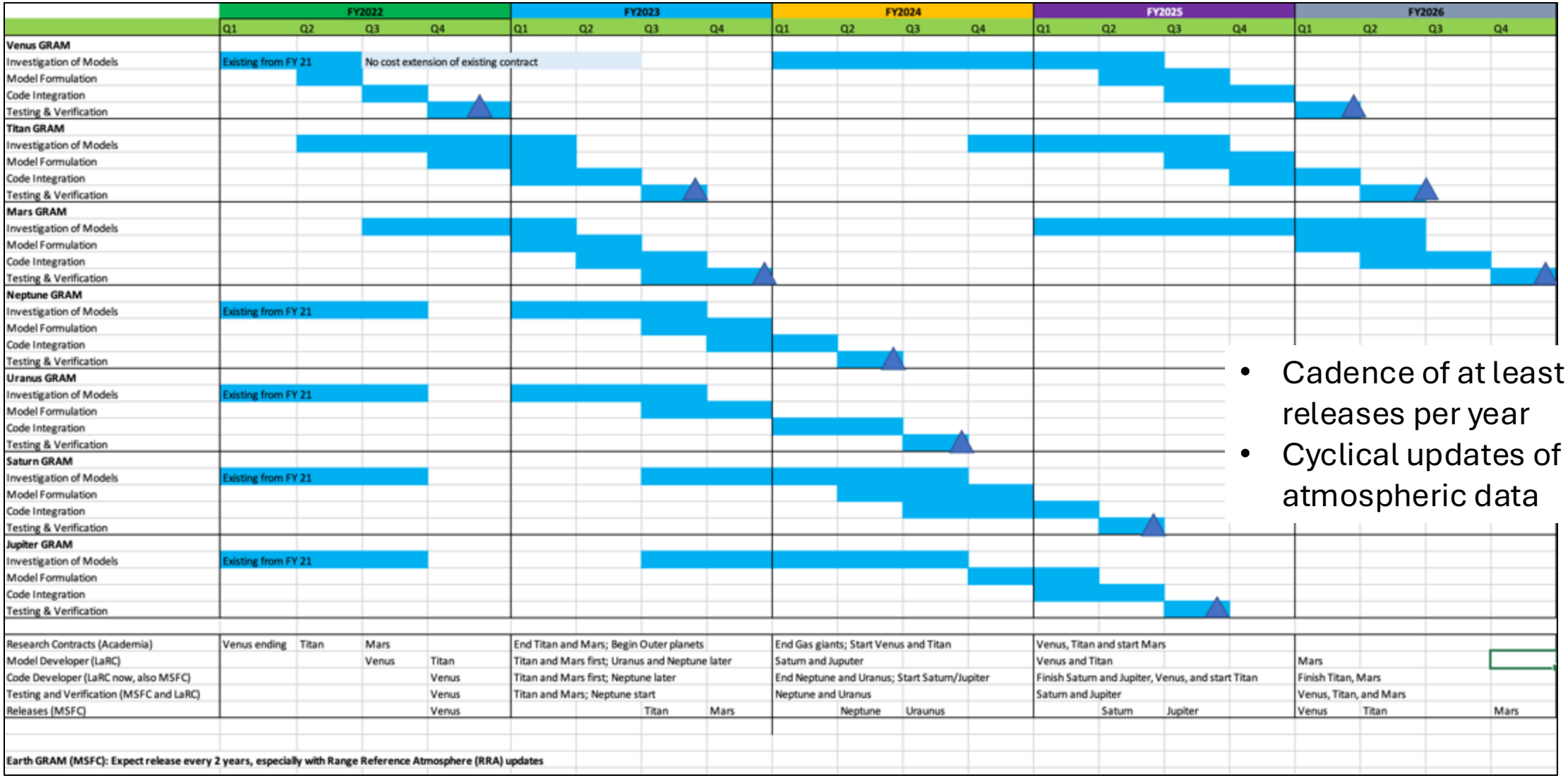
Credit: McCabe 2022

- Venus GRAM based on the Venus International Reference Atmosphere (VIRA) developed in the 1980's
  - Temperature, density, pressure identical to VIRA
  - Winds and composition has differences
- Add Magellan and Akatsuki datasets to augment VIRA
- Update winds using reanalysis data for modern entry missions
- Model wind turbulence
- Higher fidelity gravity model – this will be extensible to other planetary GRAMs too

Upcoming Updates: Mars

- Mars GRAM currently based on Global Circulation Model data runs for various solar longitude and dust scenarios
- Mars-GRAM Topography Data Upgrades
  - Added higher resolution Mars Orbiting Laser Altimeter dataset
  - Users can select between resolutions of 2, 4, 16, 32, and 64 pixels/degree
- Input Data Updates
  - New lower atmosphere data from Mars General Circulation Model (MGCM)
  - Thermosphere data from the Mars Global Ionosphere-Thermosphere Model (M-GITM)

Development Plans



- Cadence of at least two releases per year
- Cyclical updates of atmospheric data

GRAM Team

NASA Langley Research Center

Project Manager
Soumyo Dutta

Project Scientist
Kunio Sayanagi

Code Architect
James Hoffman
Analytical Mechanics Associates

Implementation Expert
John Aguirre
Analytical Mechanics Associates

Dataset Development
Justin Garland
Analytical Mechanics Associates

NASA Marshall Space Flight Center

Planetary Atmosphere Specialist
Hilary Justh

Earth-GRAM Developer
Patrick White

Dataset Development
Lee Burns
Amentum Space Exploration Group

C++ and Data Implementation
Paul Bremner

Corey Walker
Jacobs Space Exploration Group

Summary

- GRAM is an engineering-oriented atmospheric model that provides data for most planetary atmospheric bodies of interest
- GRAM code has been consolidated into C++ based GRAM Suite for ease of new data integration, code updates, and bug fixes
- APIs for C and Fortran programs have been provided. Recently, APIs for Python and MATLAB have also been added
- Updates planned for Uranus, Neptune, Saturn, Venus, and Mars
- Development plan involves approximately 2 releases per year with steady updates and incorporation of new data for planetary models
- Holds a yearly GRAM workshop virtually for customers (email hilary.l.justh@nasa.gov if you are interested in attending)