

GMAT 2014a Release Process

This page is used to track the process and progress of the R2014a release.

Table of contents

- Table of contents
 - Release Process Work Before next Cycle for R2015a
 - Proposed Roles For Next Release for R2015a
- Tasks
 - Early Tasks
 - For QA Complete (April 21)
 - For Visual Freeze (April 30)
 - For Code Freeze (April 30)
 - For App Freeze (May 5)
 - Testing of Release Candidate 1 (May 5-May 9)
 - Stage Release (May 13)
 - Release Day (May 14)
 - Post-Release
- Notes for postmortem
- Compatibility changes
- Creating an RC
- Running the script test system
- Issues for RC2

Release Process Work Before next Cycle for R2015a

- NTR needs to go in months (3-6) earlier
- Define major roles and assign them to individuals before next release
 - Proposed roles are shown below.
 - Ensure those in major roles are reachable and preferably in the office during critical stages like creating and testing RCs.
- Consider merging to production and test this before next release process
- Automate installer packaging
- Consider sanity check from mission users' scripts
- Feature Freeze/QA Complete/Beta Test must happen earlier (month before code/App Freeze)
 - We need to baseline the test results well before release process starts. Probably at Feature Freeze/QA Freeze
 - Consider allowing GATS to produce multiple reports from one test run
 - Consider feature freeze month before
 - QA Complete needs to happen much earlier (month?) probably with Feature Freeze

Proposed Roles For Next Release for R2015a

- **Release Manager (RM):** "owns" release, initiates process, creates tracking page, sends daily status updates, tracks issues to completion, makes sure everyone gets their stuff done, brings decisions to CCB/team, maintains tracking page, collects lessons learned, documents process improvements for next release
- **Build Manager (BM):** controls build system, creates RCs, sends RC availability announcements
- **Test Managers (TM):** control GUI/script testing for each RC

Target Date: May 14, 2014

Tasks

All dates are referenced to 12:00 noon EDT.

For example, a deadline of March 15 should be interpreted as March 15, 12:00 noon EDT.

Early Tasks

These are long-lead early tasks that can be completed before the detailed release cycle.

Use "(/)" for the checkmark (✓) and "(x)" for the cross (✗)

Task	Who	Status	Internal Status	Notes
Review Previous Build Release R2013b Process Notes	All	✓	✓	
Get updated legal statement/license	SPH	✓	✓	Needed by Code Freeze.
Update sample scripts	SPH	✓	✓	Needed by App Freeze. <ul style="list-style-type: none"> • Write examples that demonstrate new functionality • Clean up all errors and warnings • Remove deprecated fields
Write draft Release Notes	JJKP SPH	✓	✓	Needed by App Freeze. See Writing Release Notes
Update standard descriptive text	SPH	✓	✓	Needed by App Freeze. Will be used in User Guide, websites, release announcement.
Update version number on User Guide cover page	JJKP	✓	✓	Needed by App Freeze. Contact Katy Gammage or Mary Hrybyk-Keith. Next time: integrate sejda-console to do this automatically.
Update info on public-facing websites	JJKP	✓	N/A	Needed by Release Day. See the list of sites . Update this list as well, if necessary. <i>Updated major sites before announcement. Minor ones can be updated a bit later.</i>
Update release announcement contacts	SPH	N/A	N/A	Needed by Release Day. Located here .
Put in order for additional icons	SPH	N/A	N/A	(if necessary) Needed by Visual Freeze.

For QA Complete (April 21)

QA complete means that all known bugs are being tracked, all bug fixes are verified and documented, and the release process can begin.

Use "(/)" for the checkmark (✓) and "(x)" for the cross (✗)

Task	Who	Status	Notes
Verify that all known bugs are checked into JIRA	(All)	RQ: ✓	RQ: I'm done with this
Complete all JIRA verifications	(All)	RQ: ✓	RQ: I'm done with this
Complete QA wrap-up tasks	DSC, SPH, JJKP, RQ	RQ: ✓	RQ: I'm done with this
Address all JIRA tickets awaiting feedback	(All)	RQ: ✓	RQ: I'm done with this

For Visual Freeze (April 30)

Visual Freeze finalizes all graphical changes to the software, so that screenshots, documentation, and TestComplete can be updated.

Use "(/)" for the checkmark (✓) and "(x)" for the cross (✗)

Task	Who	Status	Internal Status	Notes
Update About panel	LOJ	✓	✓	<ol style="list-style-type: none"> 1. Update release number (i.e. R2014a). 2. Add any new libraries used. 3. Update an links.
Update splash screen	TGG	✓	✓	<ol style="list-style-type: none"> 1. Add new contributors 2. Remove contributors who did not contribute to this release 3. Design updates 4. Update SplashScreen.psd in GmatDevelopment\moredata\graphics\splash 5. Use GIMP to save a flattened TIF file and overwrite splash screen in GmatDevelopment\application\data\graphics\splash.
Update icons	TGG	✓	✓	<p>If there are any updates, additions, deletions.</p> <p>Including GMATIcon for Welcome Page</p>
Update gmat_startup_file.txt	JJKP	✓	✓	<ul style="list-style-type: none"> • Update formatting, comments • Add any additional plugins • Switch to release configuration (comment out internal plugins) <p><i>No new plugins added. Commented out internal plugins.</i></p>
Switch to release configuration in script test system	JJKP	✓	✓	<ul style="list-style-type: none"> • Create new rundef.R2014aInternal.template.m and rundef.R2014aPublic.template.m. • Switch autorun.m to use appropriate one. <p><i>4/30: switched to R2014aInternal</i></p> <p><i>7/17: tested public release using R2014aPublic</i></p>
Complete visual updates	(All)	✓	✓	<p><i>4/30: JJKP: still waiting on final updates to contributors list from KARI</i></p> <p><i>5/2: TGG: final update committed</i></p>

For Code Freeze (April 30)

Code Freeze is a freeze on the software itself before final testing.

Use "(/)" for the checkmark (✓) and "(x)" for the cross (✗)

Task	Who	Status	Internal Status	Notes
Update EOP files	WCS	✓	✓	Update eopc04_08.62-now and run smoke tests.
Update files w/ updated legal statement	LOJ	✓	✓	<ul style="list-style-type: none"> • Readme file • Source code files
Update license text	LOJ	✓	✓	Update application/License.txt file.
Final bug fixes	(All)	✓	✓	
Mark all open bugs as Affects: current release	SPH	✓	✓	

For this release: Should we branch the repo here, instead of after release? We need to allow people to continue working on unrelated items while release work is ongoing.{warning}

(DJC here) A better approach going forward would be to merge master into production, and switch the test system to the production

branch. This probably ought to be done right before building RC1. The dev team would have 2 options then:

1. Checkout branches based on what they are working on (i.e. checkout production for bug fixes for the release, and their current clone of master – every one is cloning for new work, right? – for ongoing work)
2. Alternatively, make bug fixes in master and cherry-pick merge changes into production as bugs are fixed

The former is the better approach, IMO, because it ensures that the bug fix is made on the current code base for the release.

For App Freeze (May 5)

App Freeze is a freeze on all application bundle files beyond data and code. This includes documentation, sample scripts, stuff in the `extras` folder, etc.

Use "(/)" for the checkmark (✓) and "(x)" for the cross (✗)

Task	Who	Status	Notes
Update <code>README.txt</code>	JJKP	✓	Update with major release highlights.
Update <code>extras</code> folder	JJKP	✓	Notepad++ syntax coloring file
Update PDF files in docs folder	SPH/DJC	DJC: ✓ SPH: ✓	<ul style="list-style-type: none"> • Update watermark: "Draft for Release R2013b" • Put into application/docs folder, and individual doc source folders
Gather list of compatibility changes since last release	SPH	✓	<ul style="list-style-type: none"> • Deprecated fields • Removed & disabled fields • Anything a user would need to know to make R2013b scripts compatible with this release.
Update Release Notes	SPH	✓	
Update screenshots in User Guide	(Feature leads)	JJKP: ✓ SPH: ✓ RQ: ✓ DSQ: ✓	
Test User Guide instructions & code	(Feature leads)	RQ: ✓ DSQ: ✓ JJKP: ✓ SPH: ✓	<ul style="list-style-type: none"> • Tutorials • Common Tasks • script snippets • reference page examples
Update Windows installer package	JJKP	✓	<i>No updates necessary.</i>
Update links in <code>GMAT.ini</code>	JJKP	✓	<ul style="list-style-type: none"> • Help links • Welcome page links
Update link tests in <code>TestComplete</code>	TR	✓	<ul style="list-style-type: none"> • Help buttons • Welcome Page links • Help menu links

Testing of Release Candidate 1 (May 5-May 9)

This will start with the 2014-05-05 daily build. **Repeat this phase until tests check out.** Steve will make the call.

Use "(/)" for the checkmark (✓) and "(x)" for the cross (✗)

Task	Who	Status	Notes
Update README.txt	JJKP	RC1: ✓ RC2-RC5: N/A	<i>Next time: this should probably be "Update Release Notes" to add outstanding bugs, etc.</i>
Build Windows installer	JJKP (backup: TGG)	RC1: ✓ RC2: ✓ RC2-2: ✓ RC3: ✓ RC4: ✓ RC5: ✓	Version string: R2014a-rc# <i>Next time: build manager should do this</i>
Bundle Windows zip	JJKP (backup: TGG)	RC1: ✓ RC2: ✓ RC2-2: ✓ RC3: ✓ RC4: ✓ RC5: ✓	Version string: R2014a-rc# <i>Next time: build manager should do this</i>
Run TestComplete smoke tests	TR	RC1: ✓	These are tests on the packaged versions of GMAT: the installer and the zip bundle.
Run TestComplete system test missions	TR	RC1: ✓	These are tests on the packaged versions of GMAT: the installer and the zip bundle.
Run script test system	JJKP (backup: TGG)	RC1: ✓ RC2: ✓ RC2-2: ✓ RC3: ✓ RC4: ✓ RC5: ✓	Run the internal installer tests on T4 and the public installer tests on Joel's machine. Run .zip bundle tests afterwards on same build to compare.
Test all sample scripts	SPH	RC2: ✓ RC4: ✓	At a minimum these need to be run individually by hand. I ran them by adding the folder, and they run so fast I missed some pretty big problems. Ideally, these should all be in script regression tests. Many but not all already are regression tested. Or, we should add them to GUI regression system.

Run TestComplete full regression tests	TR	RC4:	(For final RC only)
		RC5: 	

Notes

- While this cycle is ongoing is a good time to do wiki updates and cleanup.

Stage Release (May 13)

This is a soft release, putting all the files in place and updating information. Then on release day, we only need to send the announcements.

Use "(/)" for the checkmark () and "(x)" for the cross ()

Task	Who	Status	Notes
Bundle source code	DJC*		<ol style="list-style-type: none"> 1. Export the trunk code from svn that is used for the release build when that build is started 2. Wait for a go/nogo call from testing on the build 3. Archive the following folder trees into a zip file: src, plugins, build
Bundle data	DJC*		<ol style="list-style-type: none"> 1. Use the same export as used for the source bundle 2. Wait for a go/nogo call from testing on the build 3. Archive the following folder tree into a zip file: application/data
Upload source bundle to SourceForge	JJKP		<ol style="list-style-type: none"> 1. Move the zip file to SF 2. Mark as "staged" 3. Download the upload and check it
Upload data bundle to SourceForge	JJKP		<ol style="list-style-type: none"> 1. Move the zip file to SF 2. Mark as "staged" 3. Download the upload and check it
Upload Windows installer to SourceForge	JJKP		Download, install, and run after uploading.
Upload Windows zip to SourceForge	JJKP		Download and run after uploading.
Post README.rst.txt on SourceForge	JJKP		Update for new release
Upload docs to documentation site	JJKP		<ol style="list-style-type: none"> 1. Upload using SSH to http://gmat.sourceforge.net/docs/ 2. Update HTML with new section 3. Move "latest" pointer to new folder
Post internal & public release files to MESA network	JJKP		<i>2014-05-22: posted internal files only; public files are in build\VS2010_build\VR2014a</i>
Branch and tag repositories	DJC		At least tag the test system; consider branching also if the burden on the repo is low.
Make SourceForge repository backup	JJKP		<p>Follow SourceForge's instructions.</p> <p>Perform the backup on the Linode server and download the resultant .zip file, since the local network blocks rsync.</p>

* DJC can only do steps 1-3 here because of contractual constraints

Release Day (May 14)

Use "(/)" for the checkmark () and "(x)" for the cross ()

Task	Who	Status	Notes

Make files visible on SourceForge	JJKP	✓	07/21: Made visible at time of posting, as "soft" release.
Send out release announcement	SPH	✓	<ul style="list-style-type: none"> Mention GSFC in the announcement Don't include large attachments.
Post release announcement on SourceForge	SPH*	✓	
Post release announcement on GMAT Blog	JJKP	✓	
Mark as released in JIRA	SPH	✓	
Migrate code to SourceForge repository	JJKP	✓	Do we migrate to git at the same time?

This needs to be assigned to a different person; DJC cannot post to SF

Post-Release

Use "(/)" for the checkmark (✓).

Task	Who	Status	Notes
Party!	(All)	✓	Plan for week of June 19
Conduct postmortem review	SPH	✓	This includes gathering feedback, holding the postmortem meeting, and documenting the results.
Submit GSFC Metrics Summary	WCS	✓	The link to submit the metrics is invalid.No action is required for this item. It was decided on 09/25/14's GMAT meeting to check tasks that do not require any action or are inactive.
Submit NTR for next release	SPH	✓	Per Hughes: The NTR cannot be written until all features are finalized and it is too early to do that.

Notes for postmortem

Please add your notes to the [R2014a Lessons Learned](#) document.

Compatibility changes

This is a list of compatibility changes since R2013c that need to be captured in the release notes.

- ?

See [Writing Release Notes](#)

Creating an RC

1. Create the build
 - a. Log into gs580w-gmat-t4 as "gs580w-gmat-t4\builduser". The credentials are on the network drive, in the Infrastructure folder.
 - b. Start Task Scheduler.
 - c. [RC1 only] Disable the "GMAT Daily Build" task, so it doesn't run automatically during the RC cycle (this can make things overly confusing).
 - d. Manually run the "GMAT Daily Build" task.
2. Create the bundles
 - a. On your local system, navigate to GmatDevelopment\build\install\windows-nsis. Note that you do not need to pull files down from the GIT repository; this process will pull files from the remote build and create the packages in your local directory. There's a README.txt file there that explains things.
 - b. Open a MinGW Shell in this directory.
 - c. Run "make VERSION="R2014a-rc#", where "#" is the number of the RC you're creating. This will create four packages in the current directory: A .zip and a .exe file for both the internal and public versions. Note: to create only an internal version, run "make internal VERSION="R2014a-rc#".

- d. Copy the four package files to the network: \\mesa-file\595\GMAT\Builds\windows\VS2010_build\R2014a
 - e. To clean everything up afterwards, run "make clean".
3. Run the script test system on the internal installer package. See [Running the script test system](#), below.

Running the script test system

In the steps below, "#" is the RC number and "<config>" is "internal" or "public".

1. Log into gs580w-gmat-t4 as "gs580w-gmat-t4\testuser". The credentials are on the network drive, in the Infrastructure folder.
2. [RC1 only] Start Task Scheduler and disable the "Run Script Test" task. You'll need to run the RC tests manually.
3. Uninstall all previous versions of GMAT.
4. Install the version being tested.
 - a. Check "Enable MATLAB interface".
 - b. Leave "Associate file types" unchecked.
5. Update the test system repo (located at C:\Users\testuser\Documents\GmatTest)
6. In MATLAB (64-bit):
 - a. Run:


```
>> cd C:\Users\testuser\Documents\GmatTest\bin
```
 - b. Run (replace # with RC number and <config> with "internal" or "public"):


```
>> diary('..\log\R2014a-rc#-<config>.log')
```
 - c. Run (this command copies application files such as SPICE kernels needed for regression tests into the new installation of GMAT):


```
>> preparegmat('C:\Path\To\GMAT')
```
 - d. In C:\Users\testuser\Documents\GmatTest\bin, copy the appropriate template (autorundef.R2014aInternal.template.m or autorundef.R2014aPublic.template.m) to a new name (such as autorundef.R2014aRc1Internal.m) and fill in the values:
 - i. RunDef.Build: "R2014a-rc#-<config>"
 - ii. RunDef.GmatExe: path to installed GMAT.exe
 - iii. RunDef.RegressionBuild: last build date if testing RC1 (look in GmatTest\output for latest folder, such as 2013-07-31), otherwise, "R2014a-rc#-<config>"
 - e. Run:


```
>> gmattest <name of rundef.m> (e.g. autorundef.R2014aRc1Internal.m)
```
 - f. Wait until run completes (usually 8+ hours)
 - g. Run:


```
>> diary('off')
```
7. In a Windows cmd window, post results to network using robocopy:


```
>robocopy "C:\Users\testuser\Documents\GmatTest\output\R2014a-rc#-<config>"
"\\mesa-file\595\GMAT\TestResults\win7\R2014a-rc#-<config>" /E /MT /LOG:%TEMP%\robocopy.txt
```

Issues for RC2

- [GMT-4573](#)