

Guidelines of the Development and Tag Process of the Geant4 Low-Energy Electromagnetic Physics Working Group

P. Nieminen and M.G. Pia

1 Introduction

This document provides guidelines for a development and tag process. It applies to developments in the lowenergy category, in system tests and examples under the responsibility of the Low-Energy Electromagnetic Physics Working Group.

2 Development and tag process

2.1 Lowenergy category

The following items are under the responsibility of the developer:

1. Perform the development and tag process in agreement with the present document.
2. Code developments should be carried on according to the plans agreed by the WG, respecting the agreed priorities.
3. Responsibilities for code developments are agreed in WG meetings and documented in the minutes of the meetings.
4. Code developments should be carried on according to Geant4 coding guidelines.
5. Check out a copy of the design from the repository.
6. Consult with the WG coordinators if there are any questions about understanding the design documents.
7. Start the development from the most recent monthly global reference tag, with on top the most recent lowenergy tag recommended by the WG coordinators for group's use.
8. Update the work environment regularly to the announced monthly global reference tags, lowenergy tags and tags of system tests relevant to the lowenergy category.
9. Implement the code according to the design.
10. Embed in the development process the testing process described in a separate document.
11. Commit the code under development to the CVS repository frequently, and well in advance with respect to the scheduled release, even if it is not fully functional yet; the minimal requirement is that the code should compile clear of errors and warnings on at least two supported platforms. The code should be committed to the repository in time for monthly reference tags and scheduled Geant4 releases, according to the internal delivery deadlines indicated by the WG coordinators.
12. The procedure to commit code to CVS is:
 - Update the work environment to CVS head: `cvs update -A -d -P`
 - Add new files (if pertinent): `cvs add filename`
 - Commit the new code to the CVS repository: `cvs commit -m "Comment" filename`, where "comment" is a meaningful description of the development being checked in.
 - Inform the WG coordinators about the new code committed to the repository.
13. Adding new directories to the CVS repository may be done only in agreement with WG coordinators.

14. Issue "private" tags as frequently as needed. *Private tags are category tags with a format different from the official Geant4 format. Only WG coordinators should issue official category tags.*
15. Provide adequate documentation for all developments done. Documentation for the Geant4 User manuals should be committed to CVS in the appropriate directories; inform the WG coordinators when contributions to Geant4 User Documentation are checked in.
16. Sources of physics models, data etc. used in the development should be properly documented also in the code itself; reference to publications should be provided, whenever pertinent.
17. Document any numbers, constants etc. used in the code.
18. Use units explicitly throughout the code.
19. Provide information to the WG coordinators concerning new developments, to contribute to the traceability map.
20. A developer may only develop and modify code assigned to his/her own responsibility; modifications to code under somebody else's responsibility (even bug fixes) may be done only if previously agreed with the responsible for that code (if available) and with WG coordinators (always).
21. Modifications of features of the physics models may be done only if previously agreed with the WG coordinators.
22. No modifications to the design are allowed in the development phase. Modifications to the design, if needed, should be done according to a controlled process, in agreement with the design responsible(s) and the WG coordinators.

The process above should be performed as part of **iterative and incremental design-development-test cycles, not as a code-first, test-at-the-end sequence. Intermediate releases of new developments in progress, still with limited functionalities, are strongly encouraged.**

The adoption of the PSP¹ is encouraged, especially in the case of novice developers.

The following items are under the responsibility of WG coordinators:

1. Perform the tag process of the lowenergy category in agreement with the present document and with the Geant4 Tag and Release Policy documented in http://geant4.cern.ch/geant4/working_groups/testing/tag_release_policy.html
2. Provide up-to-date designs in the repository, on which to base developments.
3. Identify the WG members responsible for the development of a WG project and agree with them about undertaking the responsibility of the task.
4. Integrate developments from various sources.
5. Embed in the integration and tagging process the testing process described in a separate document.
6. Issue category tags (distinguishing tags with bug fixes and tags with new features).
7. Notify the STT² when a new tag is proposed for inclusion in system testing, preferably by the Bonsai system.
8. Notify the WG members when a category tag of general interest is issued.
9. Category tags should be issued at least once a month for the Geant4 global reference tag (if any new material in the lowenergy category is available) and in time for Geant4 scheduled releases. Additional tags may be issued at the discretion of the WG coordinators, if needed.
10. Verify that the user requirements are satisfied by the developments.
11. Update the traceability map.
12. Collect quantitative information, to be used for improvements of the development process, and analyse it; propose process improvements, based on such analysis.
13. Perform defect analysis and discuss it within the WG, for the purpose of preventing future occurrences of the same defects.
14. Train WG members to the development process.

¹ SEI's Personal Software Process

² System Testing Team

15. Verify the agreed development process to be correctly applied by all WG members.
16. Maintain the present document.

2.2 System tests associated to the Low Energy Electromagnetic WG

The following items are under the responsibility of System Tests developers:

1. Perform the development process for system tests and examples in agreement with the present document and according to the guidelines documented in the previous section for the lowenergy category code.
2. The code should be committed to the repository in time for monthly reference tags and scheduled Geant4 releases, according to the internal delivery deadlines indicated by the WG coordinators.
3. A system test must include, in addition to the code, the following files in its main directory:
 - the GNUmakefile
 - an input macro, named testID.in, to be used in regular system testing (ID being the test identifier)
 - an input macro, named testN.large_N.in, to be used in high statistics system testing
 - a reference output corresponding to the regular macro, named testID.out
 - a reference output corresponding to the high statistics macro, named testID.large_N.out

The following items are under the responsibility of System Tests responsables:

1. Perform the processes under their responsibility in agreement with the present document
2. Document the requirements for the system test; discuss them with WG coordinators; define an agreed version of the requirements document and commit it to the CVS (documents) repository.
3. Integrate test code developments from various sources.
4. Embed in the integration process the testing process described in a separate document.
4. Issue test tags (distinguishing tags with bug fixes and tags with new features), when delegated by WG coordinators; notify the WG coordinators when a new tag is issued. In exceptional cases WG coordinators may issue test tags too.
5. The code should be tagged in time for monthly reference tags and scheduled Geant4 releases, according to the internal delivery deadlines indicated by the WG coordinators and before the start of the system testing phase of scheduled releases.
5. Convey information regards the update of the traceability map to WG coordinators.

The following items are under the responsibility of WG coordinators:

1. Perform the processes under their responsibility in agreement with the present document and with the Geant4 Tag and Release Policy document.
2. Review the requirements for system tests with the responsible collaborators and approve an agreed version of the requirements document; issue an appropriate tag in the documents repository. Review and agree with them a test design; approve it and issue an appropriate tag in the documents repository.
3. Review the final version of the system tests prior to proposing it to the STT.
6. Notify the System Testing Team when a new system test tag is proposed for inclusion in a Geant4 monthly reference tag or a scheduled release, or delegate a System test responsible to do so.
4. Notify the WG members when a tag of general interest is issued, or delegate a System test responsible to do so.
5. Update the traceability map, for what concerns system testing.

2.3 Advanced examples associated to the Low Energy Electromagnetic WG

The following items are under the responsibility of Examples developers:

6. Perform the development process for advanced examples in agreement with the present document and according to the guidelines documented in the previous section for the lowenergy category code.

The following items are under the responsibility of Examples responsables:

7. Perform the processes under their responsibility in agreement with the present document.
8. Document the requirements for the example; discuss them with WG coordinators; define an agreed version of the requirements document and commit it to the CVS (documents) repository.
9. Integrate example code developments from various sources.
10. Embed in the integration process the testing process described in a separate document.
7. Issue example tags (distinguishing tags with bug fixes and tags with new features), when delegated by WG coordinators; notify the WG coordinators when a new tag is issued. In exceptional cases WG coordinators may issue example tags too.
11. Convey information regards the update of the traceability map to WG coordinators.

The following items are under the responsibility of WG coordinators:

6. Perform the processes under their responsibility in agreement with the present document and with the Geant4 Tag and Release Policy document.
7. Review the requirements for examples with the responsible collaborators and approve an agreed version of the requirements document; issue an appropriate tag in the documents repository. Review and agree with them an example design; approve it and issue an appropriate tag in the documents repository.
8. Review the final version of the example prior to proposing it to the Release Manager.
12. Notify the Release Manager when a new advanced example tag is proposed for inclusion in a Geant4 monthly reference tag or a scheduled release, or delegate responsible to do so.
9. Notify the WG members when a tag of general interest is issued, or delegate an Example responsible to do so.
10. Convey information about the availability of documentation updates to the Documentation coordinator, or delegate an Example responsible to do so.
11. Update the traceability map, for what concerns examples.

3 Notes

It is stressed that **code should be committed to the CVS repository frequently, and well in advance with respect to scheduled releases, to allow for adequate testing.** Large amounts of code committed to the repository only at the last minute before a release deadline will not be considered for inclusion in the release and will be delayed until the following scheduled release.

Documentation about CVS and its usage in Geant4 is available at <http://wwwinfo.cern.ch/asd/geant4/collaboration/cvs.html>

Any modification of the development and tag process should be discussed and agreed within the WG. Once agreed, it should be properly documented by updating the present document.

4 Acknowledgements

Many thanks to G. Cosmo for reviewing this document.