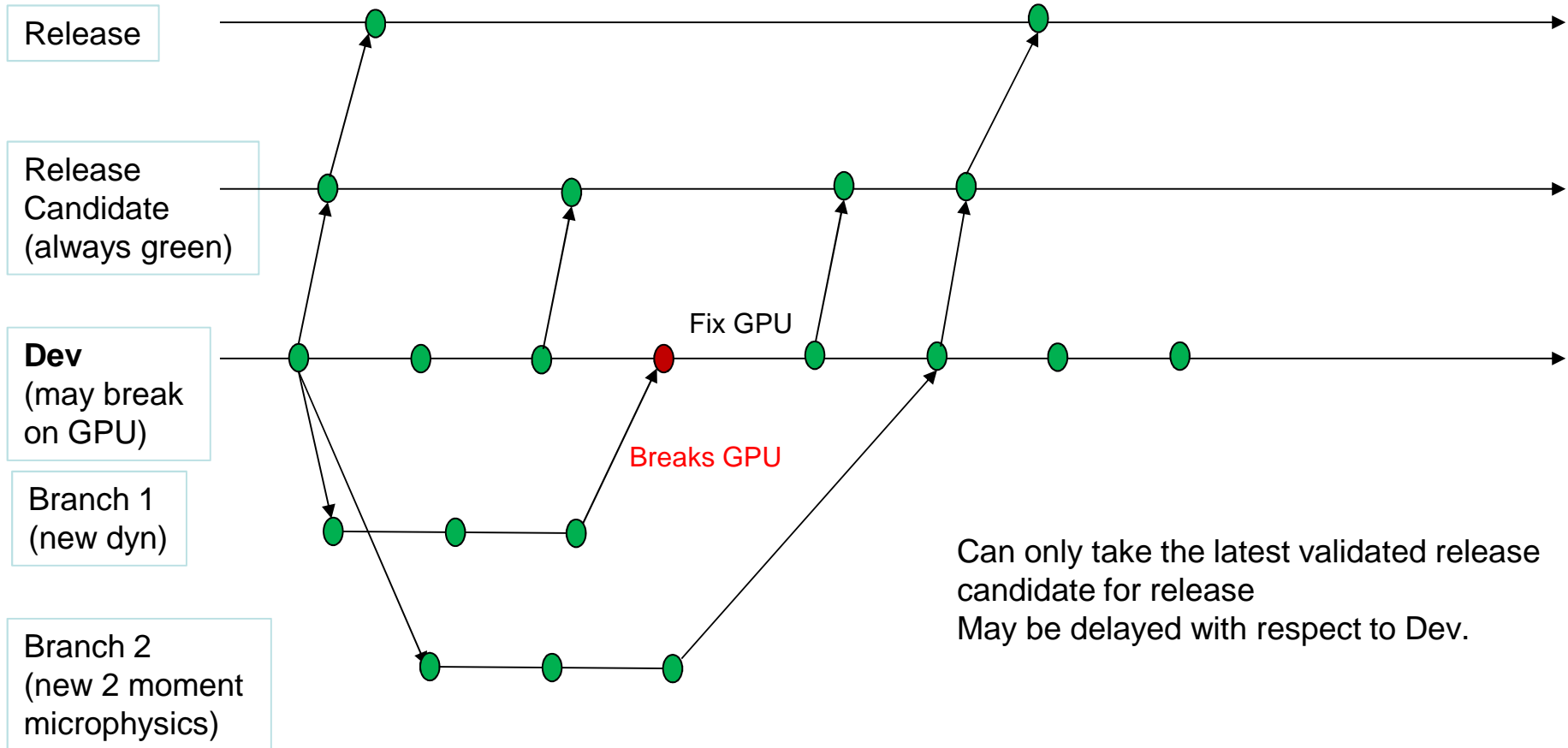# Workflow discussion

- The integration of POMPA work introduces some complexity in the COSMO model development, maintenance : CPU, GPU, single precision, double precision

- In order to keep high productivity and software quality the workflow and communication between COSMO developers need to be adapted.

- The use of a common repository is required to favor collaboration

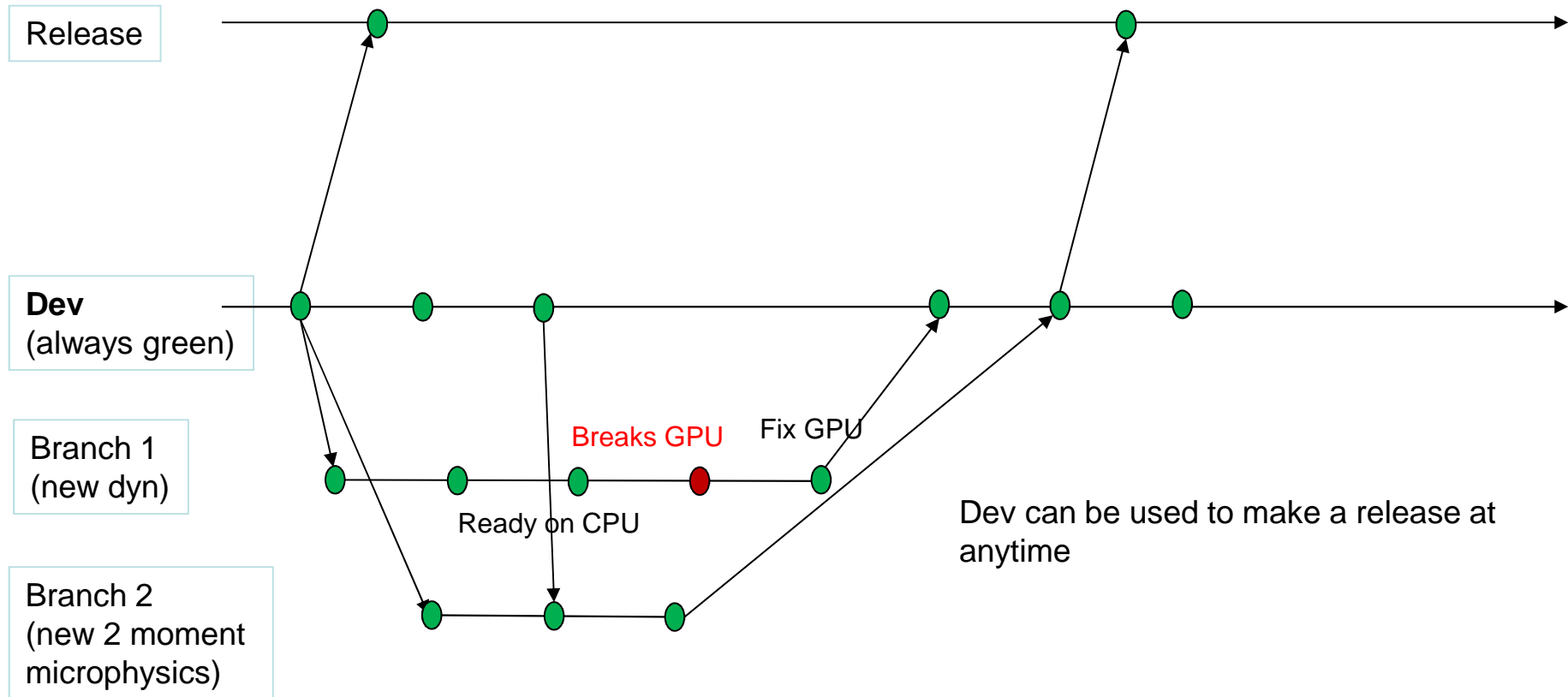- Use of automatic testing infrastructure (Jenkins – see Pascal talk)

Workflow 1 : all development via SCA Adaptation to GPU in second step

Release

Release Candidate (always green)

Dev (may break on GPU)

Branch 1 (new dyn)

Branch 2 (new 2 moment microphysics)

Fix GPU

Breaks GPU

Can only take the latest validated release candidate for release
May be delayed with respect to Dev.

Workflow 2 : Adaptation to GPU before sending code to SCA

Release

Dev
(always green)

Branch 1
(new dyn)

Branch 2
(new 2 moment
microphysics)

Breaks GPU

Fix GPU

Ready on CPU

Dev can be used to make a release at anytime

Discussion

- **Which workflow:**
  1: Code is fix/checked for GPU after it was sent to SCA. SCA and C++/GPU developer need use common repository
  2: Code is ready on both CPU and GPU before beeing sent to SCA: all invloved developer should use common repository

- **Use of revision control system (git or svn ? test users for an evaluation period ?)**

- **Use Jenkins for testing**

- **How to keep GPU/CPU single/double in sync ?**

  - Change in existing code should always be synchronized in C++/GPU
  code before any official release
  ex: new coefficient in the dynamic

  - New features, not impacting currently GPU namelists, may not be ported
  to GPU, ex: new 2 moments microphysics
  SMC should decide which new features are ported to GPU