



arm

TF-M dependency on tf-m-tests

Code repo management

Kevin Peng
2021-01-07

Dependency auto downloading in TF-M Build System

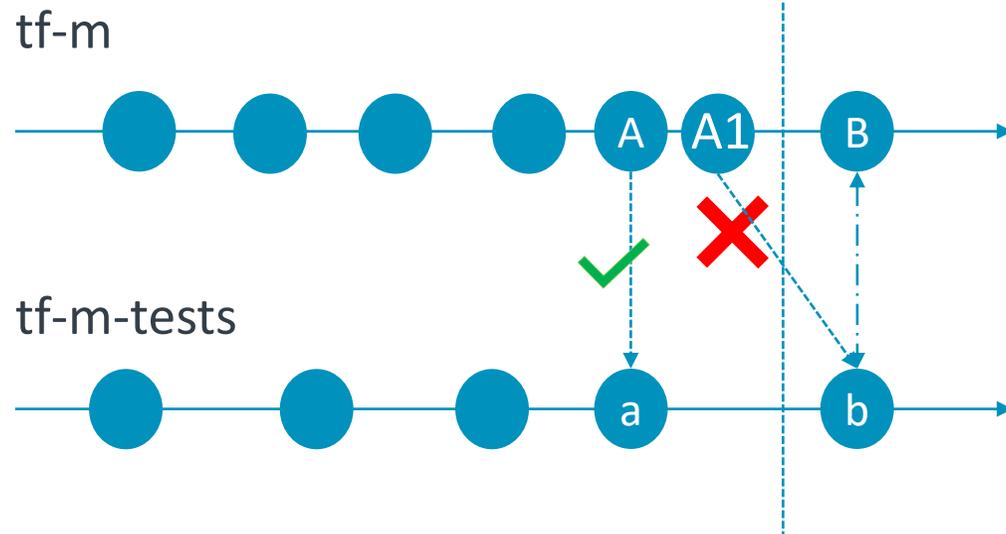
- The TF-M build system downloads dependency repos automatically on a clean build
- Great feature but has some issues with the tf-m-tests repo

```
set(TFM_TEST_REPO_PATH ..... "DOWNLOAD" .. CACHE .. PATH ..  
set(TFM_TEST_REPO_VERSION ..... "master" .. CACHE .. STRING ..  
set(CMSIS_5_PATH ..... "DOWNLOAD" .. CACHE .. PATH ..  
  
set(MCUBOOT_PATH ..... "DOWNLOAD" .. CACHE .. PATH ..  
set(MCUBOOT_VERSION ..... "81d19f0" .. CACHE .. STRING ..  
  
set(PSA_ARCH_TESTS_PATH ..... "DOWNLOAD" .. CACHE .. PATH ..  
set(PSA_ARCH_TESTS_VERSION ..... "90c8e680" .. CACHE .. STRING ..
```

```
if ("${TFM_TEST_REPO_PATH}" STREQUAL "DOWNLOAD")  
  FetchContent_Declare(tfm_test_repo  
    GIT_REPOSITORY https://git.trustedfirmware.org/TF-M/tf-m-tests.git  
    GIT_TAG ${TFM_TEST_REPO_VERSION}  
    GIT_PROGRESS TRUE  
  )
```

The issue with the tf-m-tests repo

The latest test repo is downloaded by default



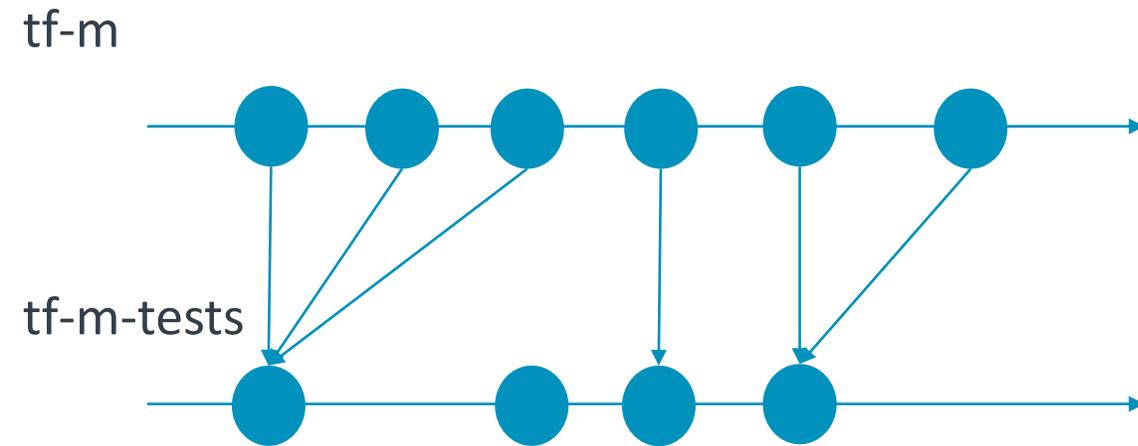
- Working on the latest code base has no issues
- When there are dependency changes merged (B and b), developers working on old versions (A) will get build broken.
- The developers must:
 - rebase their works to the latest master – which might be a huge effort
 - use a local copy of the test repo with an old compatible version – there is no explicit information of the mapping between the repos.

Why is it “master”

- The tf-m-tests repo is part of the TF-M project, not an independent project
- It's currently coupled with the main repo a lot
- The two repos should keep aligned
- Hard to manage it like an “external” project

The possible solution

Having 1-1 map between the repos



```
set(TFM_TEST_REPO_PATH ..... "DOWNLOAD"  
set(TFM_TEST_REPO_VERSION ..... "master"
```



```
set(TFM_TEST_REPO_PATH ..... "DOWNLOAD"  
set(TFM_TEST_REPO_VERSION ..... "0bdbdef0"
```

- Every commit of TF-M has a fixed workable version of tf-m-tests
- Build will not be broken even breaking patches merged
- Version information for every commit

When to update the TFM_TEST_REPO_VERSION

The version should be updated when there is a complete change in test repo

None-breaking changes

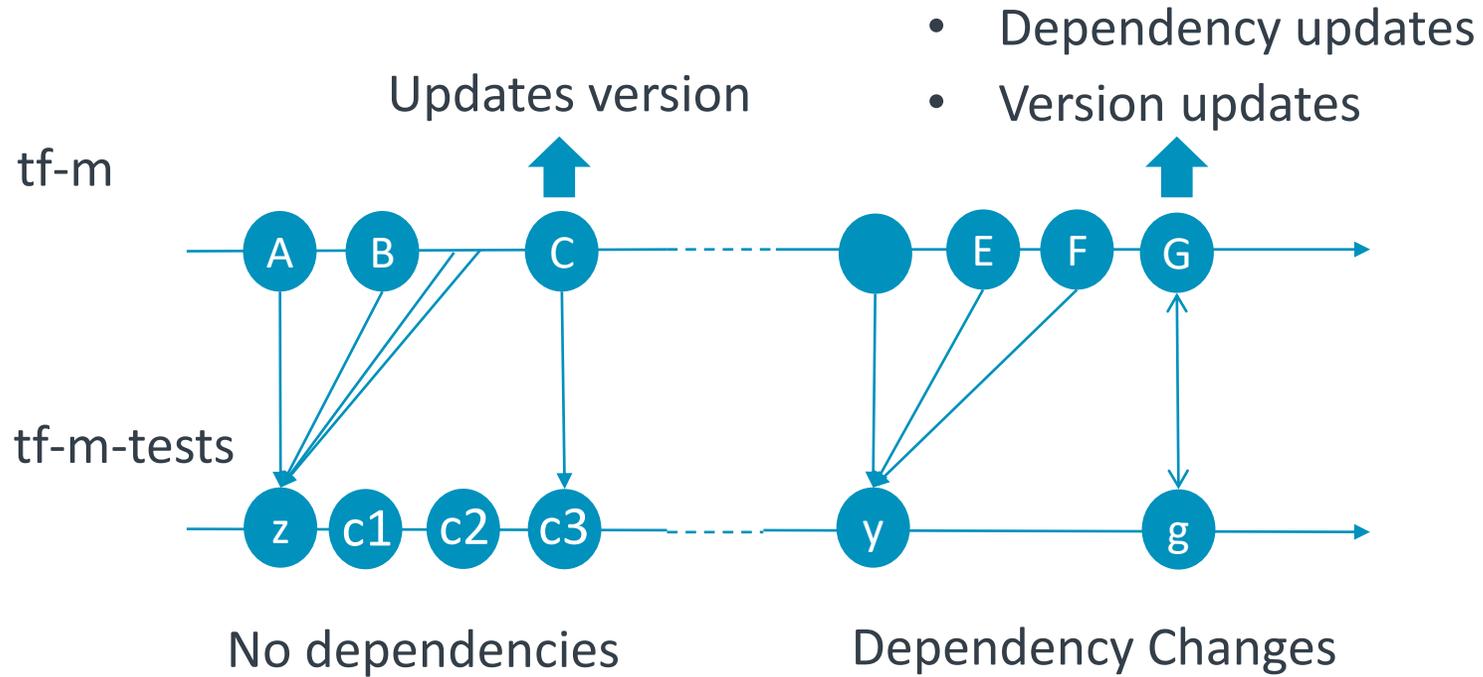
- A complete change is one or more patches which are for the same purpose, such as a new test case and an enhancement in the framework
- Patches can be merged individually provided they don't break anything
- The TFM_TEST_REPO_VERSION updates to last commit when all the patches in one change are merged
- The version shouldn't be updated between patches of one change
 - The change has not been completed, updating the version wouldn't bring benefits but overhead

Dependency patches

- The version is updated along with the dependency patch in TF-M

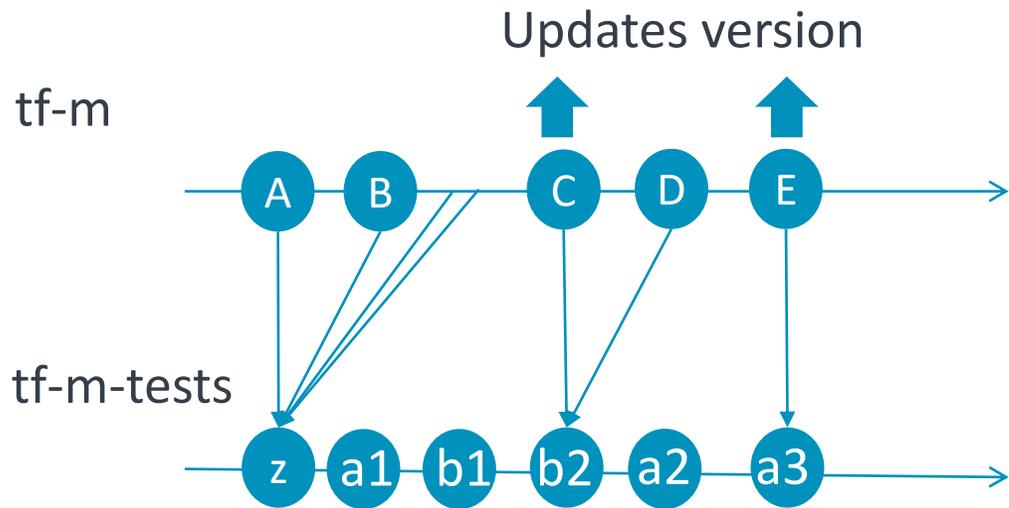
Working Model

Simplest cases



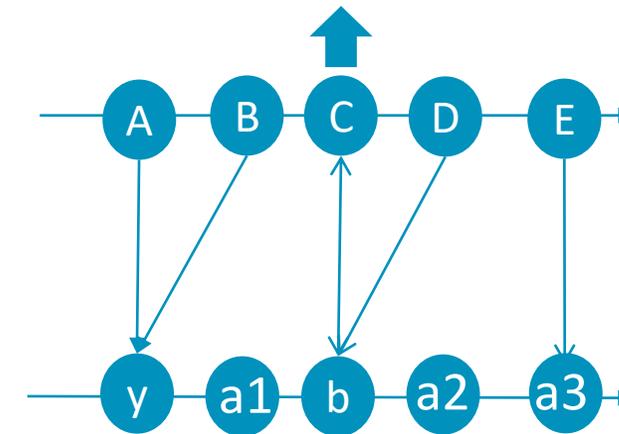
Working Model

Complex cases



Both a & b are no dependency changes

- Dependency updates
- Version updates



a has no dependency
b is dependency change

arm

Thank You

Danke

Merci

谢谢

ありがとう

Gracias

Kiitos

감사합니다

धन्यवाद

شكرًا

ধন্যবাদ

תודה

More feedbacks are welcome

- mailing list: tf-m@lists.trustedfirmware.org
- Patch: <https://review.trustedfirmware.org/c/TF-M/trusted-firmware-m/+7550>



The Arm trademarks featured in this presentation are registered trademarks or trademarks of Arm Limited (or its subsidiaries) in the US and/or elsewhere. All rights reserved. All other marks featured may be trademarks of their respective owners.

www.arm.com/company/policies/trademarks