

git_comments:

1. Script to build ccache for debian and ubuntu based images. Manually specify x86 gcc versions so that this script remains compatible with dockcross (which uses an ARM based gcc by default).
2. Script to build ccache for debian and ubuntu based images. Manually specify x86 gcc versions so that this script remains compatible with dockcross (which uses an ARM based gcc by default).

git_commits:

1. **summary:** [MXNET-553] Restructure dockcross dockerfiles to fix caching (#11302)
message: [MXNET-553] Restructure dockcross dockerfiles to fix caching (#11302) * Add ccache reporting to CI * Restructure dockcross dockerfiles to fix caching

github_issues:

1. **title:** Docker cache is not being used
body: As can be seen for all [latest builds](http://jenkins.mxnet-ci.amazon-ml.com/blue/organizations/jenkins/incubator-mxnet/detail/PR-11252/1/pipeline/70): docker cache is not being reused and images are always getting rebuild. This is due to the ccache change <https://github.com/apache/incubator-mxnet/pull/11151> The problem is that due to [multiheaded docker build](https://github.com/apache/incubator-mxnet/blob/master/ci/docker/Dockerfile.build.armv7#L21-L31) the first build does not get a tag and is considered an intermediate layer. The `--cache-from` parameter in [build.py](https://github.com/apache/incubator-mxnet/blob/master/ci/build.py#L79) restricts the cache only to tagged images and that leads to rebuilds.
2. **title:** Docker cache is not being used
body: As can be seen for all [latest builds](http://jenkins.mxnet-ci.amazon-ml.com/blue/organizations/jenkins/incubator-mxnet/detail/PR-11252/1/pipeline/70): docker cache is not being reused and images are always getting rebuild. This is due to the ccache change <https://github.com/apache/incubator-mxnet/pull/11151> The problem is that due to [multiheaded docker build](https://github.com/apache/incubator-mxnet/blob/master/ci/docker/Dockerfile.build.armv7#L21-L31) the first build does not get a tag and is considered an intermediate layer. The `--cache-from` parameter in [build.py](https://github.com/apache/incubator-mxnet/blob/master/ci/build.py#L79) restricts the cache only to tagged images and that leads to rebuilds.
3. **title:** Docker cache is not being used
body: As can be seen for all [latest builds](http://jenkins.mxnet-ci.amazon-ml.com/blue/organizations/jenkins/incubator-mxnet/detail/PR-11252/1/pipeline/70): docker cache is not being reused and images are always getting rebuild. This is due to the ccache change <https://github.com/apache/incubator-mxnet/pull/11151> The problem is that due to [multiheaded docker build](https://github.com/apache/incubator-mxnet/blob/master/ci/docker/Dockerfile.build.armv7#L21-L31) the first build does not get a tag and is considered an intermediate layer. The `--cache-from` parameter in [build.py](https://github.com/apache/incubator-mxnet/blob/master/ci/build.py#L79) restricts the cache only to tagged images and that leads to rebuilds.
4. **title:** Docker cache is not being used
body: As can be seen for all [latest builds](http://jenkins.mxnet-ci.amazon-ml.com/blue/organizations/jenkins/incubator-mxnet/detail/PR-11252/1/pipeline/70): docker cache is not being reused and images are always getting rebuild. This is due to the ccache change <https://github.com/apache/incubator-mxnet/pull/11151> The problem is that due to [multiheaded docker build](https://github.com/apache/incubator-mxnet/blob/master/ci/docker/Dockerfile.build.armv7#L21-L31) the first build does not get a tag and is considered an intermediate layer. The `--cache-from` parameter in [build.py](https://github.com/apache/incubator-mxnet/blob/master/ci/build.py#L79) restricts the cache only to tagged images and that leads to rebuilds.
5. **title:** Docker cache is not being used
body: As can be seen for all [latest builds](http://jenkins.mxnet-ci.amazon-ml.com/blue/organizations/jenkins/incubator-mxnet/detail/PR-11252/1/pipeline/70): docker cache is not being reused and images are always getting rebuild. This is due to the ccache change <https://github.com/apache/incubator-mxnet/pull/11151> The problem is that due to [multiheaded docker build](https://github.com/apache/incubator-mxnet/blob/master/ci/docker/Dockerfile.build.armv7#L21-L31) the first build does not get a tag and is considered an intermediate layer. The `--cache-from`

parameter in [build.py](https://github.com/apache/incubator-mxnet/blob/master/ci/build.py#L79) restricts the cache only to tagged images and that leads to rebuilds.

6. **title:** Docker cache is not being used

body: As can be seen for all [latest builds](http://jenkins.mxnet-ci.amazon-ml.com/blue/organizations/jenkins/incubator-mxnet/detail/PR-11252/1/pipeline/70): docker cache is not being reused and images are always getting rebuild. This is due to the ccache change https://github.com/apache/incubator-mxnet/pull/11151 The problem is that due to [multiheaded docker build](https://github.com/apache/incubator-mxnet/blob/master/ci/docker/Dockerfile.build.armv7#L21-L31) the first build does not get a tag and is considered an intermediate layer. The `--cache-from` parameter in [build.py](https://github.com/apache/incubator-mxnet/blob/master/ci/build.py#L79) restricts the cache only to tagged images and that leads to rebuilds.

7. **title:** Docker cache is not being used

body: As can be seen for all [latest builds](http://jenkins.mxnet-ci.amazon-ml.com/blue/organizations/jenkins/incubator-mxnet/detail/PR-11252/1/pipeline/70): docker cache is not being reused and images are always getting rebuild. This is due to the ccache change https://github.com/apache/incubator-mxnet/pull/11151 The problem is that due to [multiheaded docker build](https://github.com/apache/incubator-mxnet/blob/master/ci/docker/Dockerfile.build.armv7#L21-L31) the first build does not get a tag and is considered an intermediate layer. The `--cache-from` parameter in [build.py](https://github.com/apache/incubator-mxnet/blob/master/ci/build.py#L79) restricts the cache only to tagged images and that leads to rebuilds.

8. **title:** Docker cache is not being used

body: As can be seen for all [latest builds](http://jenkins.mxnet-ci.amazon-ml.com/blue/organizations/jenkins/incubator-mxnet/detail/PR-11252/1/pipeline/70): docker cache is not being reused and images are always getting rebuild. This is due to the ccache change https://github.com/apache/incubator-mxnet/pull/11151 The problem is that due to [multiheaded docker build](https://github.com/apache/incubator-mxnet/blob/master/ci/docker/Dockerfile.build.armv7#L21-L31) the first build does not get a tag and is considered an intermediate layer. The `--cache-from` parameter in [build.py](https://github.com/apache/incubator-mxnet/blob/master/ci/build.py#L79) restricts the cache only to tagged images and that leads to rebuilds.

github_issues_comments:

1. I don't have permissions to add labels, but I believe that this should be tagged with the `CI` label.
2. http://jenkins.mxnet-ci.amazon-ml.com/blue/organizations/jenkins/incubator-mxnet/detail/PR-11210/8/pipeline something is happening with ccache is it related or a different issue?
3. @ThomasDelteil it is indeed related to ccache, but is a different error
4. thanks @ThomasDelteil for the link!
5. @sandeep-krishnamurthy could you add label "CI", "Docker" to this?
6. The issue should have been fixed by https://github.com/apache/incubator-mxnet/pull/11302

github_pulls:

1. **title:** [MXNET-553] Restructure dockcross dockerfiles to fix caching

body: ## Description ## This PR restructures the dockcross Dockerfiles to remove their multi-head components. These multi-head components made it difficult to properly provide a remote cache for the Dockerfiles, and was slowing down builds. A simple work-around for this was to generalize the ubuntu_ccache.sh file such that it works with ubuntu, debian, and in cross-compilation containers. We can now use this script the same way in a variety of containers, which eliminates the need for a special multi-head docker stage. Should address #11257. ## Checklist ## #### Essentials #### Please feel free to remove inapplicable items for your PR. - [x] The PR title starts with [MXNET-\$JIRA_ID], where \$JIRA_ID refers to the relevant [JIRA issue](https://issues.apache.org/jira/projects/MXNET/issues) created (except PRs with tiny changes) - [x] Changes are complete (i.e. I finished coding on this PR) - [x] All changes have test coverage: - Ran test_docker_cache.py, ensured both existing tests pass. - [x] Code is well-documented: - [x] To the my best knowledge, examples are either not affected by this change, or have been fixed to be compatible with this change

2. **title:** [MXNET-553] Restructure dockcross dockerfiles to fix caching

body: ## Description ## This PR restructures the dockcross Dockerfiles to remove their multi-head components. These multi-head components made it difficult to properly provide a remote cache for the Dockerfiles, and was slowing down builds. A simple work-around for this was to generalize the

ubuntu_ccache.sh file such that it works with ubuntu, debian, and in cross-compilation containers. We can now use this script the same way in a variety of containers, which eliminates the need for a special multi-head docker stage. Should address #11257. ## Checklist ## #### Essentials #### Please feel free to remove inapplicable items for your PR. - [x] The PR title starts with [MXNET-\$JIRA_ID], where \$JIRA_ID refers to the relevant [JIRA issue](https://issues.apache.org/jira/projects/MXNET/issues) created (except PRs with tiny changes) - [x] Changes are complete (i.e. I finished coding on this PR) - [x] All changes have test coverage: - Ran test_docker_cache.py, ensured both existing tests pass. - [x] Code is well-documented: - [x] To the my best knowledge, examples are either not affected by this change, or have been fixed to be compatible with this change

3. **title:** [MXNET-553] Restructure dockcross dockerfiles to fix caching

body: ## Description ## This PR restructures the dockcross Dockerfiles to remove their multi-head components. These multi-head components made it difficult to properly provide a remote cache for the Dockerfiles, and was slowing down builds. A simple work-around for this was to generalize the ubuntu_ccache.sh file such that it works with ubuntu, debian, and in cross-compilation containers. We can now use this script the same way in a variety of containers, which eliminates the need for a special multi-head docker stage. Should address #11257. ## Checklist ## #### Essentials #### Please feel free to remove inapplicable items for your PR. - [x] The PR title starts with [MXNET-\$JIRA_ID], where \$JIRA_ID refers to the relevant [JIRA issue](https://issues.apache.org/jira/projects/MXNET/issues) created (except PRs with tiny changes) - [x] Changes are complete (i.e. I finished coding on this PR) - [x] All changes have test coverage: - Ran test_docker_cache.py, ensured both existing tests pass. - [x] Code is well-documented: - [x] To the my best knowledge, examples are either not affected by this change, or have been fixed to be compatible with this change

label: code-design

4. **title:** [MXNET-553] Restructure dockcross dockerfiles to fix caching

body: ## Description ## This PR restructures the dockcross Dockerfiles to remove their multi-head components. These multi-head components made it difficult to properly provide a remote cache for the Dockerfiles, and was slowing down builds. A simple work-around for this was to generalize the ubuntu_ccache.sh file such that it works with ubuntu, debian, and in cross-compilation containers. We can now use this script the same way in a variety of containers, which eliminates the need for a special multi-head docker stage. Should address #11257. ## Checklist ## #### Essentials #### Please feel free to remove inapplicable items for your PR. - [x] The PR title starts with [MXNET-\$JIRA_ID], where \$JIRA_ID refers to the relevant [JIRA issue](https://issues.apache.org/jira/projects/MXNET/issues) created (except PRs with tiny changes) - [x] Changes are complete (i.e. I finished coding on this PR) - [x] All changes have test coverage: - Ran test_docker_cache.py, ensured both existing tests pass. - [x] Code is well-documented: - [x] To the my best knowledge, examples are either not affected by this change, or have been fixed to be compatible with this change

5. **title:** [MXNET-553] Restructure dockcross dockerfiles to fix caching

body: ## Description ## This PR restructures the dockcross Dockerfiles to remove their multi-head components. These multi-head components made it difficult to properly provide a remote cache for the Dockerfiles, and was slowing down builds. A simple work-around for this was to generalize the ubuntu_ccache.sh file such that it works with ubuntu, debian, and in cross-compilation containers. We can now use this script the same way in a variety of containers, which eliminates the need for a special multi-head docker stage. Should address #11257. ## Checklist ## #### Essentials #### Please feel free to remove inapplicable items for your PR. - [x] The PR title starts with [MXNET-\$JIRA_ID], where \$JIRA_ID refers to the relevant [JIRA issue](https://issues.apache.org/jira/projects/MXNET/issues) created (except PRs with tiny changes) - [x] Changes are complete (i.e. I finished coding on this PR) - [x] All changes have test coverage: - Ran test_docker_cache.py, ensured both existing tests pass. - [x] Code is well-documented: - [x] To the my best knowledge, examples are either not affected by this change, or have been fixed to be compatible with this change

6. **title:** [MXNET-553] Restructure dockcross dockerfiles to fix caching

body: ## Description ## This PR restructures the dockcross Dockerfiles to remove their multi-head components. These multi-head components made it difficult to properly provide a remote cache for the Dockerfiles, and was slowing down builds. A simple work-around for this was to generalize the ubuntu_ccache.sh file such that it works with ubuntu, debian, and in cross-compilation containers. We can now use this script the same way in a variety of containers, which eliminates the need for a special multi-head docker stage. Should address #11257. ## Checklist ## #### Essentials #### Please feel free to remove inapplicable items for your PR. - [x] The PR title starts with [MXNET-\$JIRA_ID], where \$JIRA_ID refers to the relevant [JIRA issue](https://issues.apache.org/jira/projects/MXNET/issues) created (except PRs with tiny changes) - [x] Changes are complete (i.e. I finished coding on this PR) - [x] All changes

have test coverage: - Ran test_docker_cache.py, ensured both existing tests pass. - [x] Code is well-documented: - [x] To the my best knowledge, examples are either not affected by this change, or have been fixed to be compatible with this change

7. **title:** [MXNET-553] Restructure dockcross dockerfiles to fix caching

body: ## Description ## This PR restructures the dockcross Dockerfiles to remove their multi-head components. These multi-head components made it difficult to properly provide a remote cache for the Dockerfiles, and was slowing down builds. A simple work-around for this was to generalize the ubuntu_ccache.sh file such that it works with ubuntu, debian, and in cross-compilation containers. We can now use this script the same way in a variety of containers, which eliminates the need for a special multi-head docker stage. Should address #11257. ## Checklist ## #### Essentials #### Please feel free to remove inapplicable items for your PR. - [x] The PR title starts with [MXNET-\$JIRA_ID], where \$JIRA_ID refers to the relevant [JIRA issue](https://issues.apache.org/jira/projects/MXNET/issues) created (except PRs with tiny changes) - [x] Changes are complete (i.e. I finished coding on this PR) - [x] All changes have test coverage: - Ran test_docker_cache.py, ensured both existing tests pass. - [x] Code is well-documented: - [x] To the my best knowledge, examples are either not affected by this change, or have been fixed to be compatible with this change

8. **title:** [MXNET-553] Restructure dockcross dockerfiles to fix caching

body: ## Description ## This PR restructures the dockcross Dockerfiles to remove their multi-head components. These multi-head components made it difficult to properly provide a remote cache for the Dockerfiles, and was slowing down builds. A simple work-around for this was to generalize the ubuntu_ccache.sh file such that it works with ubuntu, debian, and in cross-compilation containers. We can now use this script the same way in a variety of containers, which eliminates the need for a special multi-head docker stage. Should address #11257. ## Checklist ## #### Essentials #### Please feel free to remove inapplicable items for your PR. - [x] The PR title starts with [MXNET-\$JIRA_ID], where \$JIRA_ID refers to the relevant [JIRA issue](https://issues.apache.org/jira/projects/MXNET/issues) created (except PRs with tiny changes) - [x] Changes are complete (i.e. I finished coding on this PR) - [x] All changes have test coverage: - Ran test_docker_cache.py, ensured both existing tests pass. - [x] Code is well-documented: - [x] To the my best knowledge, examples are either not affected by this change, or have been fixed to be compatible with this change

9. **title:** [MXNET-553] Restructure dockcross dockerfiles to fix caching

body: ## Description ## This PR restructures the dockcross Dockerfiles to remove their multi-head components. These multi-head components made it difficult to properly provide a remote cache for the Dockerfiles, and was slowing down builds. A simple work-around for this was to generalize the ubuntu_ccache.sh file such that it works with ubuntu, debian, and in cross-compilation containers. We can now use this script the same way in a variety of containers, which eliminates the need for a special multi-head docker stage. Should address #11257. ## Checklist ## #### Essentials #### Please feel free to remove inapplicable items for your PR. - [x] The PR title starts with [MXNET-\$JIRA_ID], where \$JIRA_ID refers to the relevant [JIRA issue](https://issues.apache.org/jira/projects/MXNET/issues) created (except PRs with tiny changes) - [x] Changes are complete (i.e. I finished coding on this PR) - [x] All changes have test coverage: - Ran test_docker_cache.py, ensured both existing tests pass. - [x] Code is well-documented: - [x] To the my best knowledge, examples are either not affected by this change, or have been fixed to be compatible with this change

10. **title:** [MXNET-553] Restructure dockcross dockerfiles to fix caching

body: ## Description ## This PR restructures the dockcross Dockerfiles to remove their multi-head components. These multi-head components made it difficult to properly provide a remote cache for the Dockerfiles, and was slowing down builds. A simple work-around for this was to generalize the ubuntu_ccache.sh file such that it works with ubuntu, debian, and in cross-compilation containers. We can now use this script the same way in a variety of containers, which eliminates the need for a special multi-head docker stage. Should address #11257. ## Checklist ## #### Essentials #### Please feel free to remove inapplicable items for your PR. - [x] The PR title starts with [MXNET-\$JIRA_ID], where \$JIRA_ID refers to the relevant [JIRA issue](https://issues.apache.org/jira/projects/MXNET/issues) created (except PRs with tiny changes) - [x] Changes are complete (i.e. I finished coding on this PR) - [x] All changes have test coverage: - Ran test_docker_cache.py, ensured both existing tests pass. - [x] Code is well-documented: - [x] To the my best knowledge, examples are either not affected by this change, or have been fixed to be compatible with this change

11. **title:** [MXNET-553] Restructure dockcross dockerfiles to fix caching

body: ## Description ## This PR restructures the dockcross Dockerfiles to remove their multi-head components. These multi-head components made it difficult to properly provide a remote cache for the Dockerfiles, and was slowing down builds. A simple work-around for this was to generalize the

ubuntu_ccache.sh file such that it works with ubuntu, debian, and in cross-compilation containers. We can now use this script the same way in a variety of containers, which eliminates the need for a special multi-head docker stage. Should address #11257. ## Checklist ## #### Essentials #### Please feel free to remove inapplicable items for your PR. - [x] The PR title starts with [MXNET-\$JIRA_ID], where \$JIRA_ID refers to the relevant [JIRA issue](https://issues.apache.org/jira/projects/MXNET/issues) created (except PRs with tiny changes) - [x] Changes are complete (i.e. I finished coding on this PR) - [x] All changes have test coverage: - Ran test_docker_cache.py, ensured both existing tests pass. - [x] Code is well-documented: - [x] To the my best knowledge, examples are either not affected by this change, or have been fixed to be compatible with this change

12. **title:** [MXNET-553] Restructure dockcross dockerfiles to fix caching

body: ## Description ## This PR restructures the dockcross Dockerfiles to remove their multi-head components. These multi-head components made it difficult to properly provide a remote cache for the Dockerfiles, and was slowing down builds. A simple work-around for this was to generalize the ubuntu_ccache.sh file such that it works with ubuntu, debian, and in cross-compilation containers. We can now use this script the same way in a variety of containers, which eliminates the need for a special multi-head docker stage. Should address #11257. ## Checklist ## #### Essentials #### Please feel free to remove inapplicable items for your PR. - [x] The PR title starts with [MXNET-\$JIRA_ID], where \$JIRA_ID refers to the relevant [JIRA issue](https://issues.apache.org/jira/projects/MXNET/issues) created (except PRs with tiny changes) - [x] Changes are complete (i.e. I finished coding on this PR) - [x] All changes have test coverage: - Ran test_docker_cache.py, ensured both existing tests pass. - [x] Code is well-documented: - [x] To the my best knowledge, examples are either not affected by this change, or have been fixed to be compatible with this change

13. **title:** [MXNET-553] Restructure dockcross dockerfiles to fix caching

body: ## Description ## This PR restructures the dockcross Dockerfiles to remove their multi-head components. These multi-head components made it difficult to properly provide a remote cache for the Dockerfiles, and was slowing down builds. A simple work-around for this was to generalize the ubuntu_ccache.sh file such that it works with ubuntu, debian, and in cross-compilation containers. We can now use this script the same way in a variety of containers, which eliminates the need for a special multi-head docker stage. Should address #11257. ## Checklist ## #### Essentials #### Please feel free to remove inapplicable items for your PR. - [x] The PR title starts with [MXNET-\$JIRA_ID], where \$JIRA_ID refers to the relevant [JIRA issue](https://issues.apache.org/jira/projects/MXNET/issues) created (except PRs with tiny changes) - [x] Changes are complete (i.e. I finished coding on this PR) - [x] All changes have test coverage: - Ran test_docker_cache.py, ensured both existing tests pass. - [x] Code is well-documented: - [x] To the my best knowledge, examples are either not affected by this change, or have been fixed to be compatible with this change

14. **title:** [MXNET-553] Restructure dockcross dockerfiles to fix caching

body: ## Description ## This PR restructures the dockcross Dockerfiles to remove their multi-head components. These multi-head components made it difficult to properly provide a remote cache for the Dockerfiles, and was slowing down builds. A simple work-around for this was to generalize the ubuntu_ccache.sh file such that it works with ubuntu, debian, and in cross-compilation containers. We can now use this script the same way in a variety of containers, which eliminates the need for a special multi-head docker stage. Should address #11257. ## Checklist ## #### Essentials #### Please feel free to remove inapplicable items for your PR. - [x] The PR title starts with [MXNET-\$JIRA_ID], where \$JIRA_ID refers to the relevant [JIRA issue](https://issues.apache.org/jira/projects/MXNET/issues) created (except PRs with tiny changes) - [x] Changes are complete (i.e. I finished coding on this PR) - [x] All changes have test coverage: - Ran test_docker_cache.py, ensured both existing tests pass. - [x] Code is well-documented: - [x] To the my best knowledge, examples are either not affected by this change, or have been fixed to be compatible with this change

15. **title:** [MXNET-553] Restructure dockcross dockerfiles to fix caching

body: ## Description ## This PR restructures the dockcross Dockerfiles to remove their multi-head components. These multi-head components made it difficult to properly provide a remote cache for the Dockerfiles, and was slowing down builds. A simple work-around for this was to generalize the ubuntu_ccache.sh file such that it works with ubuntu, debian, and in cross-compilation containers. We can now use this script the same way in a variety of containers, which eliminates the need for a special multi-head docker stage. Should address #11257. ## Checklist ## #### Essentials #### Please feel free to remove inapplicable items for your PR. - [x] The PR title starts with [MXNET-\$JIRA_ID], where \$JIRA_ID refers to the relevant [JIRA issue](https://issues.apache.org/jira/projects/MXNET/issues) created (except PRs with tiny changes) - [x] Changes are complete (i.e. I finished coding on this PR) - [x] All changes have test coverage: - Ran test_docker_cache.py, ensured both existing tests pass. - [x] Code is well-

documented: - [x] To the my best knowledge, examples are either not affected by this change, or have been fixed to be compatible with this change

16. **title:** [MXNET-553] Restructure dockcross dockerfiles to fix caching

body: ## Description ## This PR restructures the dockcross Dockerfiles to remove their multi-head components. These multi-head components made it difficult to properly provide a remote cache for the Dockerfiles, and was slowing down builds. A simple work-around for this was to generalize the ubuntu_ccache.sh file such that it works with ubuntu, debian, and in cross-compilation containers. We can now use this script the same way in a variety of containers, which eliminates the need for a special multi-head docker stage. Should address #11257. ## Checklist ## #### Essentials #### Please feel free to remove inapplicable items for your PR. - [x] The PR title starts with [MXNET-\$JIRA_ID], where \$JIRA_ID refers to the relevant [JIRA issue](https://issues.apache.org/jira/projects/MXNET/issues) created (except PRs with tiny changes) - [x] Changes are complete (i.e. I finished coding on this PR) - [x] All changes have test coverage: - Ran test_docker_cache.py, ensured both existing tests pass. - [x] Code is well-documented: - [x] To the my best knowledge, examples are either not affected by this change, or have been fixed to be compatible with this change

17. **title:** [MXNET-553] Restructure dockcross dockerfiles to fix caching

body: ## Description ## This PR restructures the dockcross Dockerfiles to remove their multi-head components. These multi-head components made it difficult to properly provide a remote cache for the Dockerfiles, and was slowing down builds. A simple work-around for this was to generalize the ubuntu_ccache.sh file such that it works with ubuntu, debian, and in cross-compilation containers. We can now use this script the same way in a variety of containers, which eliminates the need for a special multi-head docker stage. Should address #11257. ## Checklist ## #### Essentials #### Please feel free to remove inapplicable items for your PR. - [x] The PR title starts with [MXNET-\$JIRA_ID], where \$JIRA_ID refers to the relevant [JIRA issue](https://issues.apache.org/jira/projects/MXNET/issues) created (except PRs with tiny changes) - [x] Changes are complete (i.e. I finished coding on this PR) - [x] All changes have test coverage: - Ran test_docker_cache.py, ensured both existing tests pass. - [x] Code is well-documented: - [x] To the my best knowledge, examples are either not affected by this change, or have been fixed to be compatible with this change

18. **title:** [MXNET-553] Restructure dockcross dockerfiles to fix caching

body: ## Description ## This PR restructures the dockcross Dockerfiles to remove their multi-head components. These multi-head components made it difficult to properly provide a remote cache for the Dockerfiles, and was slowing down builds. A simple work-around for this was to generalize the ubuntu_ccache.sh file such that it works with ubuntu, debian, and in cross-compilation containers. We can now use this script the same way in a variety of containers, which eliminates the need for a special multi-head docker stage. Should address #11257. ## Checklist ## #### Essentials #### Please feel free to remove inapplicable items for your PR. - [x] The PR title starts with [MXNET-\$JIRA_ID], where \$JIRA_ID refers to the relevant [JIRA issue](https://issues.apache.org/jira/projects/MXNET/issues) created (except PRs with tiny changes) - [x] Changes are complete (i.e. I finished coding on this PR) - [x] All changes have test coverage: - Ran test_docker_cache.py, ensured both existing tests pass. - [x] Code is well-documented: - [x] To the my best knowledge, examples are either not affected by this change, or have been fixed to be compatible with this change

19. **title:** [MXNET-553] Restructure dockcross dockerfiles to fix caching

body: ## Description ## This PR restructures the dockcross Dockerfiles to remove their multi-head components. These multi-head components made it difficult to properly provide a remote cache for the Dockerfiles, and was slowing down builds. A simple work-around for this was to generalize the ubuntu_ccache.sh file such that it works with ubuntu, debian, and in cross-compilation containers. We can now use this script the same way in a variety of containers, which eliminates the need for a special multi-head docker stage. Should address #11257. ## Checklist ## #### Essentials #### Please feel free to remove inapplicable items for your PR. - [x] The PR title starts with [MXNET-\$JIRA_ID], where \$JIRA_ID refers to the relevant [JIRA issue](https://issues.apache.org/jira/projects/MXNET/issues) created (except PRs with tiny changes) - [x] Changes are complete (i.e. I finished coding on this PR) - [x] All changes have test coverage: - Ran test_docker_cache.py, ensured both existing tests pass. - [x] Code is well-documented: - [x] To the my best knowledge, examples are either not affected by this change, or have been fixed to be compatible with this change

20. **title:** [MXNET-553] Restructure dockcross dockerfiles to fix caching

body: ## Description ## This PR restructures the dockcross Dockerfiles to remove their multi-head components. These multi-head components made it difficult to properly provide a remote cache for the Dockerfiles, and was slowing down builds. A simple work-around for this was to generalize the ubuntu_ccache.sh file such that it works with ubuntu, debian, and in cross-compilation containers. We can

now use this script the same way in a variety of containers, which eliminates the need for a special multi-head docker stage. Should address #11257. ## Checklist ## #### Essentials #### Please feel free to remove inapplicable items for your PR. - [x] The PR title starts with [MXNET-\$JIRA_ID], where \$JIRA_ID refers to the relevant [JIRA issue](https://issues.apache.org/jira/projects/MXNET/issues) created (except PRs with tiny changes) - [x] Changes are complete (i.e. I finished coding on this PR) - [x] All changes have test coverage: - Ran test_docker_cache.py, ensured both existing tests pass. - [x] Code is well-documented: - [x] To the my best knowledge, examples are either not affected by this change, or have been fixed to be compatible with this change

label: code-design

21. **title:** [MXNET-553] Restructure dockcross dockerfiles to fix caching

body: ## Description ## This PR restructures the dockcross Dockerfiles to remove their multi-head components. These multi-head components made it difficult to properly provide a remote cache for the Dockerfiles, and was slowing down builds. A simple work-around for this was to generalize the ubuntu_ccache.sh file such that it works with ubuntu, debian, and in cross-compilation containers. We can now use this script the same way in a variety of containers, which eliminates the need for a special multi-head docker stage. Should address #11257. ## Checklist ## #### Essentials #### Please feel free to remove inapplicable items for your PR. - [x] The PR title starts with [MXNET-\$JIRA_ID], where \$JIRA_ID refers to the relevant [JIRA issue](https://issues.apache.org/jira/projects/MXNET/issues) created (except PRs with tiny changes) - [x] Changes are complete (i.e. I finished coding on this PR) - [x] All changes have test coverage: - Ran test_docker_cache.py, ensured both existing tests pass. - [x] Code is well-documented: - [x] To the my best knowledge, examples are either not affected by this change, or have been fixed to be compatible with this change

github_pulls_comments:

1. Would appreciate a review from @marcoabreu and @lebeg.
2. Excellent work @KellenSunderland! Do you think there is any need to make the centos installations also robust to compiler changes? Maybe mentioning https://github.com/apache/incubator-mxnet/issues/11257 would be good as well.
3. **body:** @lebeg I think calling the compiler with \$CC should be sufficient for CentOS, at least for the time being. We can always update in the future if needed. @larroy I think there's a feature request that will make this easier in the future. Hopefully v19 will include the feature req. It is certainly a breaking change from versions prior to 17, but it's made as a response to some valid security concerns. There's also a work around, but I think implementing this would add quite a lot of complexity.
label: code-design
4. Great job, Kellen! Thanks a lot! I think the ccache is now not being applied to all builds. Examples: <http://jenkins.mxnet-ci.amazon-ml.com/blue/organizations/jenkins/incubator-mxnet/detail/PR-11302/2/pipeline/59> <http://jenkins.mxnet-ci.amazon-ml.com/blue/organizations/jenkins/incubator-mxnet/detail/PR-11302/2/pipeline/54> On some others, it works: <http://jenkins.mxnet-ci.amazon-ml.com/blue/organizations/jenkins/incubator-mxnet/detail/PR-11302/2/pipeline/52> <http://jenkins.mxnet-ci.amazon-ml.com/blue/organizations/jenkins/incubator-mxnet/detail/PR-11302/2/pipeline/51> Have a look at the execution durations to see whether ccache was actually used or not. Things that still don't work and haven't worked before are GPU builds. Those are a separate issue.
5. LGTM I can try what I asked myself as I'm not sure my question was well understood.
6. @larroy Sounds good, we can also chat offline about if it in more detail if I've misunderstood.
@marcoabreu Not sure I fully understand the examples you've given. Do they relate to the issue that this PR addresses (docker cache misses?).
7. Yes, I got the feeling that your changes are causing some builds to not use ccache anymore. It'd be good if you could verify that
8. @marcoabreu Gotcha. I think what would be useful would be to include some more information about what ccache is doing into the logs. I'll see if I can create a (separate) PR to do that, and then rebase.
9. That's an excellent idea! Maybe just printing the cache hit/miss statistics after a build (not globally using ccache -s as it will include everything).
10. Could you elaborate Pedro? Kellen removes the multi head in his PR while your log seems to be referencing a dockerfile that still has it. Could it be possible you checked out the wrong commit?
11. @marcoabreu @larroy: No actually the error Pedro pointed out was a valid error. I've fixed it and am investigating why the build succeeded earlier. Edit: looks like the earlier run didn't tests Android/64 so that would explain it. Thanks for the testing Pedro.

12. Ah your PR did not change Android Arm64 because we just merged it and that caused the race condition in the git merge. Thanks for elaborating
13. Any other data you want to see here @marcoabreu ? No rush on the merge but lmk if you need anything else.
14. I compared the two following runs: <http://jenkins.mxnet-ci.amazon-ml.com/blue/organizations/jenkins/incubator-mxnet/detail/PR-11302/6/pipeline/58> <http://jenkins.mxnet-ci.amazon-ml.com/blue/organizations/jenkins/incubator-mxnet/detail/master/1012/pipeline/56/> I noticed the following slow downs - partially quite significantly: CPU clang 3.9, CPU clang 3.9 MKLDNN, CPU clang 5.0, CPU clang 5.0 MKLDNN, CPU MKLDNN, CPU: Openblas, GPU CMake, GPU CMake MKLDNN, GPU CUDA 9.1, GPU MKLDNN, NVidia Jetson ARMv8 The other ones have not been slowed down or improved. I think we should investigate this before we merge. What I noticed is that the CentOS ones are still exactly the same while almost all other jobs got slower by up to factors like 10x
15. Right but why would we see speed ups before the docker cache is pushed? On Tue, Jun 19, 2018, 7:48 AM Marco de Abreu <notifications@github.com> wrote: > I compared the two following runs: > > <http://jenkins.mxnet-ci.amazon-ml.com/blue/organizations/jenkins/incubator-mxnet/detail/PR-11302/6/pipeline/58> > > <http://jenkins.mxnet-ci.amazon-ml.com/blue/organizations/jenkins/incubator-mxnet/detail/master/1012/pipeline/56/> > > I noticed the following slow downs - partially quite significantly: > CPU clang 3.9, CPU clang 3.9 MKLDNN, CPU clang 5.0, CPU clang 5.0 MKLDNN, > CPU MKLDNN, CPU: Openblas, GPU CMake, GPU CMake MKLDNN, GPU CUDA 9.1, GPU > MKLDNN, NVidia Jetson ARMv8 > > The other ones have not been slowed down or improved. > > I think we should investigate this before we merge. What I noticed is that > the CentOS ones are still exactly the same while almost all other jobs got > slower by up to factors like 10x > > — > You are receiving this because you were mentioned. > Reply to this email directly, view it on GitHub > <<https://github.com/apache/incubator-mxnet/pull/11302#issuecomment-398281560>>, > or mute the thread > <<https://github.com/notifications/unsubscribe-auth/AHGTE4B-17x3IR7zmYuG7Xsri9n74YR4ks5t-JCxgaJpZM4Upb4P>> > . >
16. Oops, I forgot to take that time into account. Thank you and sorry for the inconvenience.
17. Thanks man, appreciate it. I'll make sure and monitor master for a couple days and verify that the cache is being utilized.

github_pulls_reviews:

1. Can't we set this environment variable in the docker file and then unset it? Since it's specific to dockcross.
2. **body:** We want to ensure we're maintaining the CC and CXX env vars from the 'dockcross/linux-armv7' image, so to do this we'd have to store those and then restore them. Seems more complicated than just overriding them for a single command to me. What would be the advantage, readability?
label: code-design

jira_issues:

1. **summary:** Restructure dockcross dockerfiles to fix caching
description: Restructure the dockcross Dockerfiles to remove their multi-head components. This multi-head component made it difficult to properly provide a remote cache for the Dockerfiles, and was slowing down builds. A simple work-around for this was to generalize the ubuntu_ccache.sh file such that it works with ubuntu, debian, and in cross-compilation containers. We can now use this script the same way in a variety of containers, which eliminates the need for a special multi-head docker stage.
label: code-design
2. **summary:** Restructure dockcross dockerfiles to fix caching
description: Restructure the dockcross Dockerfiles to remove their multi-head components. This multi-head component made it difficult to properly provide a remote cache for the Dockerfiles, and was slowing down builds. A simple work-around for this was to generalize the ubuntu_ccache.sh file such that it works with ubuntu, debian, and in cross-compilation containers. We can now use this script the same way in a variety of containers, which eliminates the need for a special multi-head docker stage.

jira_issues_comments: