# Objective

This document is designed to provide information on the standards to the branch names and processes for the branches in source control.

## Merge Process

#### Development

* Database DevOps team is responsible for merging changes into the Development Branch
* Merge is done using Pull Requests
  + Pull Requests need to have a Note that informs Database Devops the assigned release for the code
  + This note is found in the tag for the Rational User Story

#### Release\_ActiveRelease(the name changes Example Release\_2020\_Drake)

* Database DevOps team is responsible for merging changes into the Release Branch
* Not all repositories follow this standard
  + Affected repositories are
    - NPAD
    - Protracts

#### Release\_InactiveRelease(the name changes Example Release\_2020\_Hops)

* Database DevOps team is responsible for merging changes into the Release Branch
* Not all repositories follow this standard
  + Affected repositories are
    - NPAD
    - Protracts

#### Master

* Database DevOps team is responsible for merging changes into the Master Branch
* There are different triggers for merges
  + Merge is done using Pull Requests
  + Database DevOps will merge after production releases

## Development Branch

* This branch is used for any new development
* Database Devops will perform daily merges from Develop
* Most teams will be using this branch

## Release Branch

* Branch managed by the DevOps Team
* Most repositories this will be a static name
* For NPAD and Protracts
  + - This branch name will change per release schedule we are using (ex. Release\_2020\_A)
    - This name should match with what is set in the Rational User Story
  + For code that is assigned to the future release (ex. Release\_2020\_B)
    - If code cut off is not completed, then the developer will create a long running Feature Branch from Development
      * The developer will do daily Pulls from the Development branch to keep the feature branch up to date
    - If code cut off is completed, then the Developer will use the Feature Branch from Development Branch
    - Code cut off will be determined on a as needed basis
      * When the features are complete
      * A date is given, this is not always given
      * If teams decide that we have enough future release work to create the new branch

## Master Branch

* Branch managed by the DevOps Team
* Production Hotfixes
  + Developer will create a Feature Branch from Master
  + Developer will create a Pull Request when ready to merge into Master
* Devops Team will perform downward merges to other branches



## Release Stacks (Protracts and NPAD only subject to changes)

This could change but it is the information the Brian Bode gave me. Here is the A and B release stacks

Release A

* Dev (we do not manage this instance)
* Test
* Int
* Uat
* Load? (may need to include Load in release A, but currently targeting B)
* **Train**
* **Prod**

Release B

* Dev2 (we do not manage this instance)
* Test2
* Int? (may change the purpose of Int to be a ‘next release’ anchor (release B) for automated deployments)
* Load