

브랜치 관리 git-flow

Git & Github
for pull request

강환수 교수



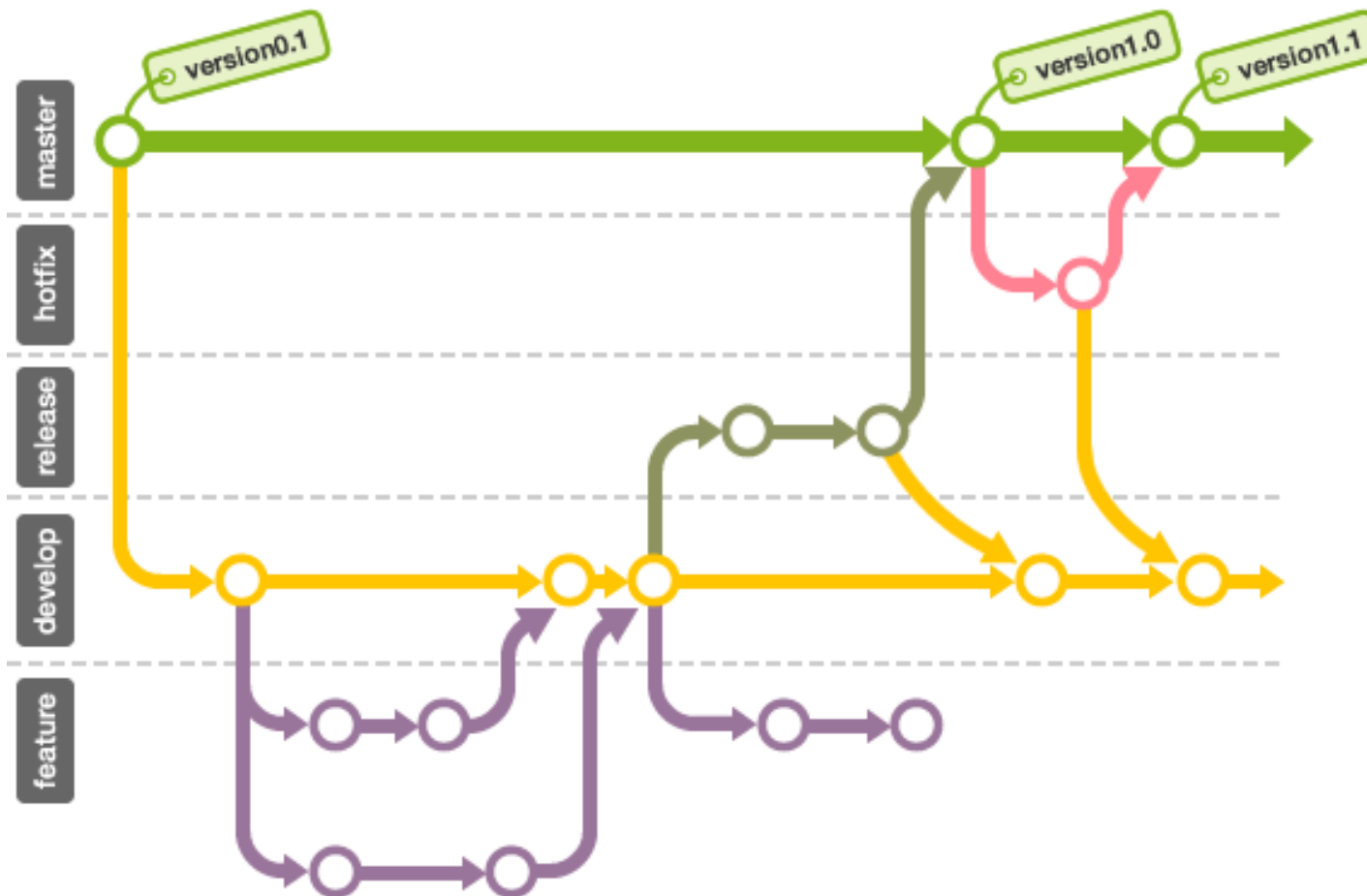
AI Experts
Who Lead
The Future

01

프로젝트 개발 브랜치 종류

- Goal

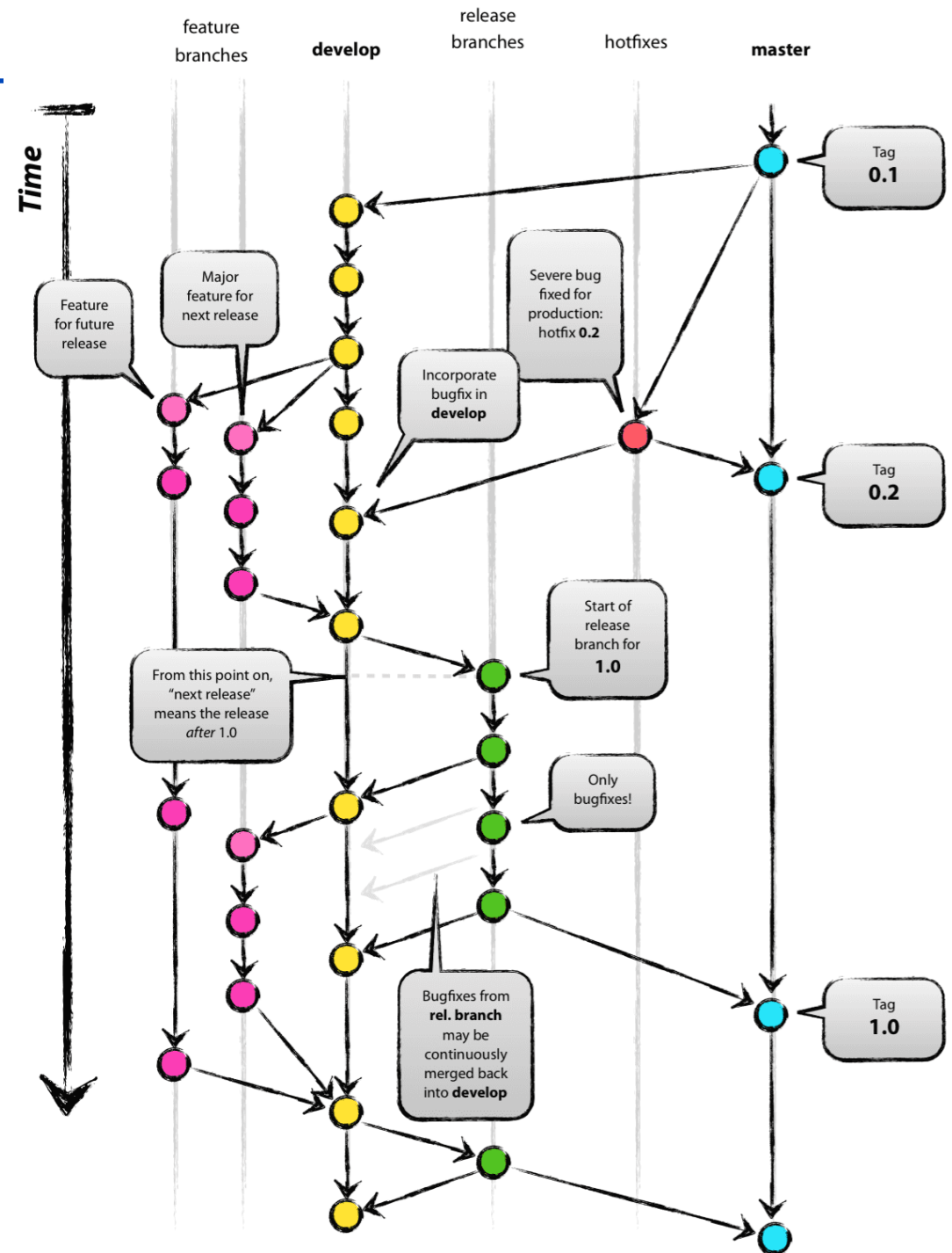
- Gitflow Workflow에서 사용하는 Git Branch 종류를 이해
- Gitflow Workflow에서 사용하는 Git Branch 사용법을 이해



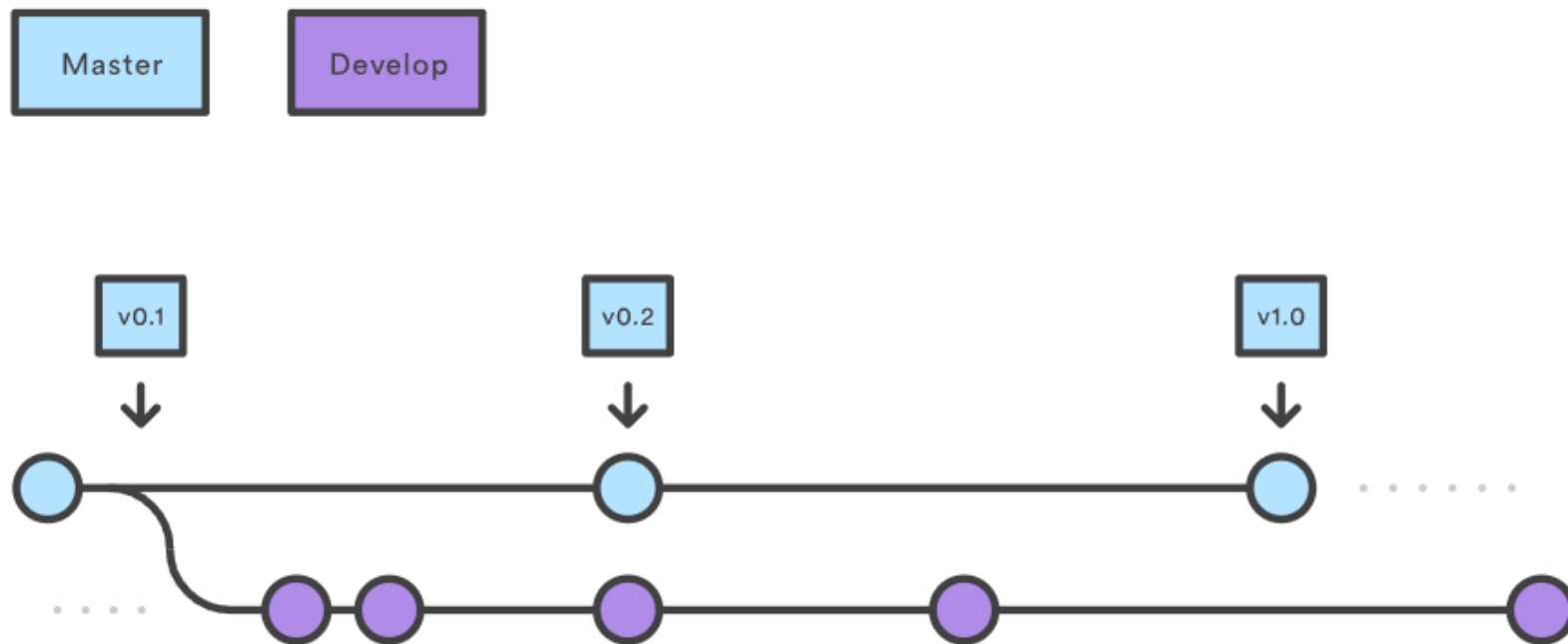
Git Branch 종류

- 총 5가지의 브랜치

- 항상 유지되는 메인 브랜치들
 - **main(master), develop**
- 일정 기간 동안만 유지되는 보조 브랜치들
 - **feature, release, hotfix**

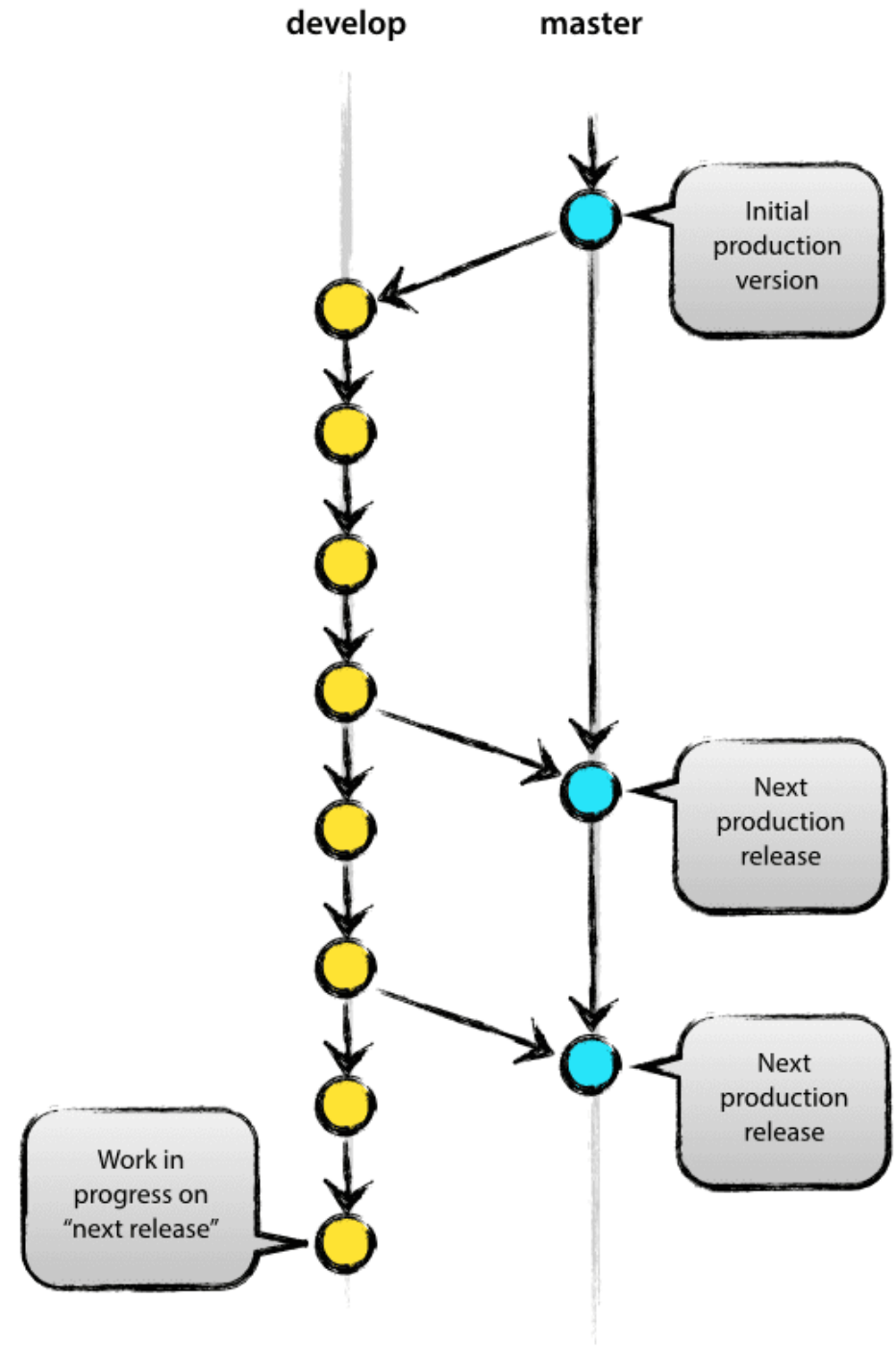


- 제품으로 출시되는 브랜치 배포(Release) 이력을 관리하기 위해 사용
 - 즉, 배포 가능한 상태만을 관리
 - 커밋할 때에는 태그를 사용하여 배포 번호를 기록

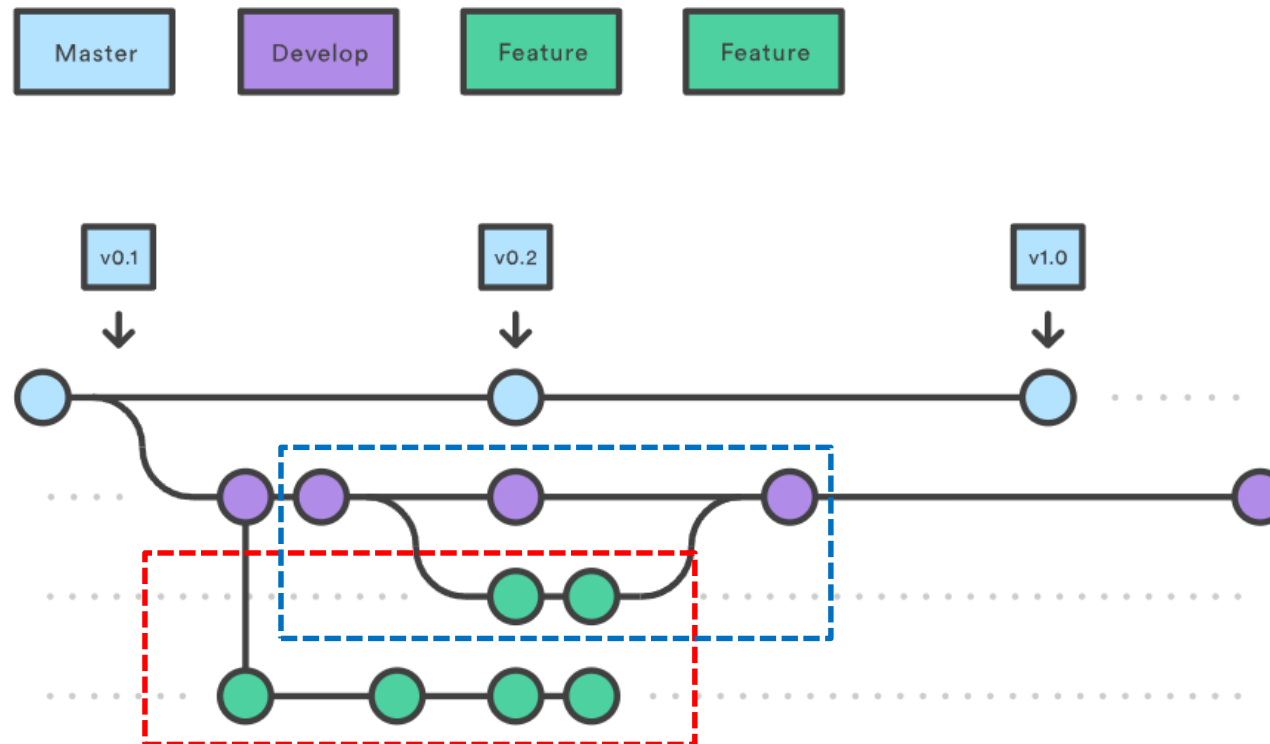


Develop branch

- 다음 출시 버전을 개발하는 브랜치
 - 기능 개발을 위한 브랜치들을 병합하기 위해 사용
 - 평소에는 이 브랜치를 기반으로 개발을 진행
 - 즉, 모든 기능이 추가되고 버그가 수정되어 배포 가능한 안정적인 상태
 - develop 브랜치를 'master' 브랜치에 병합(merge)



- 기능을 개발하는 브랜치
 - 새로운 기능 개발 및 버그 수정이 필요할 때마다 'develop' 브랜치에서 분기
 - 자신의 로컬 저장소에서 관리
 - feature 브랜치에서의 작업은 기본적으로 공유할 필요가 없기 때문
- develop 브랜치에서 개발이 완료되면
 - develop 브랜치로 병합(merge)하여 다른 사람들과 공유

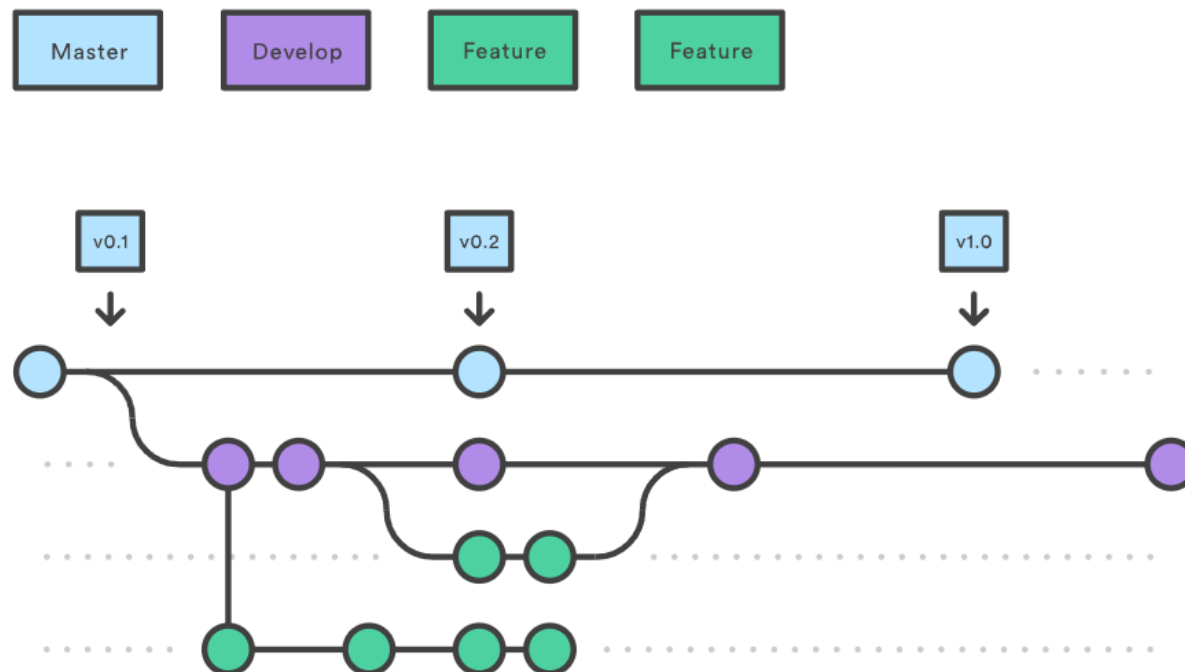


5 과정

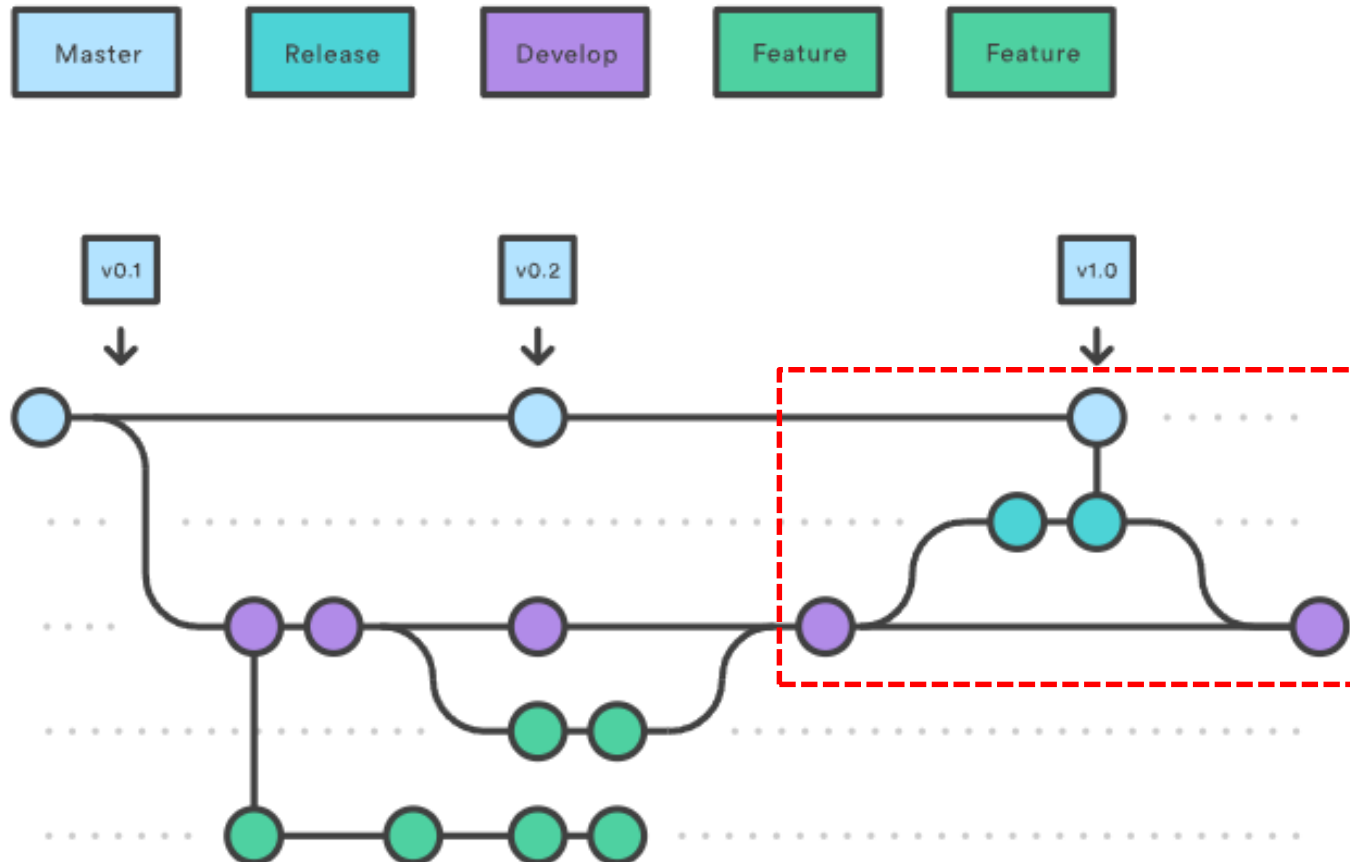
- 1. 'develop' 브랜치에서 새로운 기능에 대한 feature 브랜치를 분기
- 2. 새로운 기능에 대한 작업 수행
- 3. 작업이 끝나면 'develop' 브랜치로 병합(merge)
- 4. 더 이상 필요하지 않은 feature 브랜치는 삭제
- 5. 새로운 기능에 대한 'feature' 브랜치를 중앙 원격 저장소에 push

feature 브랜치 이름 정하기

- [feature/기능요약] 형식을 추천
 - feature/login



- 이번 출시 버전을 준비하는 브랜치
 - 배포를 위한 전용 브랜치를 사용
 - 한 팀이 해당 배포를 준비하는 동안 다른 팀은 다음 배포를 위한 기능 개발이 계속 가능
- 딱딱 끊어지는 개발 단계를 정의하기에 good
 - '이번 주에 버전 1.0 배포를 목표로 한다!'



절차 1: release 브랜치를 분기

- 시기

- 'develop' 브랜치에서 배포할 수 있는 수준의 기능이 모이면
- 또는 정해진 배포 일정이 되면

- 절차

- release 브랜치를 만드는 순간부터 배포 사이클이 시작
 - 배포를 위한 최종적인 버그 수정, 문서 추가 등 릴리스와 직접적으로 관련된 작업을 수행
- release 브랜치에 새로운 기능을 추가로 병합(merge)하지 않음

절차 2: 'release' 브랜치에서 배포 가능한 상태가 되면

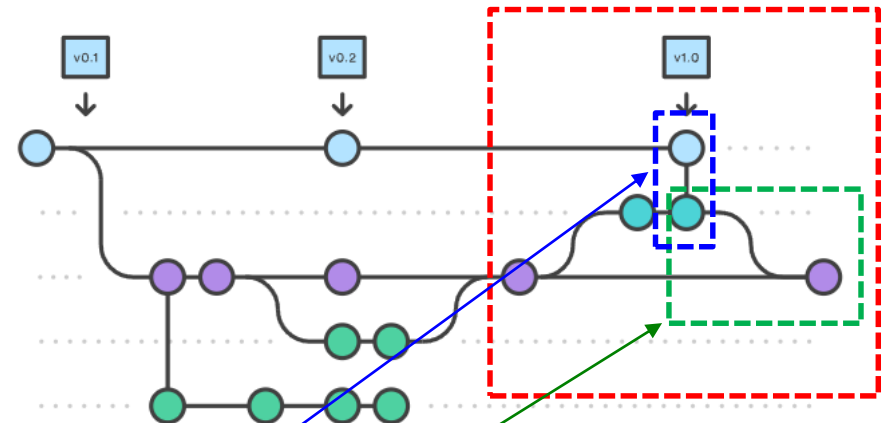
- 배포 가능한 상태(배포 준비가 완료 상태)

- 새로운 기능을 포함한 상태로 모든 기능이 정상적으로 동작하는 상태
- 'master' 브랜치에 병합

- 이때, 병합한 커밋에 Release 버전 태그를 부여!

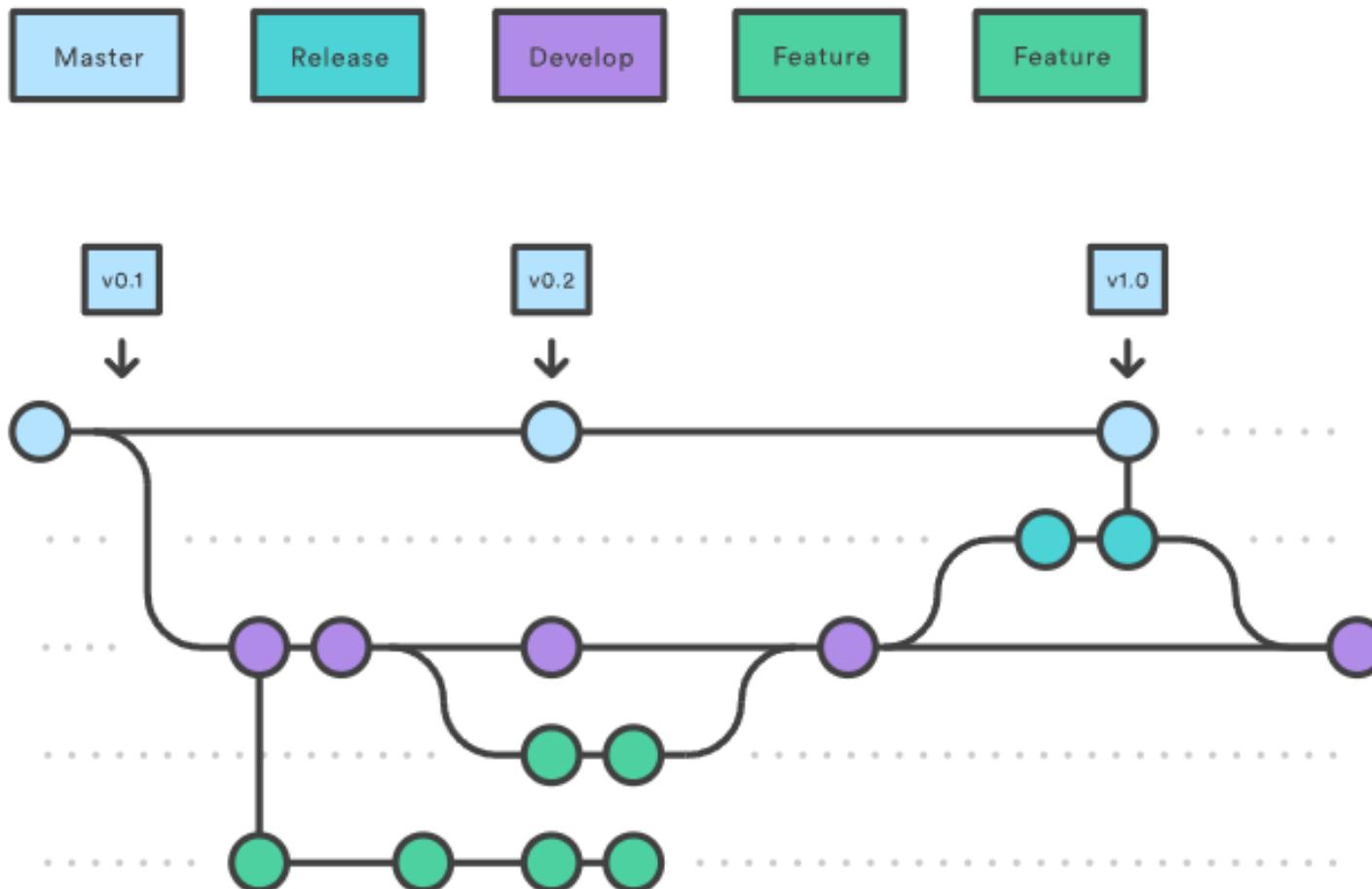
- 배포를 준비하는 동안 release 브랜치가 변경되었을 수 있으므로 배포 완료 후 'develop' 브랜치에도 병합

- 다음 번 배포(Release)를 위한 개발 작업은 'develop' 브랜치에서 계속 진행



- release 브랜치 이름 정하기

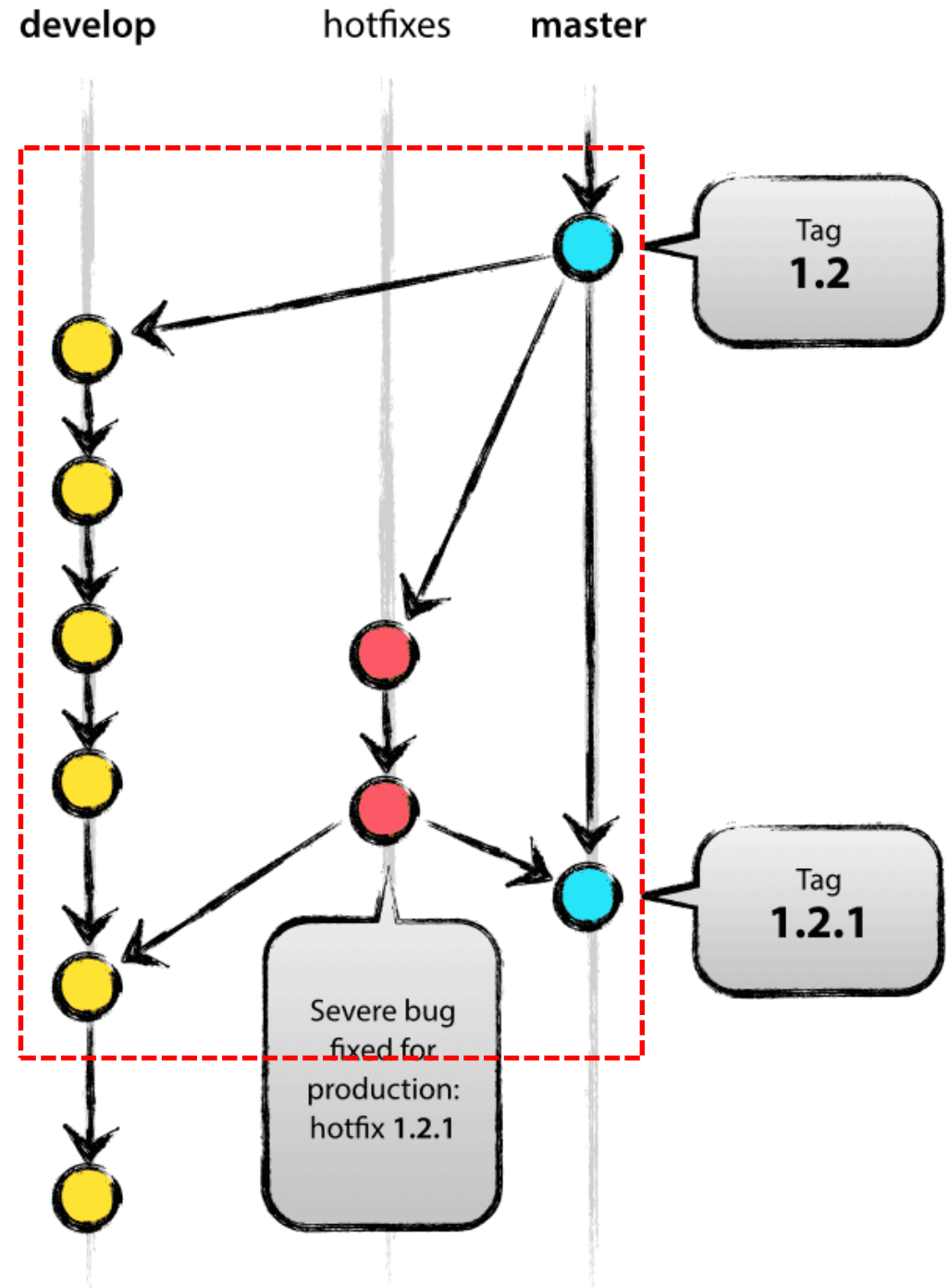
- release-RB_* 또는 release-* 또는 release/* 처럼 이름 짓는 것이 일반적인 관례
- [release-*] 형식을 추천
 - release-1.2



- 버그를 수정하거나 새로운 기능을 포함한 상태로 모든 기능이 정상적으로 동작하는지 확인
- 릴리즈 브랜치의 이름
 - 관례적으로 브랜치 이름 앞에 'release-' 를 붙임
 - 다음 번 릴리즈를 위한 개발 작업은 'develop' 브랜치에서 계속 진행
- 모든 준비를 마치고 배포 가능한 상태가 되면 'master' 브랜치로 병합
 - 병합한 커밋에 릴리즈 번호 태그를 추가
- 릴리즈 브랜치에서 기능을 점검하며 발견한 버그 수정 사항은 'develop' 브랜치에도 적용
 - 그러므로 배포 완료 후 'develop' 브랜치에 대해서도 병합 작업을 수행

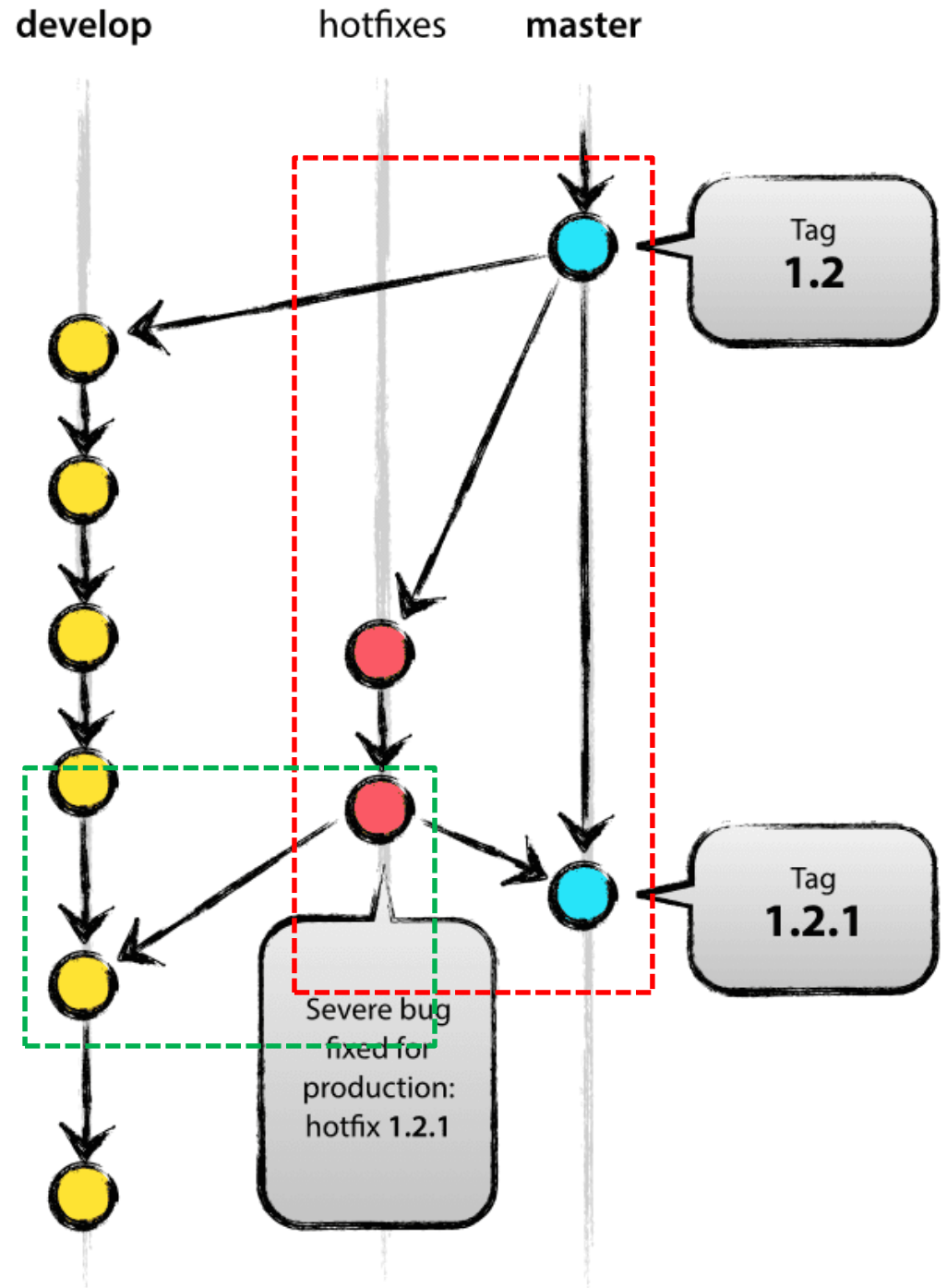
Hotfix Branch 1/3

- 출시 버전에서 발생한 버그를 수정하는 브랜치
 - 배포한 버전에 긴급하게 수정을 해야 할 필요가 있을 경우
 - 'master' 브랜치에서 분기하는 브랜치
 - 바로 배포가 가능한 'master' 브랜치에서 직접 브랜치를 만들어
 - 필요한 부분만을 수정한 후 다시 'master' 브랜치에 병합하여 이를 배포
- 'develop' 브랜치에서 분기하지 않는 이유
 - 'develop' 브랜치에서 문제가 되는 부분을 수정하여 배포 가능한 버전을 만들기에는
 - 시간도 많이 소요되고 안정성을 보장하기도 어려우므로



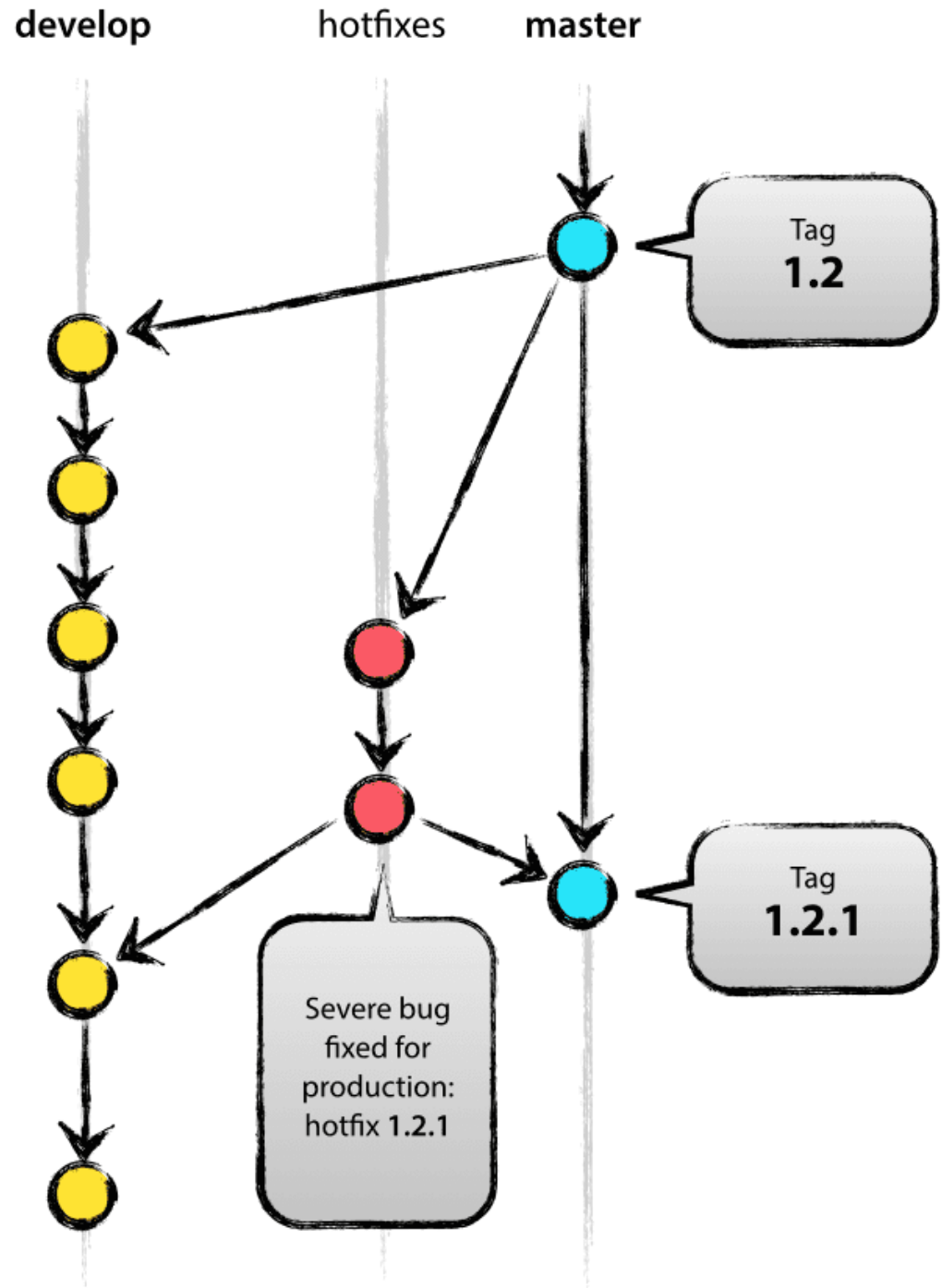
Hotfix Branch 2/3

- 배포한 버전에 긴급하게 수정을 해야 할 필요가 있을 경우
 - 'master' 브랜치에서 hotfix 브랜치를 분기
 - 'hotfix' 브랜치만 master에서 바로 분기
 - 문제가 되는 부분만을 빠르게 수정
 - 긴급 수정 내용을 안정적으로 다시 배포
 - 'master' 브랜치에 병합(merge)
 - 새로운 버전 이름으로 태그를 설정
 - 'develop' 브랜치에도 병합(merge)
 - hotfix 브랜치에서의 변경 사항



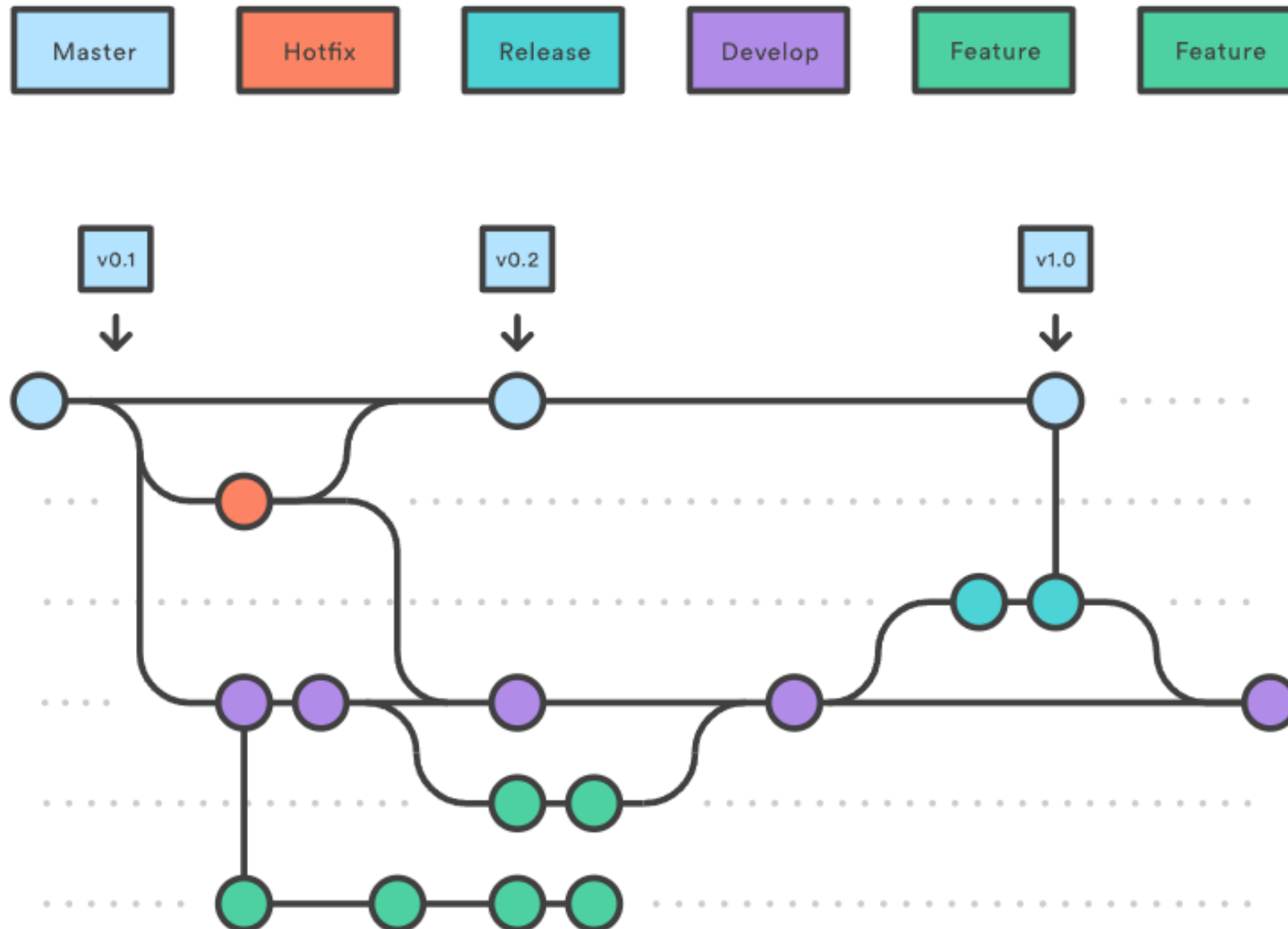
Hotfix Branch 3/3

- 버그 수정만을 위한 'hotfix' 브랜치를 따로 만들었기 때문에
 - 다음 배포를 위해 개발하던 작업 내용에 전혀 영향을 주지 않음
 - 'hotfix' 브랜치는 master 브랜치를 부모로 하는 임시 브랜치라고 생각
- hotfix 브랜치 이름 정하기
 - [hotfix-*] 형식을 추천
 - EX) hotfix-1.2.1



Branch 전체 흐름 리뷰

오픈소스 소프트웨어를 위한 깃과 깃허브 Python language

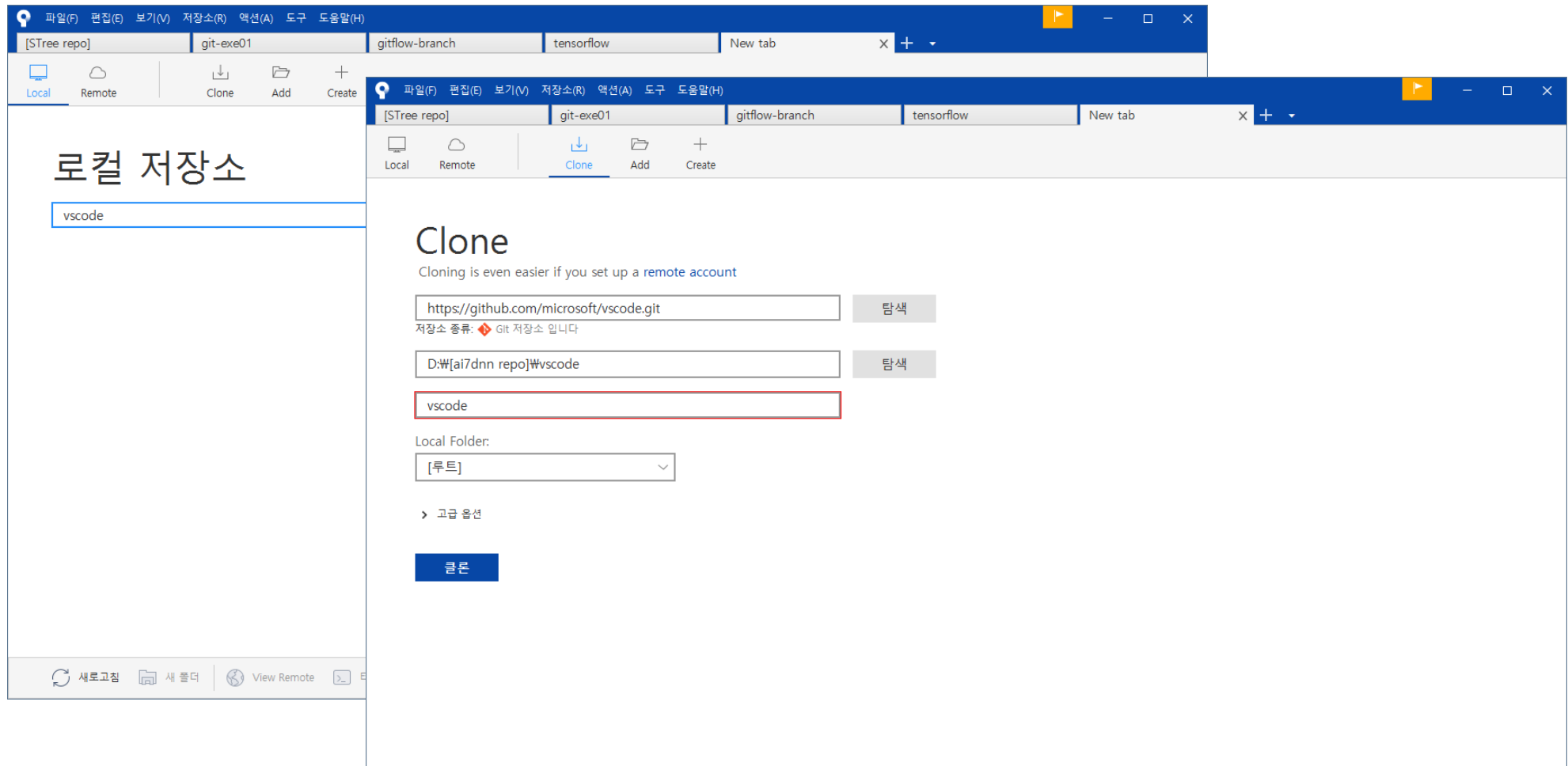


AI Experts
Who Lead
The Future

02

소스트리로
깃허브 브랜치 확인

- 비주얼 스튜디오 코드
 - <https://github.com/microsoft/vscode.git>
- 저장소 폴더를 만들어 “탐색”을 클릭해 지정
 - 클론 클릭



소스트리에서의 vscode 브랜치 흐름

오픈소스 소프트웨어를 위한 깃과 깃허브 Python language

The screenshot shows the VS Code interface with the Git extension. The '모든 브랜치' (All Branches) view is active, displaying a list of commits and their details. The selected commit is 'Revert "Revert "show reload button when extension with different targetplatform is updated"' by Sandeep Somavarapu.

날짜	작업자	커밋
12 3 2022 19:13	Benjamin Pasero <benjamin.pasero@microsoft.com>	3eae27e2c1
12 3 2022 19:07	Benjamin Pasero <benjamin.pasero@microsoft.com>	ca6abd6347
12 3 2022 19:02	Benjamin Pasero <benjamin.pasero@microsoft.com>	4d4d10c0f7
12 3 2022 16:23	Benjamin Pasero <benjamin.pasero@microsoft.com>	3e5c7e2c5f
12 3 2022 11:51	meganrogge <mrogge@microsoft.com>	7fda221774
12 3 2022 10:34	Megan Rogge <mrogge@microsoft.com>	8ca417c5ab
12 3 2022 9:43	Raymond Zhao <raymondzhao@microsoft.com>	3c2a690847
12 3 2022 9:37	Rob Lourens <roblourens@gmail.com>	e4cf231274
12 3 2022 9:02	SteVen Batten <sbatten@microsoft.com>	58966a981f
12 3 2022 8:59	SteVen Batten <sbatten@microsoft.com>	421e8bd7a7
12 3 2022 8:48	Raymond Zhao <raymondzhao@microsoft.com>	57f703ac05
12 3 2022 7:57	meganrogge <mrogge@microsoft.com>	a32a0f5dd4
12 3 2022 7:42	meganrogge <mrogge@microsoft.com>	ad79ee466k
12 3 2022 7:29	Daniel Imms <2193314+Tyriar@users.noreply.g>	b38f0a6b97
12 3 2022 7:17	Daniel Imms <2193314+Tyriar@users.noreply.g>	3164ed22a6
12 3 2022 7:12	Alexandru Dima <alexdimam@microsoft.com>	5f5e1ca791
12 3 2022 7:08	Raymond Zhao <raymondzhao@microsoft.com>	de784f76f7
12 3 2022 6:53	Daniel Imms <2193314+Tyriar@users.noreply.g>	5738c241b7
12 3 2022 6:17	Matt Bierner <matb@microsoft.com>	a1f4c800b4
12 3 2022 6:17	Matt Bierner <matb@microsoft.com>	b83e48cf00
12 3 2022 6:14	Alex Dima <alexdimam@microsoft.com>	cf8969e450
12 3 2022 6:08	Sandeep Somavarapu <sasomava@microsoft.cc>	5a7b43ee2f
12 3 2022 6:08	Sandeep Somavarapu <sasomava@microsoft.cc>	cf4fe193cf6
12 3 2022 6:08	Sandeep Somavarapu <sasomava@microsoft.cc>	7b42b0502
12 3 2022 6:08	Sandeep Somavarapu <sasomava@microsoft.cc>	4cc9f9ee2f
12 3 2022 6:08	Sandeep Somavarapu <sasomava@microsoft.cc>	0006ca4db7
12 3 2022 6:07	Alex Dima <alexdimam@microsoft.com>	5b3c42cb2f

커밋: 5a7b43ee2f12efb347cf00c736e51f54fc408303 [5a7b43e]
 상위 항목: cf4fe193cf
 작성자: Sandeep Somavarapu <sasomava@microsoft.com>
 날짜: 2022년 3월 11일 금요일 오후 9:54:12
 커밋한 사람: Sandeep Somavarapu
 커밋 날짜: 2022년 3월 11일 금요일 오후 10:08:10

Revert "Revert "show reload button when extension with different targetplatform is updated""

This reverts commit dfd530095900d98bdeaa5cde514e00b75b8177e.

src/vs/workbench/contrib/extensions/browser/extensionsActions.ts

```

1427 1427 .....const runningExtensionServer = this.extensionManagementServer
1428 1428 .....
1429 1429 .....if (isSameExtensionRunning) {
1430 1430 .....    // Different version of same extension is running. Require
1431 1431 .....    // Different version or target platform of same extension
1432 1432 .....    // Different version or target platform of same extension
1433 1433 .....    if (this.extension.version !== runningExtension.version) {
1434 1434 .....        this.enabled = true;
1435 1435 .....        this.label = localize('reloadRequired', 'Reload Required');
1436 1436 .....        this.tooltip = localize('postUpdateTooltip', 'Please
    
```

소스트리에서의 tensorflow 브랜치 흐름

오픈소스 소프트웨어를 위한 깃과 깃허브 Python language

The screenshot displays the GitFlow branch model in a code editor. The interface is divided into several sections:

- Workspace:** Located on the left, it includes a file explorer showing the project structure with files like README.md, develop.txt, file.txt, and release-file.txt.
- Commit History:** A table in the center lists commits, including merges, updates, and initial commits, with columns for commit hash, author, date, and message.
- File Viewer:** The bottom right section shows the content of the README.md file, which includes instructions for setting up the development environment.

The commit history table is as follows:

Commit Hash	Author	Date	Message
d066a1c	ai7dnn <70050528+ai7dnn@users.noreply.github.com>	13 3 2022 11:10	Initial commit
0e7fc70	ai7dnn <70050528+ai7dnn@users.noreply.github.com>	12 3 2022 17:26	Update README.md
ab7c051	ai7dnn <70050528+ai7dnn@users.noreply.github.com>	12 3 2022 17:28	Update README.md
48f751c	ai7dnn <70050528+ai7dnn@users.noreply.github.com>	13 3 2022 9:20	Update README.md
9beffaf	ai7dnn <70050528+ai7dnn@users.noreply.github.com>	13 3 2022 9:24	Merge pull request #2 from ai7dnn/release/file
e71b359	ai7dnn <70050528+ai7dnn@users.noreply.github.com>	13 3 2022 10:08	Update README.md
1bf3f9f	ai7dnn <70050528+ai7dnn@users.noreply.github.com>	13 3 2022 10:10	Create develop
f066f98	ai7dnn <70050528+ai7dnn@users.noreply.github.com>	13 3 2022 10:20	Update README.md
b885257	ai7dnn <70050528+ai7dnn@users.noreply.github.com>	13 3 2022 10:22	Create file.txt
bec0424	ai7dnn <70050528+ai7dnn@users.noreply.github.com>	13 3 2022 10:22	Update README.md
f87b5c3	ai7dnn <70050528+ai7dnn@users.noreply.github.com>	13 3 2022 10:28	Update README.md
7f7c506	ai7dnn <70050528+ai7dnn@users.noreply.github.com>	13 3 2022 10:29	Create release-file.txt
a200f9a	ai7dnn <70050528+ai7dnn@users.noreply.github.com>	13 3 2022 10:30	Update README.md
40285d3	ai7dnn <70050528+ai7dnn@users.noreply.github.com>	13 3 2022 11:10	Merge

The README.md file content is as follows:

```

1 1 부분: 2-12 줄
2 2 ...learning gitflow of branch
3 3 ...
4 4 ...### main init
5 5 +
6 6 ...### hotfix/issue01 init
7 7 +
8 8 +### develop init
9 9 +### feature/file init
10 10 +### feature/file exit
11 11 +### release/file init
12 12 +### release/file exit
    
```