

The background features abstract, overlapping geometric shapes in various shades of blue, ranging from light sky blue to deep navy blue. These shapes are primarily located on the left and right sides of the frame, creating a modern, tech-oriented aesthetic.

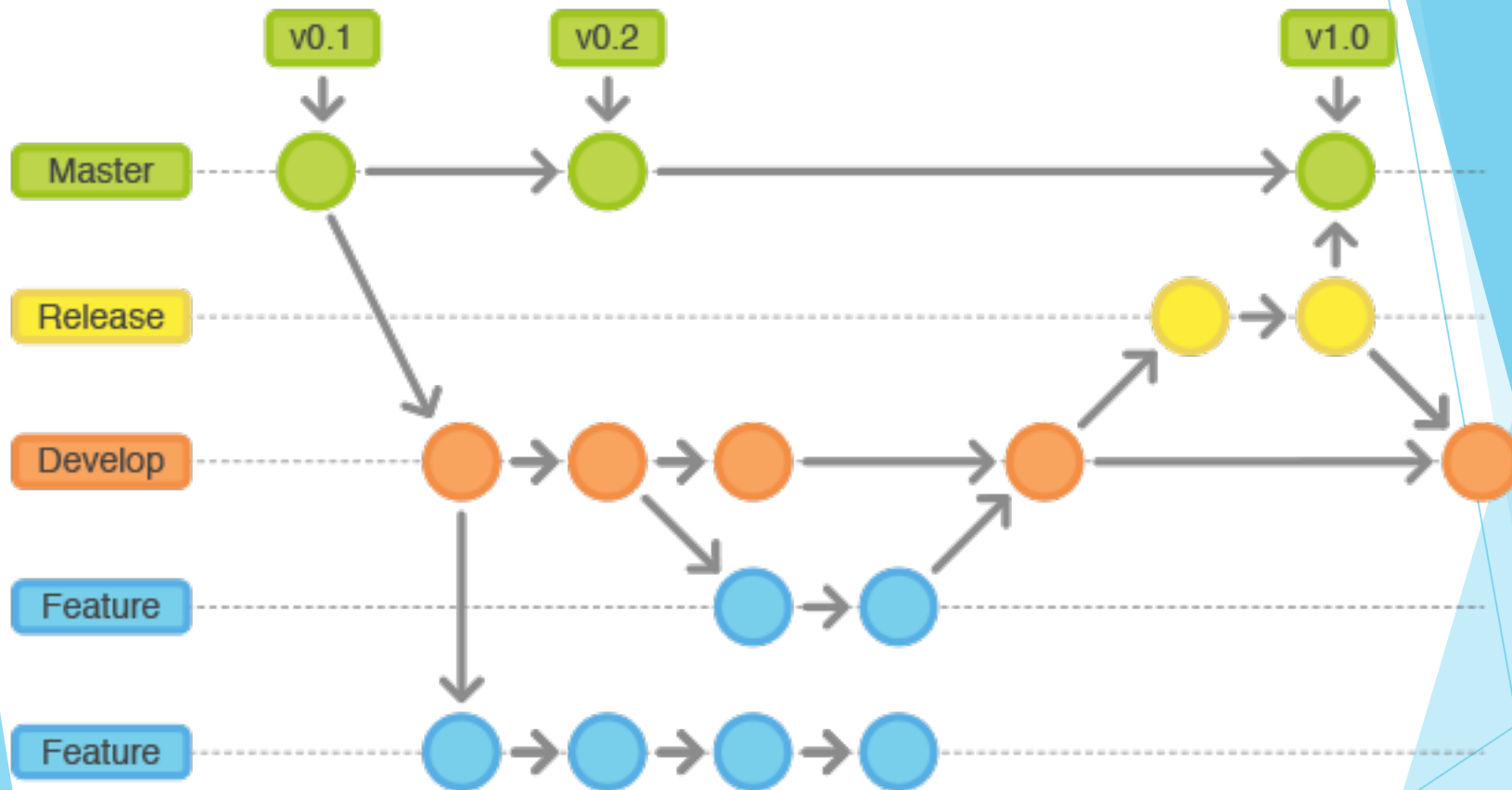
GIT, Android Studio

Jan Masner

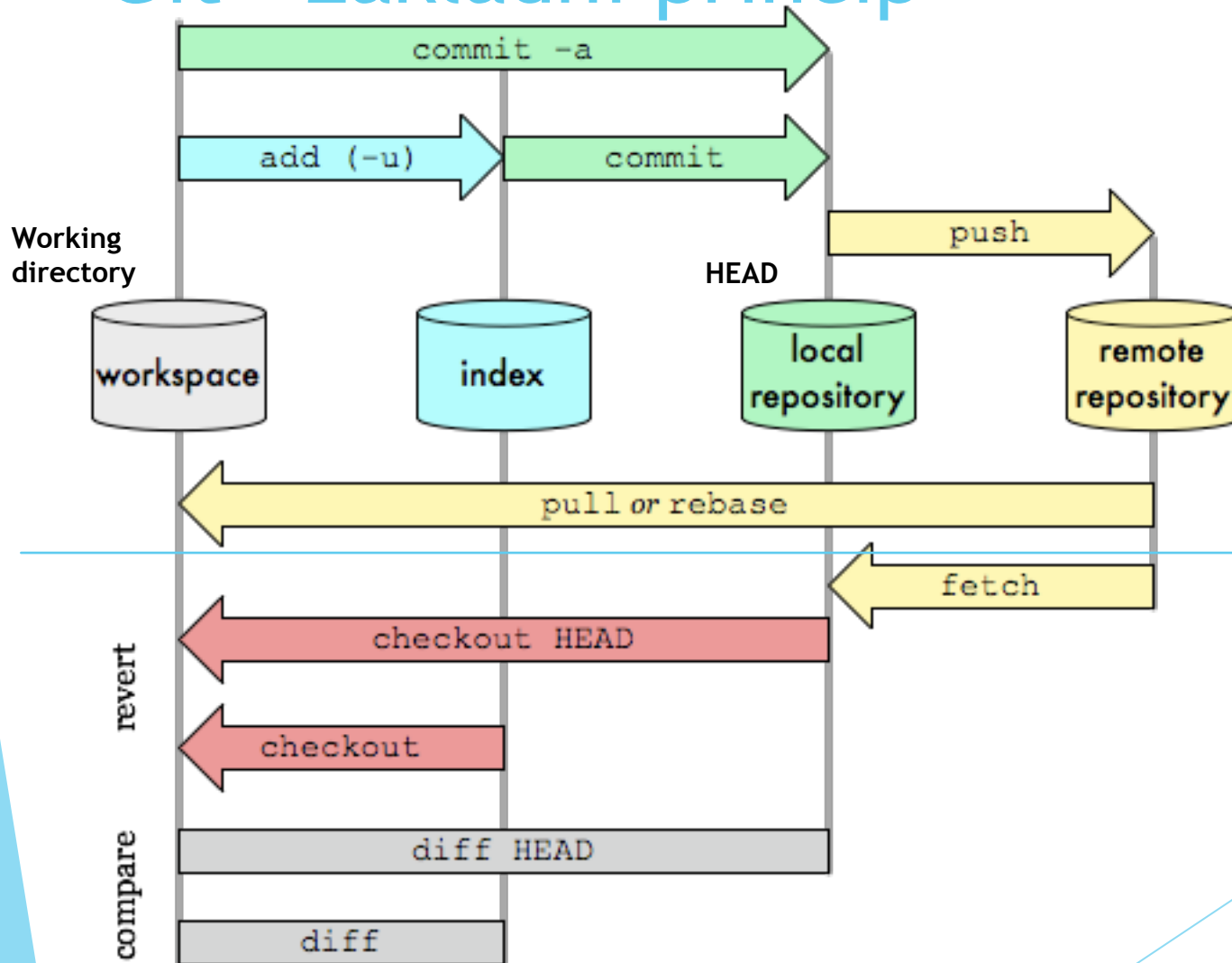
Git - verzovací systém

- ▶ Systém pro správu verzí
- ▶ Zaznamenává změny kódu na jednotlivých řádcích
- ▶ Změny binárních souborů (obrázky, videa, všechno co nejsou textové soubory)
- ▶ Možnost vytváření větví
- ▶ **Distribuo**vaný - lokální a vzdálený repozitář/e
 - ▶ Vzdálené nejsou nutné
- ▶ Původně pro vývoj jádra Linuxu
- ▶ Open Source
 - ▶ GPL v2

Větve



Git - základní princip



Nejdůležitější příkazy

- ▶ > `git init`
- ▶ > `git add -A`
- ▶ > `git commit -a -m „popis commitu“`
- ▶ > `git push`
- ▶ > `git pull`

Založení lokálního repozitáře

1. Udělat účet na Githubu
2. Vytvořit složku
3. Spustit **git bash**
 - ▶ Přesunout se do dané složky
4. > **git init**
 - ▶ Vytvoří lokální repozitář
 - ▶ Je možné jej pak nahrát na vzdálený
5. > **git status**

Přidání souborů do lokálního repozitáře

▶ `git add`

- ▶ Přidá soubory do *Indexu*
- ▶ Git o souborech/změnách ví, jsou jako nové
- ▶ IDE dělá automaticky, nebo se zeptá
- ▶ > `git add -A`

▶ `git commit`

- ▶ Aktualizuje změny do repozitáře
- ▶ Přidá nové soubory do repozitáře
- ▶ NOVÁ VERZE
- ▶ K jednotlivým verzím (commitům) se dá vracet
- ▶ > `git commit -a -m "komentář"`

Přidání souborů do lokálního repozitáře

1. Vytvořit soubor
2. `> git status`
3. `> git add prvni.xml`
4. `> git status`
5. `> git commit`
6. Vytvořit soubor *druhy.xml*, *treti.xml*
7. `> git status`
8. `> git add "*.xml" (add -all/add -A)`
9. `> git commit -a -m "Přidání 2 souborů"`

Může se hodit

- ▶ Udělejte několik změn a commitněte
- ▶ > git příkaz **-help**
 - ▶ Zobrazí nápovědu k danému příkazu
- ▶ > git **log**
 - ▶ Zobrazí historii commitů
 - ▶ git log --summary

Vzdálené repozitáře push

- ▶ Vytvořte si repozitář na githubu
- ▶ `> git remote add origin https://github.com/...`
 - ▶ `origin` je název vzdáleného repozitáře
- ▶ `git push -u origin master`
 - ▶ `master` je název výchozí větve
 - ▶ `-u` → zapamatuje si parametry (`origin master`), příště stačí jen `git push`

Stažení ze serveru

clone, pull

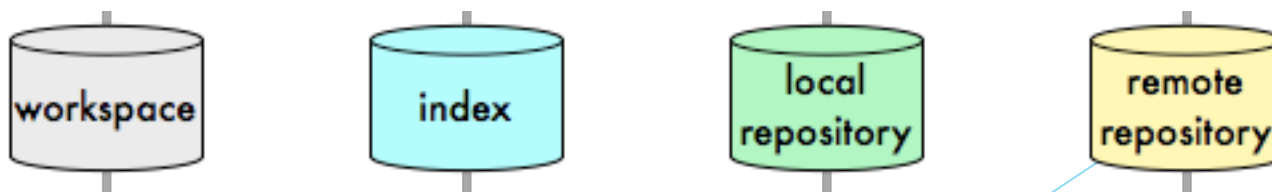
- ▶ Udělejte si dvojice
- ▶ Naklonujte si projekt kolegy
- ▶ `git clone https://github.com/...`
- ▶ Udělejte změnu
- ▶ `commit, push`
- ▶ `> git pull`
 - ▶ Stažení změn
 - ▶ Měl by předcházet `commit`

Merge, Konflikty

1. Udělejte ve dvojici změnu na jiném řádku
 2. Udělejte ve dvojici změnu na stejném řádku
- ▶ Jeden dá změnu do repozitáře na githubu
 - ▶ Druhý - commit + pull
 - ▶ Zkuste se podívat potom do souboru
 - ▶ Konflikty v tomto případě nutné vyřešit ručně

Zobrazení změn - Diff

- ▶ `git diff`
 - ▶ Workspace <-> index
- ▶ `git diff HEAD`
 - ▶ Workspace <-> poslední commit
- ▶ `git diff master origin/master`
 - ▶ Poslední commit <-> vzdálený repozitář
- ▶ `git diff origin/master`
 - ▶ Workspace <-> vzdálený repozitář



Práce s větvemi

- ▶ Vytvořte novou větev s názvem „vetev“
- ▶ > git **branch** vetev; > git **checkout** vetev
 - ▶ git **checkout -b** vetev
 - ▶ Vytvoří novou větev a přepne se na ní
- ▶ Udělejte změny v nové větvi
- ▶ > git checkout master
 - ▶ Přepne zpátky do masteru
- ▶ > git merge vetev
 - ▶ Připojíme změny z větve
 - ▶ vetev pořád existuje

Další příkazy

- ▶ Rebase
 - ▶ Obdoba merge, dělá jinak
- ▶ Reset
 - ▶ Rušení změn, návrat ke commitům
- ▶ `git reset --hard <tag/branch/commit id>`
 - ▶ Návrat ke konkrétnímu commitu
- ▶ Fetch
 - ▶ Stáhne obraz vzdáleného repozitáře