



# Git Lesson Plans For Unit Level Training

# The Problems (Broadly)

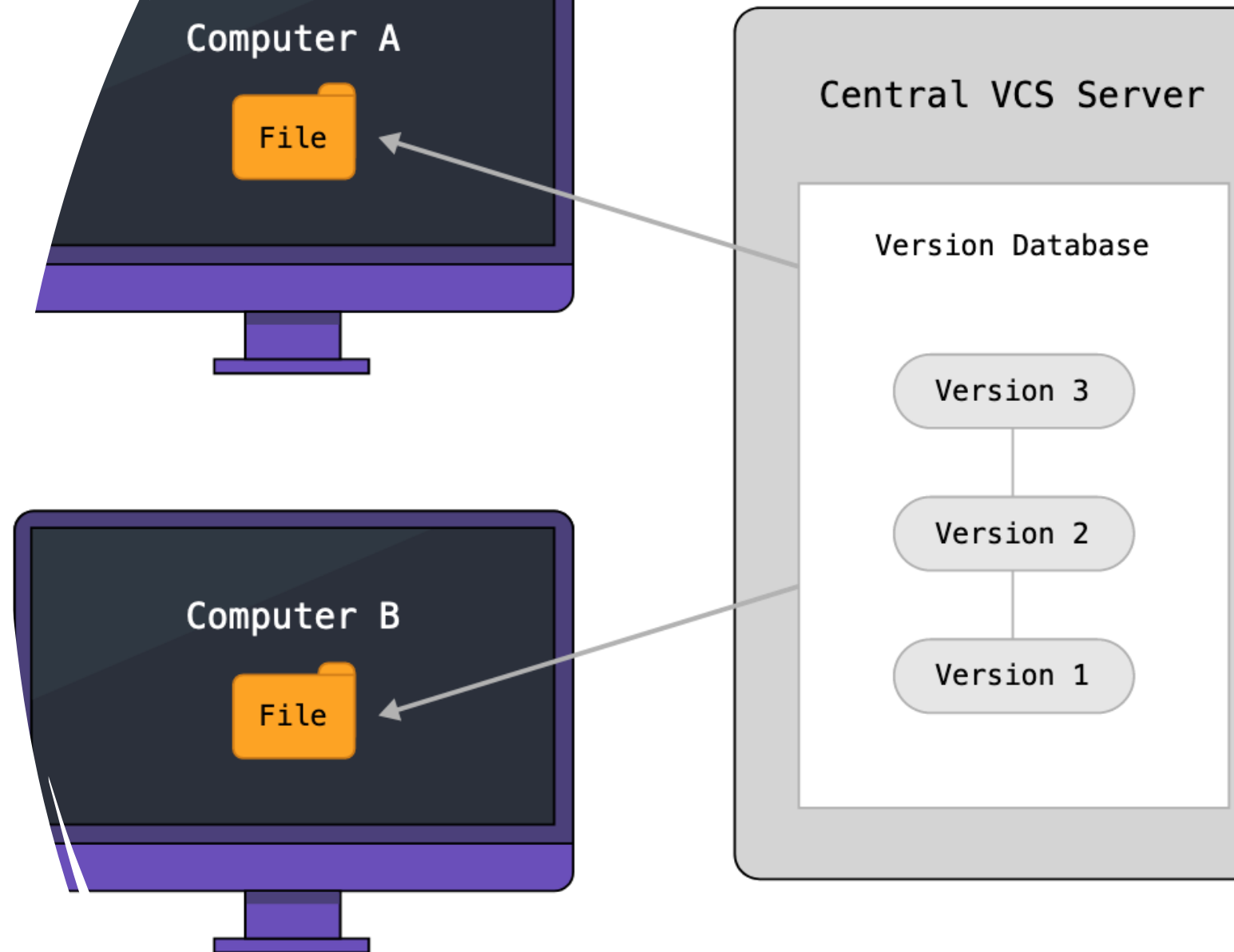
1. Inadequate Knowledge Management – for Detachment and Individual Training
2. Slides instead of Detailed Lesson Plans
3. No Feedback between Developer, Instructors, and Students
4. Inadequate Version Control
5. No Synchronized Training Content Updates

See [Last slide](#) for more

# Knowledge Management Now

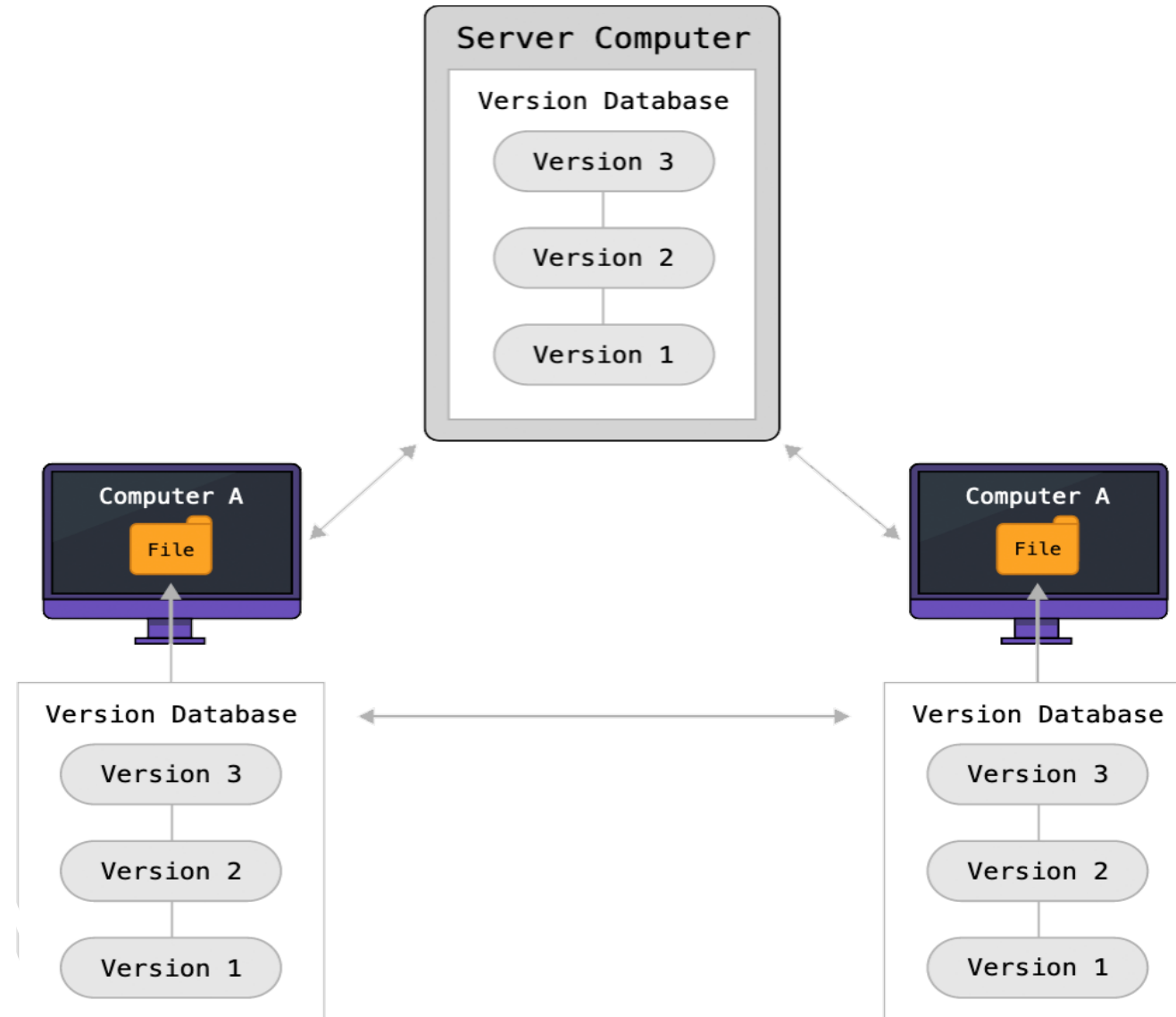
---

- TRADOC centralized system for approval and distribution of training (very exclusive) proponent level
- Limited access to developed training resources by units
- Limited interactions between
  - Instructors
  - Students
  - Operational Force



# Better Knowledge Management

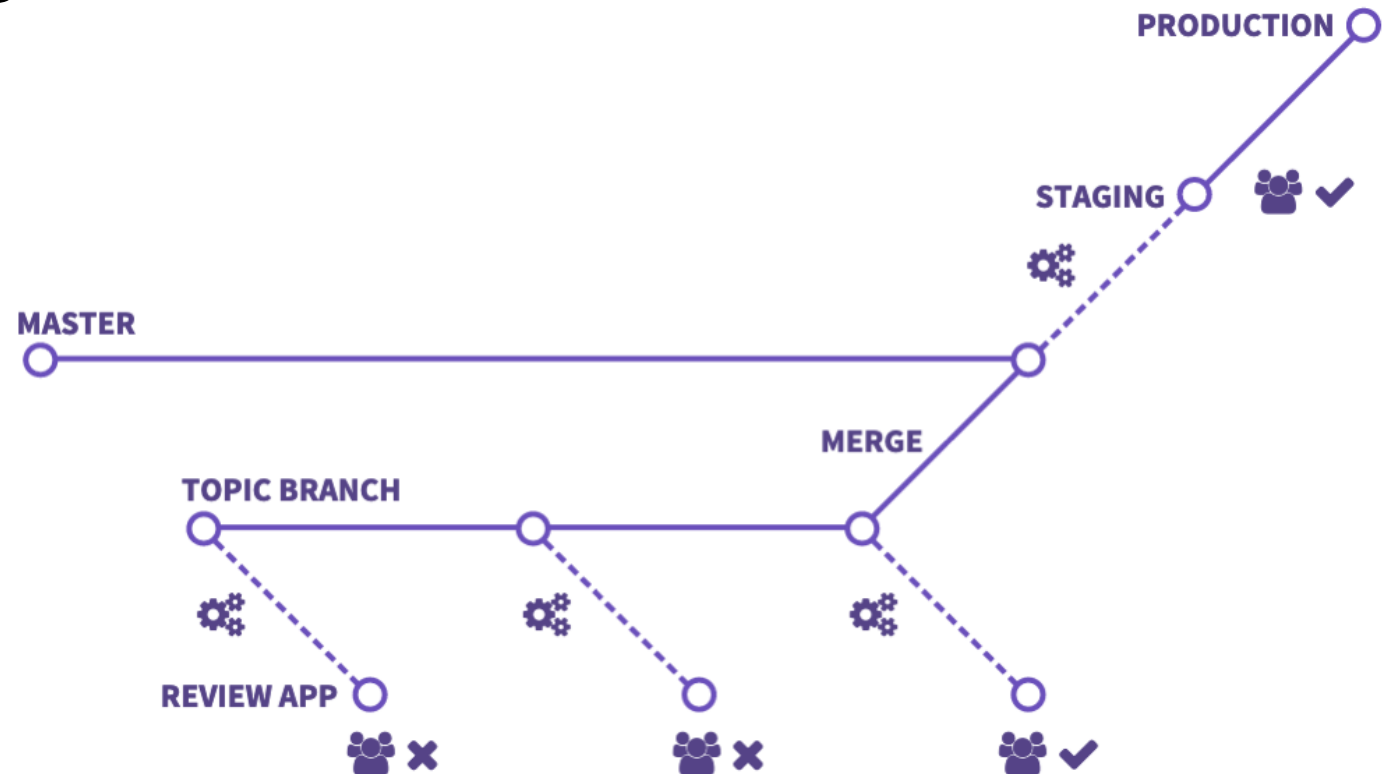
- Office365 real time sync of training slides and instructor content
- Good
  - Rapid Updates
  - Some Permission Controls
- Bad
  - Version Control
  - Expanding on Training
  - Team Management
  - Ground Level Feedback
  - Ground Level Updates
  - Word Formatted – Hard to Version Control



# Git – Even Better

*Git is an open source, distributed version control system used for DevSecOps and Agile project management.*

- Low-cost solution for providing a library of training
  - PlatformOne
  - Locally Hosted
- Provides direct feedback from unit trainers and students for fast fixes and updates.
- Provides SCRUM and Agile functions



# The Lesson Content Problems Now

- Slides instead of Lesson Plans
  - Inconsistency in training environment setup (hard to communicate on slide only content)
  - Unable to transfer actual instruction of content without extensive communication
  - Lack of transparency or processes to push updates and take feedback
- False Starts “recreating the wheel”



# The Possible: DET and Individual Lesson Plans

- Training Library using Git and Markdown Lesson Plans
  - Unit Training Knowledge Management
  - Instant Feedback (Issues)
  - Ground Truth Updates (Pull Requests)
- Lesson Plans
  - Lesson modularity ideal for Unit Training limitations
  - Intent, Requirements, Configuration, and References available by default
  - Expand who can teach

Find

Prev

Next

All

×

## Equipment and Material Required for Instruction

Lesson Material, Ammo, Expendable, etc.

Equipment	Student Ratio	Instructor Ratio	Quantity	Expendable
#MacOS-Computer , #MacOS-Keyboard , #MacOS-Touchpad	1:1	1:2		no
#Slides			1	no
#Projector			1	no
#ShortcutStickerMacOS	1:1	1:1		yes
#CheatSheet_app	1:1	1:2		
#Terminal_app	1:1	1:2		
#printer_paper	1:1			yes
#sticky_notes	4:1			yes

## Pre-Requisite Tasks

None

## Knowledge Required

1. K1 - Understanding of common differences between WindowsOS and MacOS
3. K3 - Be aware of QWERTY keyboard
4. K4 - Understand general computer home directory layout
5. K5 - Knowledge of opening apps on device
6. K6 - Knowledge of logging in and out of a computer

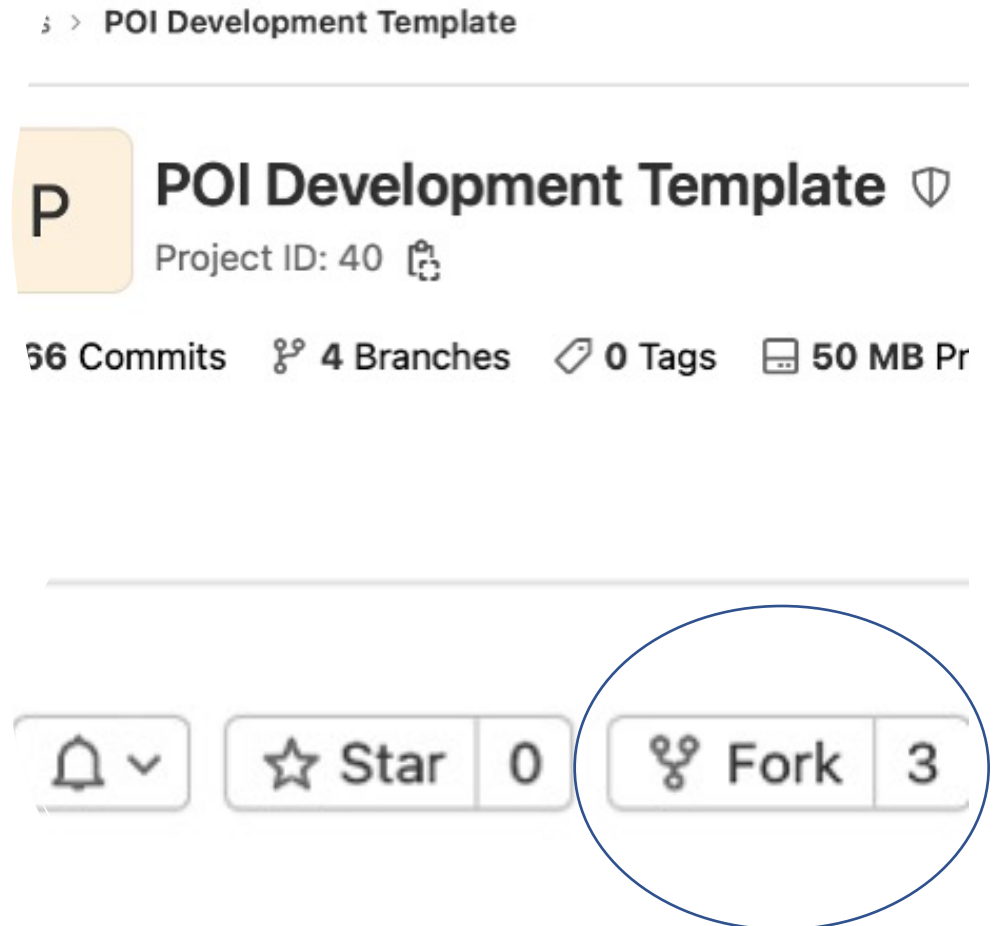
## Skills Required

1. S1 - Typing ability approximately 20wpm or more

# Use Prototype

---

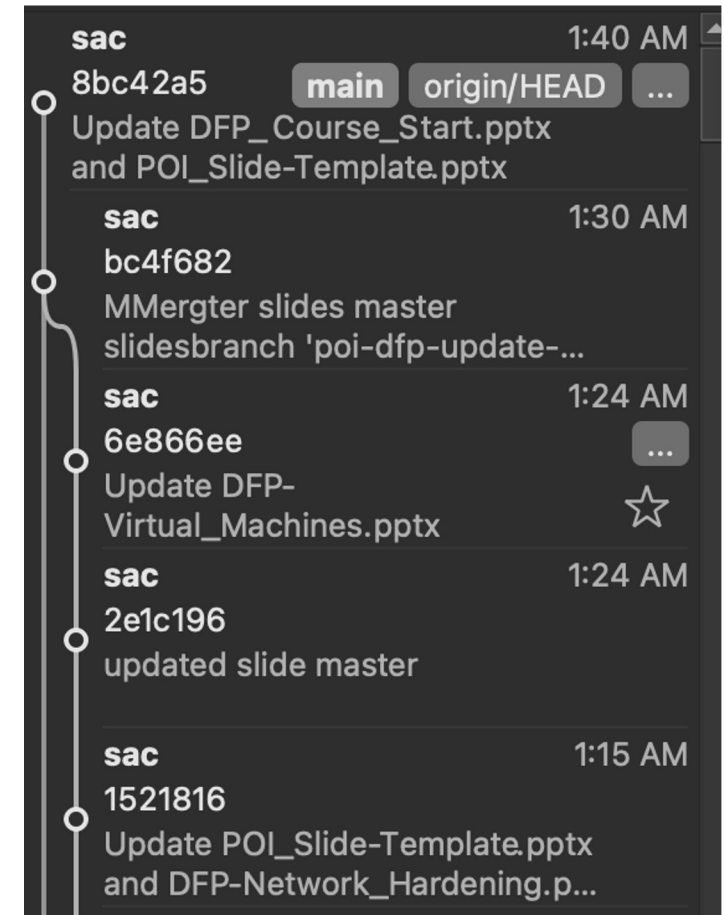
1. [Create Account:](https://gitlab.irregularchat.com)  
<https://gitlab.irregularchat.com>
2. [POI Development Template:](https://gitlab.irregularchat.com/saclab/poi-development)  
<https://gitlab.irregularchat.com/saclab/poi-development>
3. Create a Branch for a new Lesson





# Beyond Prototype

- Move to DODIN
  - PlatformOne (NIPR)
  - SIPR
- Expand Content
- Expand Access
- Develop Non-Formal Training for Unit Instructors
- Develop Command Level AMETL



# References



- Why Git: <https://page.gitlab.com/rs/194-VVC-221/images/gitlab-moving-to-git-whitepaper.pdf>
- Learn Git Basics: <https://yt.irregularchat.com/watch?v=RG0j5yH7evk>
- PlatformOne: <https://p1.dso.mil/resources>
- TRADOC Pamphlet (TP) 350-70

Assumptions for the gaps identified include units training on operationally relevant tasks which have not been incorporated into doctrine and proponent training, unit training lacking training support packages (TSP), training developed for partner forces, and contracted training on capabilities available in support of operations.

- Detachment and Individual level training is currently being contracted out or developed from scratch or by reverse engineering the performance steps of evaluations. The NCOs developing this training are not taking advantage of the experiential learning method or other instructional design standards because they are not aware of the methods or resources.
- Detachment and Individual level training that is developed lacks adequate knowledge management, approval, dissemination, and update mechanisms. These mechanisms exist in proponent level tools, such as the ITN, which are unavailable to unit level trainers. Without these mechanisms, units continuously recreate training without fully benefiting from lessons learned in previous training iterations within their own unit or within the entire community training on the same objectives.
- PowerPoint slides and hands on imitation are nearly exclusively used to develop unit training. These slides do not adequately relay the intention of the training, the requirements for hardware, software, handouts, media, and configuration, or context for unit trainers. Trainers who are subject matter experts at their craft often lack instructional design and instructor training leading to less effective transfer of knowledge and lower operational effectiveness and application.

- Students of Unit training often provide feedback to the trainer directly but do not have established processes to provide feedback to the training developer. This feedback can range from minor fixes to changes in operational ground truth. Developers struggle to obtain this ground truth via emails, surveys, AARs, and interviews.
- Changes are made without clear identification of what has been changed and why as a change to a single sentence represents a full change to the entire POI file with the current systems. Git automatically provides specific labels on exactly who changed what, when, how, and why. This changelog along with detailed lesson plans helps trainers to accurately train Units to the standards and intentions of the course developer.
- Contracted training is often viewed as “Trained for Life” and does not provide commander’s with and accurate understanding of Soldier’s proficiency. Using a Git unit training library will allow further integration with contracted training and allow units to build off contracted training to develop more effective TTPs better understanding of individual competency.
- Unit training content currently ranges from accessible on MilSuite or iSpace to being stuck on someone's computer or email. Allowing units to build off each other’s training content and to have a library of training will rapidly advance the quality and quantity of available training and enable units to train to standard more often and with a better understanding of what is required.
-