GitHub Basics: Building Your Coding Portfolio

Ahmed Shahan

Treasurer
IEEE NSU RAS SBC

LIB 607

24th September, 2023

IEEE NSU IAS STUDENT BRANCH CHAPTER
IEEE NSU STUDENT BRANCH
IEEE NSU STUDENT BRANCH CHAPTER

6:30pm - 8:00pm

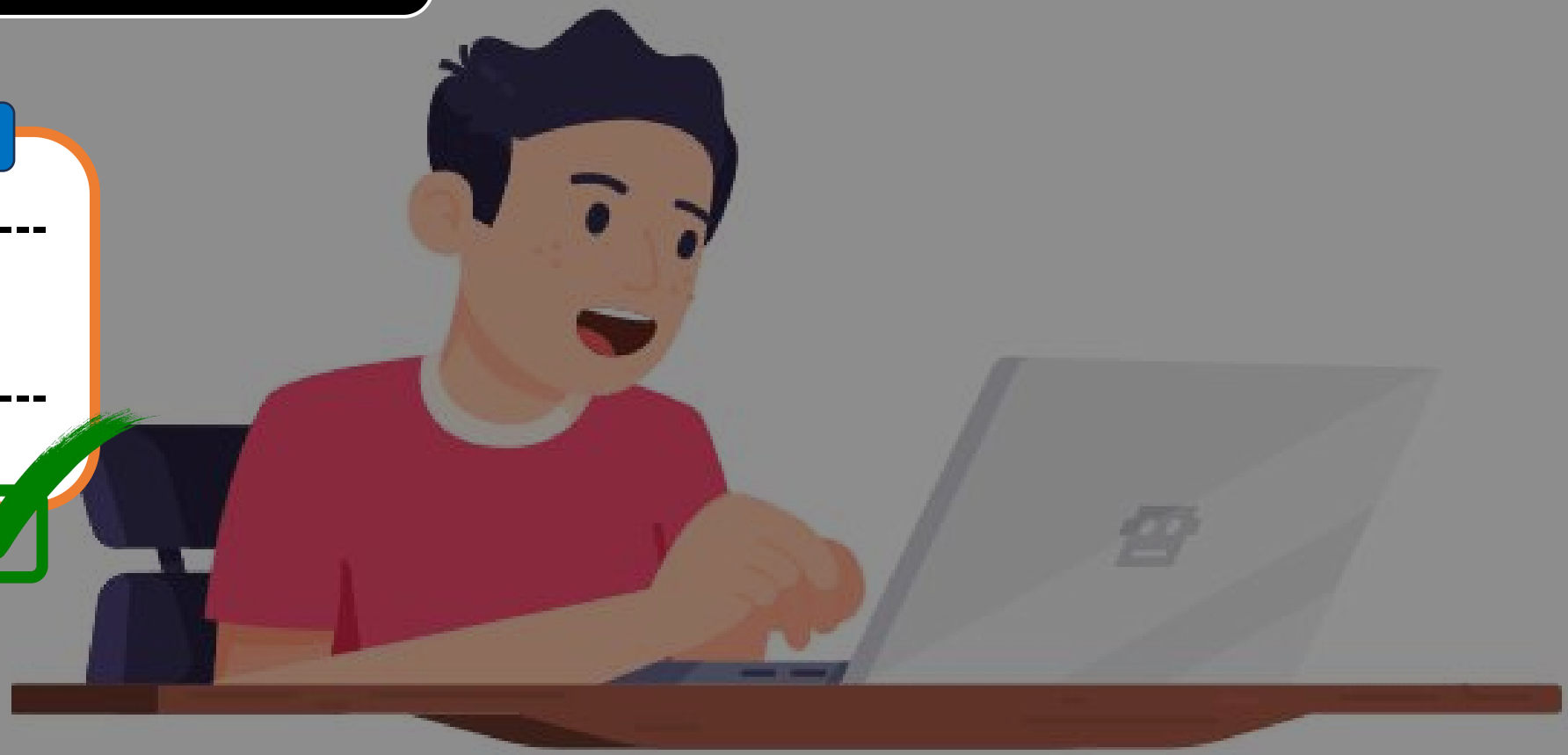
Stay Motivated & Consistent While Coding

Scenario I

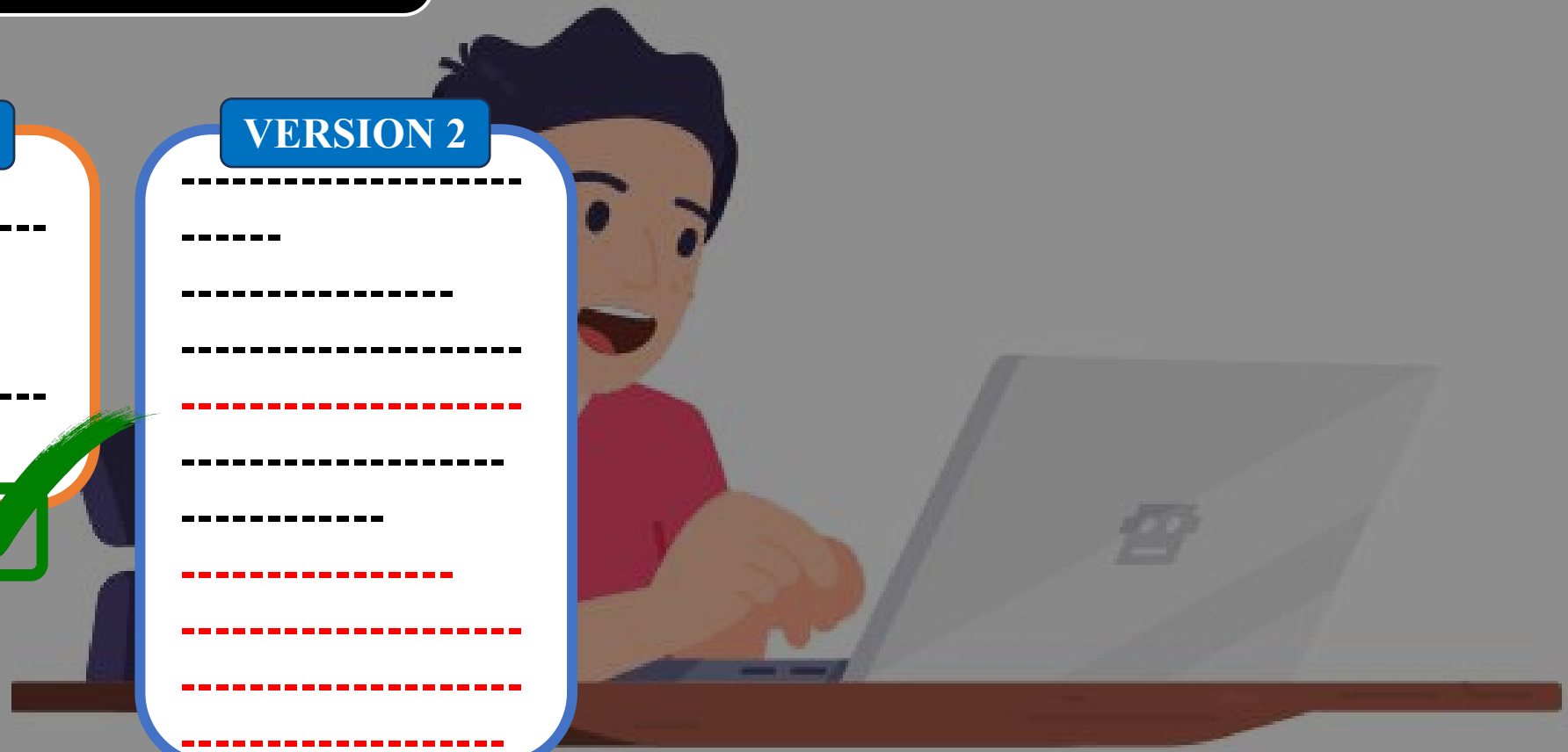
VERSION 1

Scenario I

VERSION 1



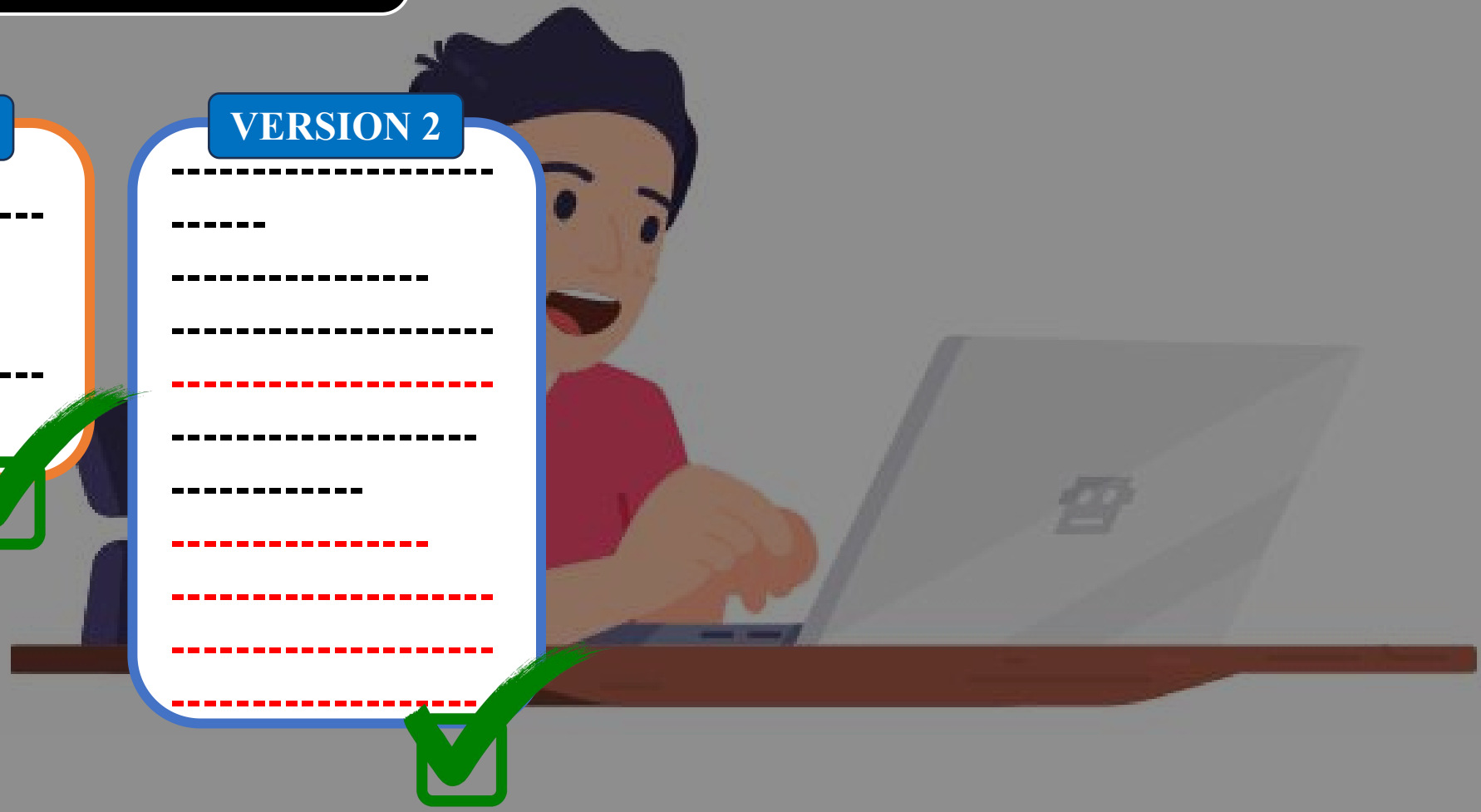
Scenario I



VERSION 1

VERSION 2

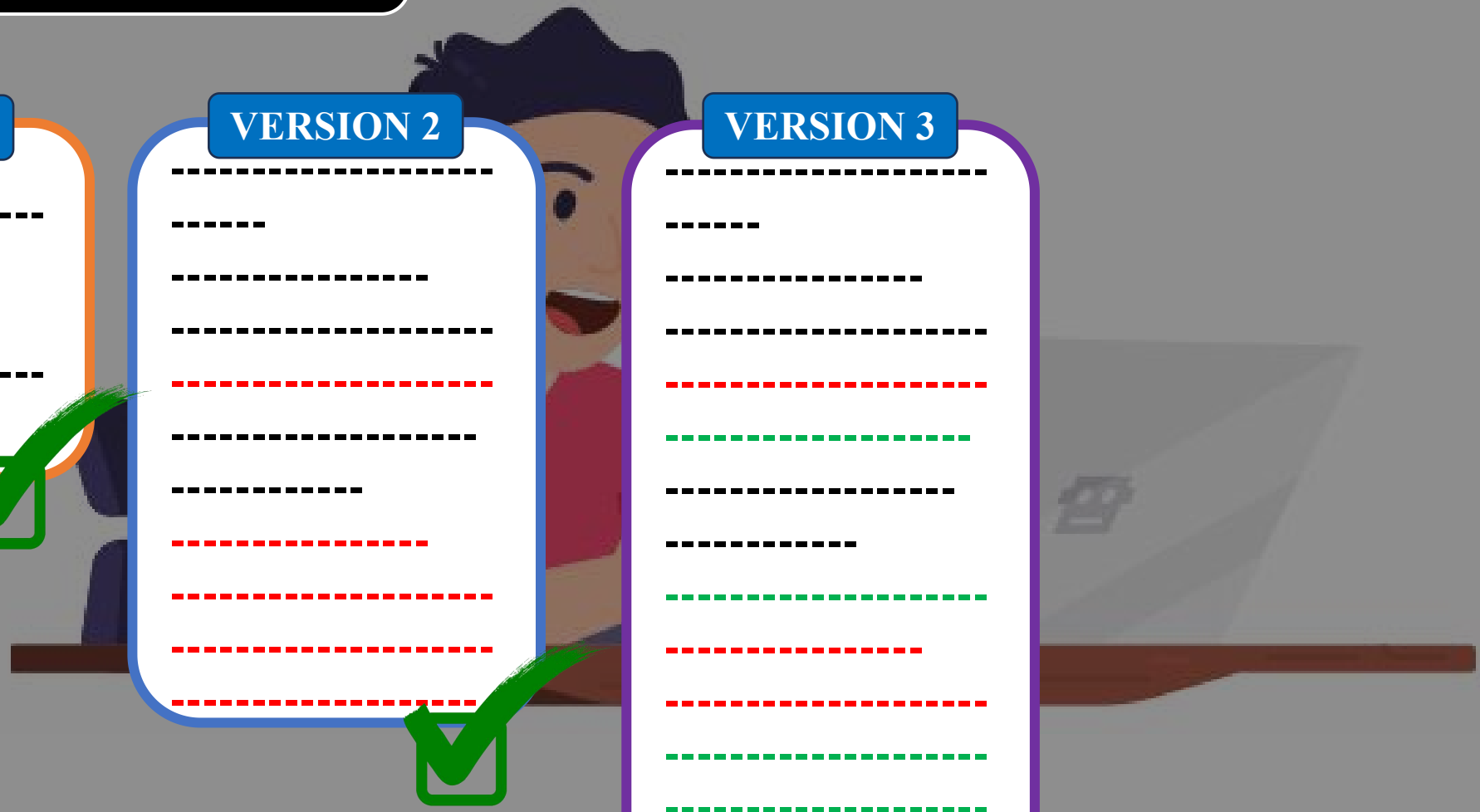
Scenario I



VERSION 1

VERSION 2

Scenario I

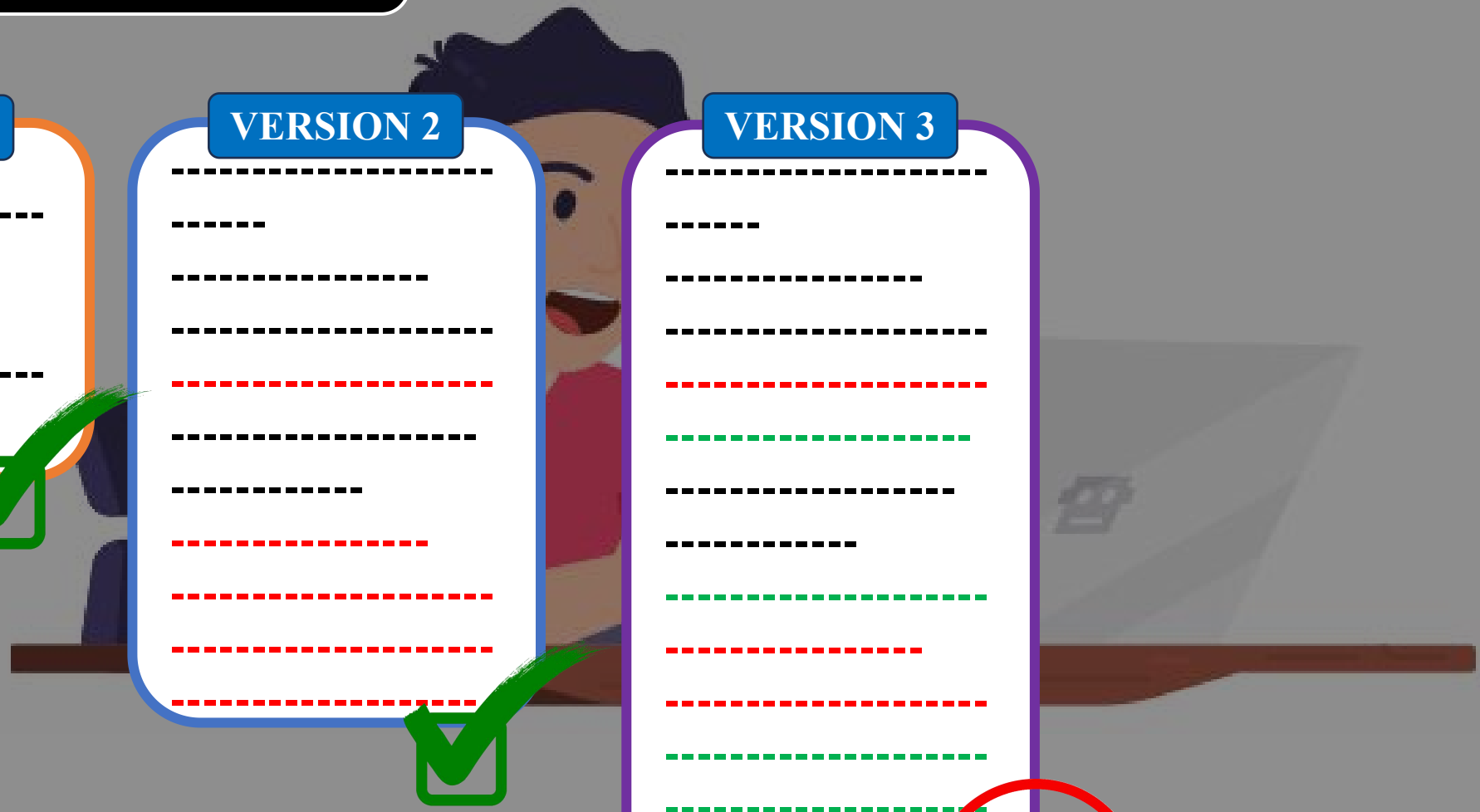


VERSION 1

VERSION 2

VERSION 3

Scenario I



VERSION 1

VERSION 2

VERSION 3

Scenario I

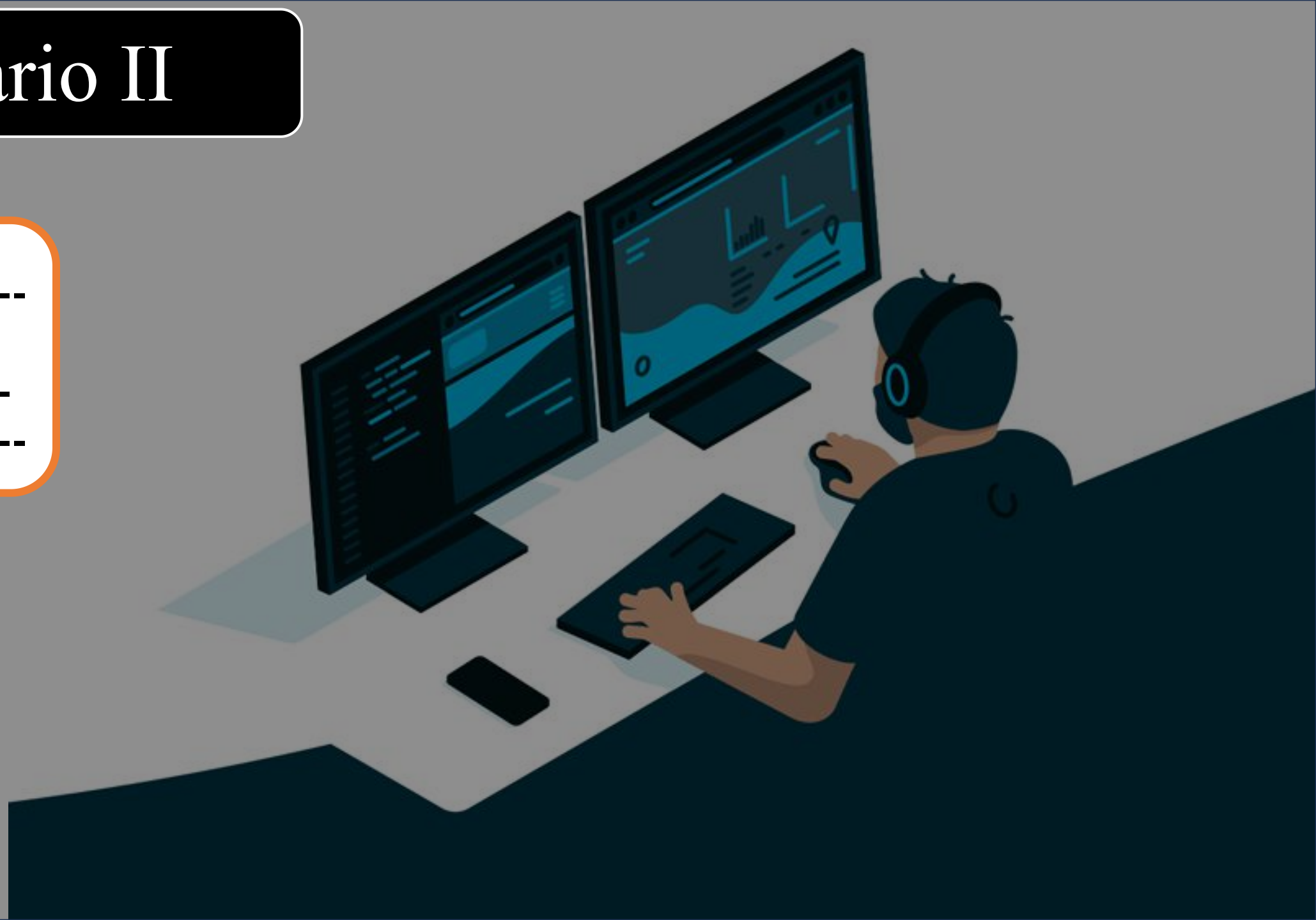
VERSION 1

VERSION 2

VERSION 3

Scenario II

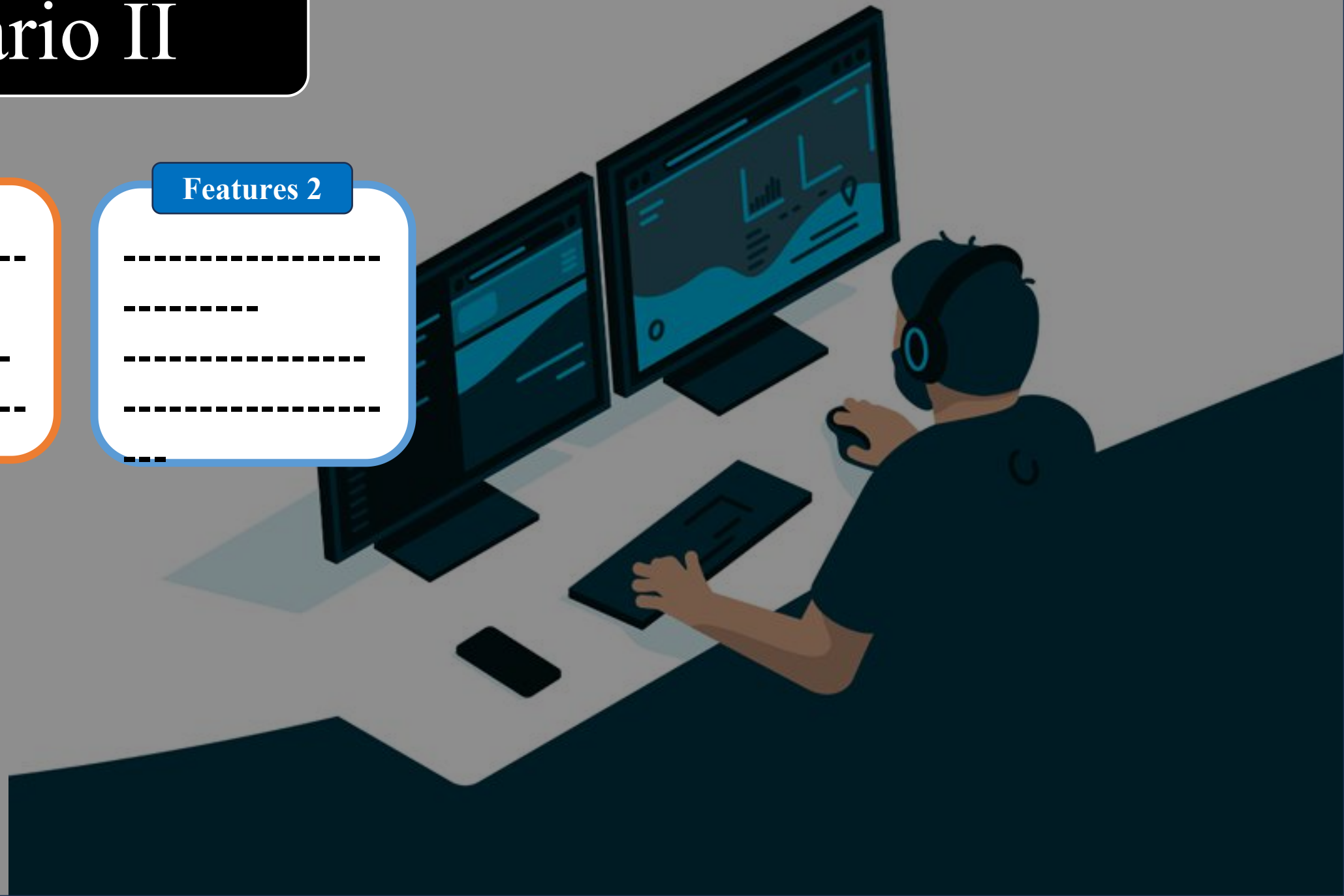
Features 1



Scenario II

Features 1

Features 2

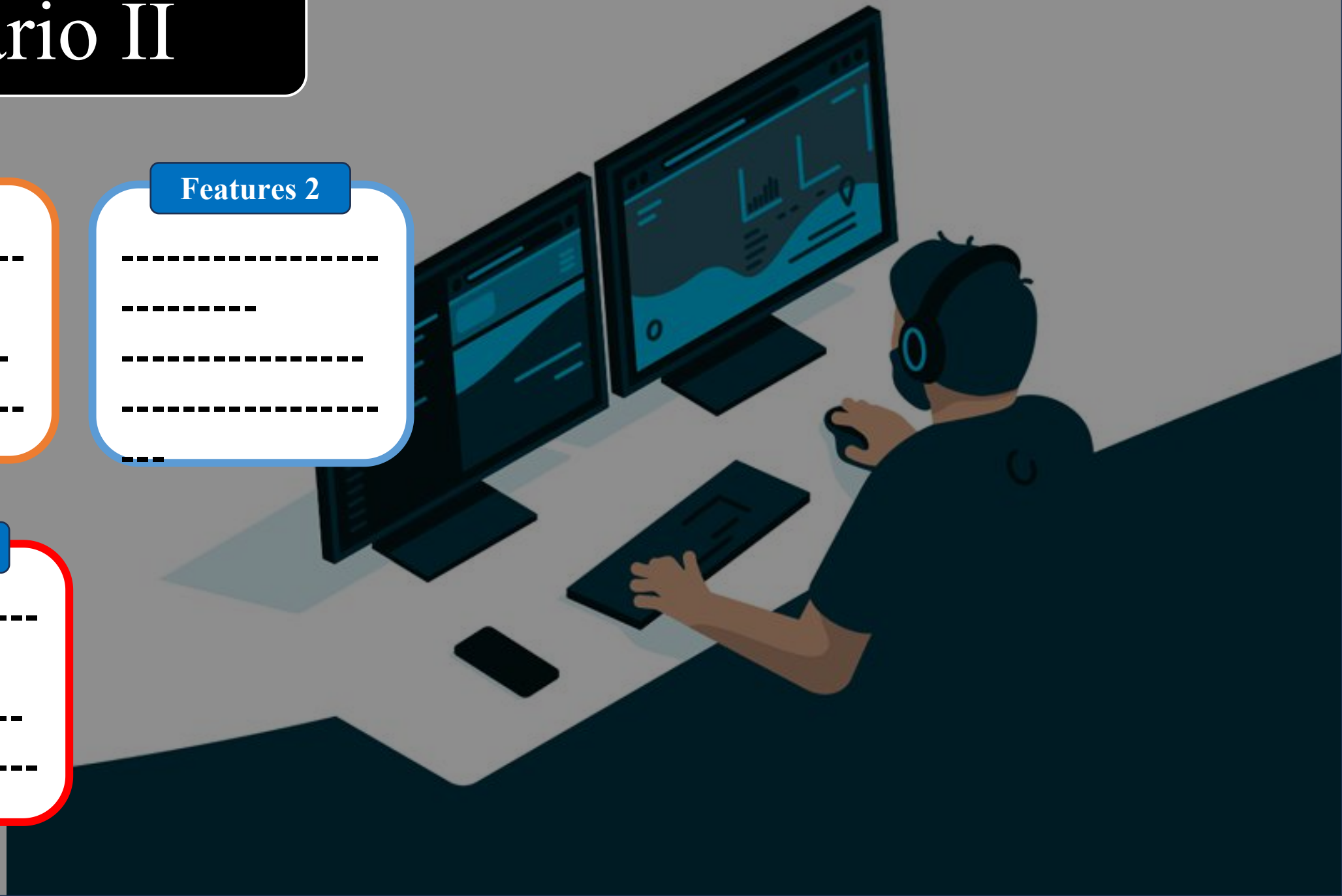


Scenario II

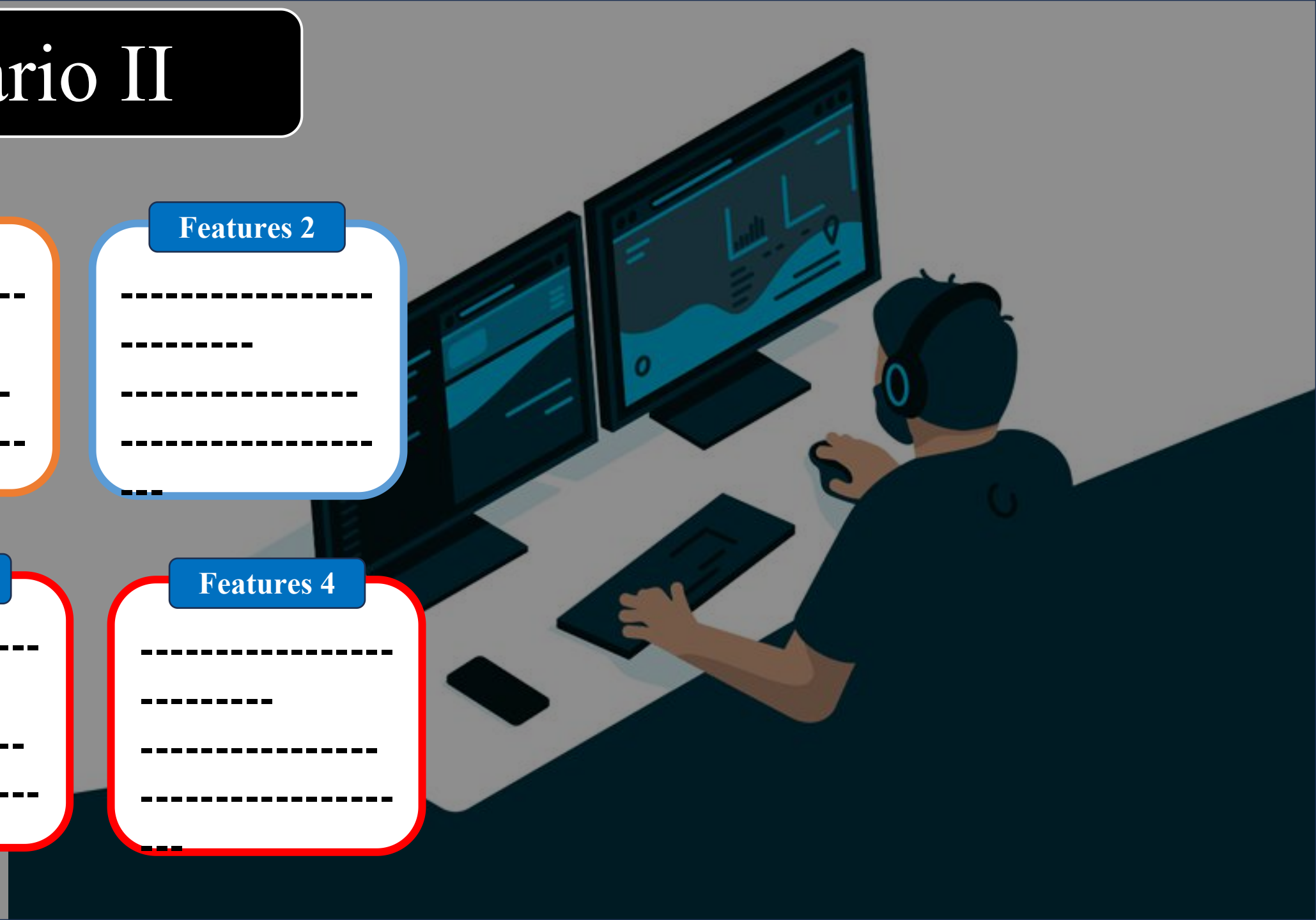
Features 1

Features 2

Features 3



Scenario II



Features 1

Features 2

Features 3

Features 4

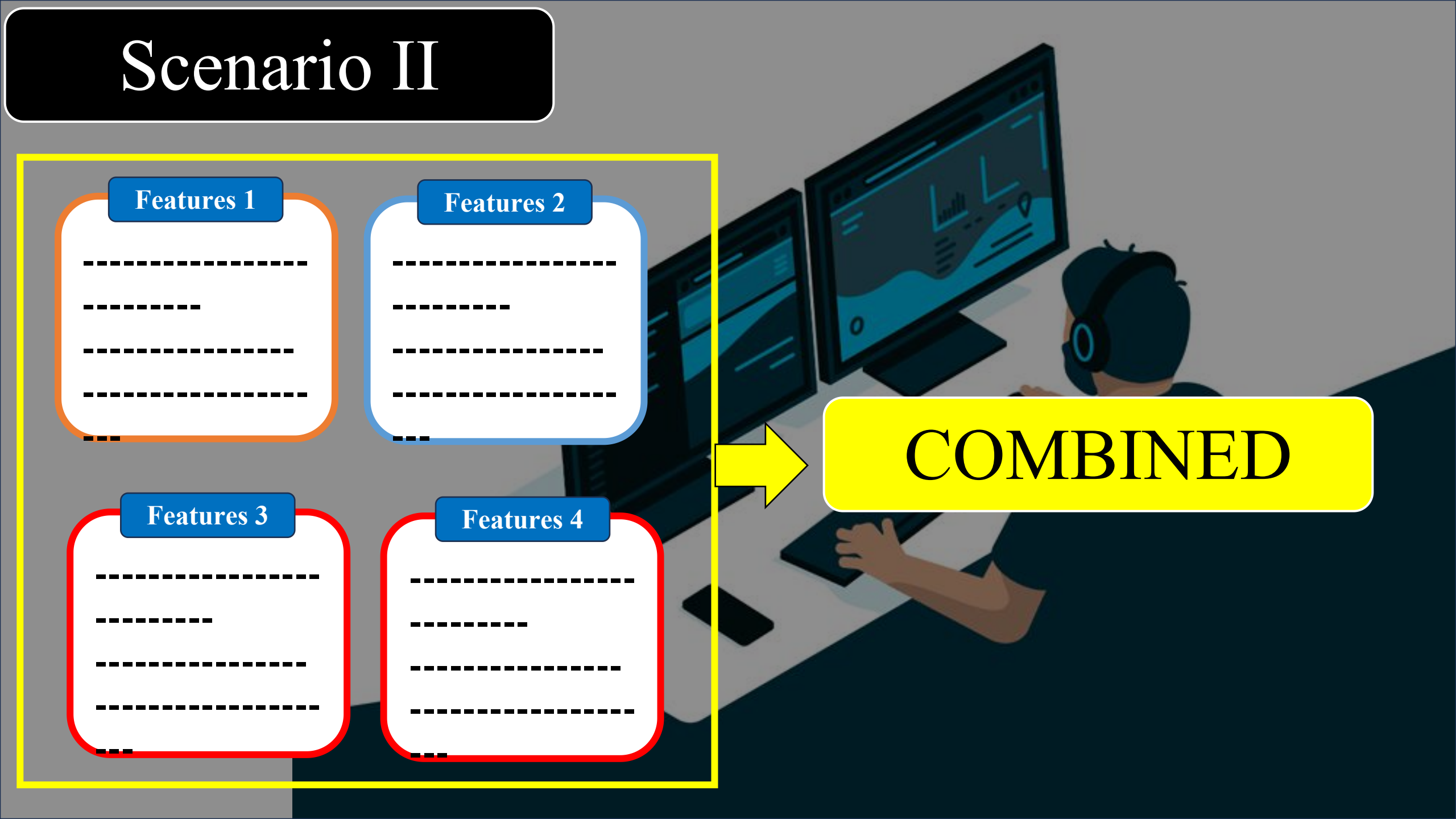
Scenario II

Features 1

Features 2

Features 3

Features 4



COMBINED

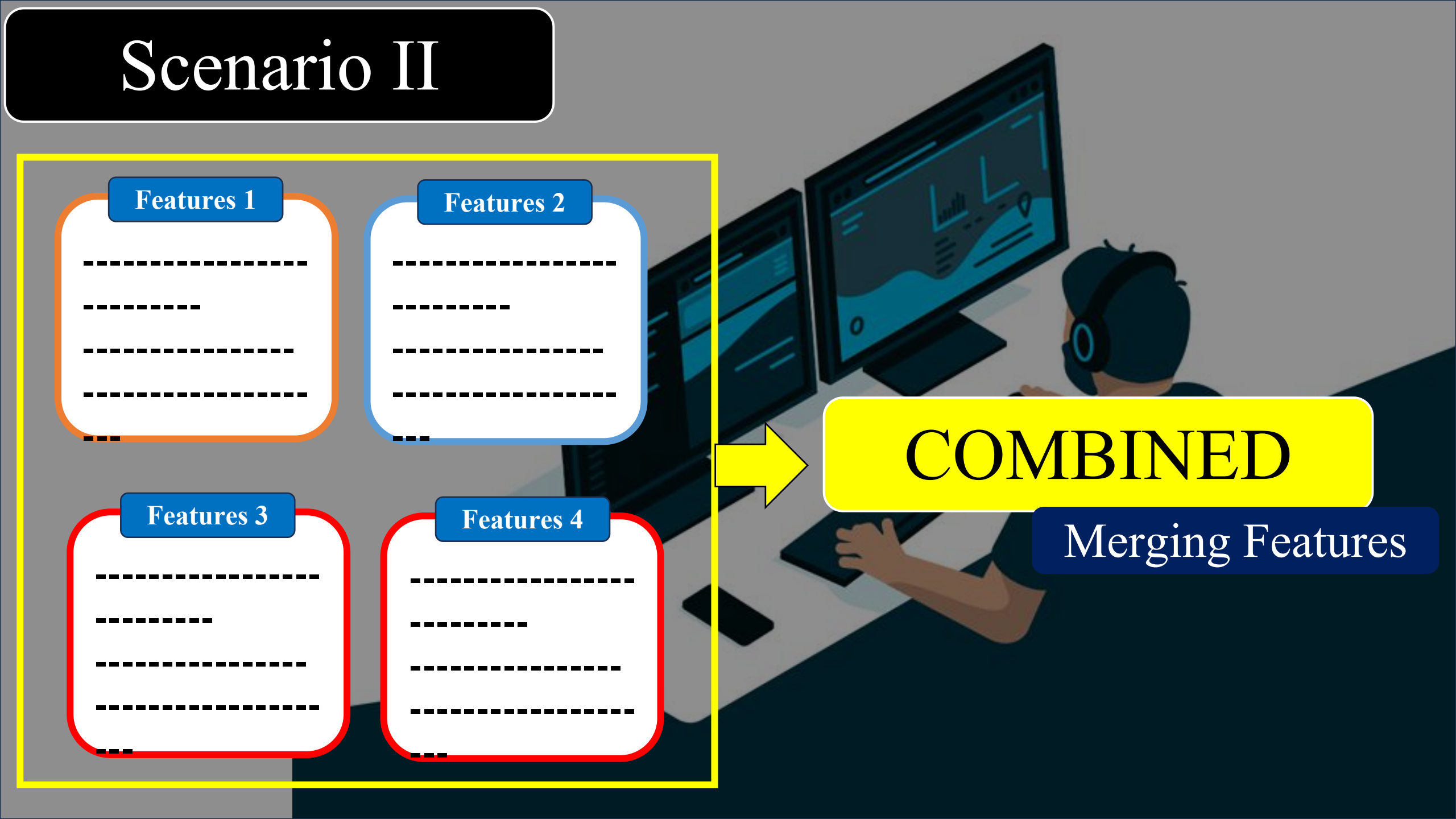
Scenario II

Features 1

Features 2

Features 3

Features 4



COMBINED

Merging Features

Scenario III

Bangladesh

UK

USA

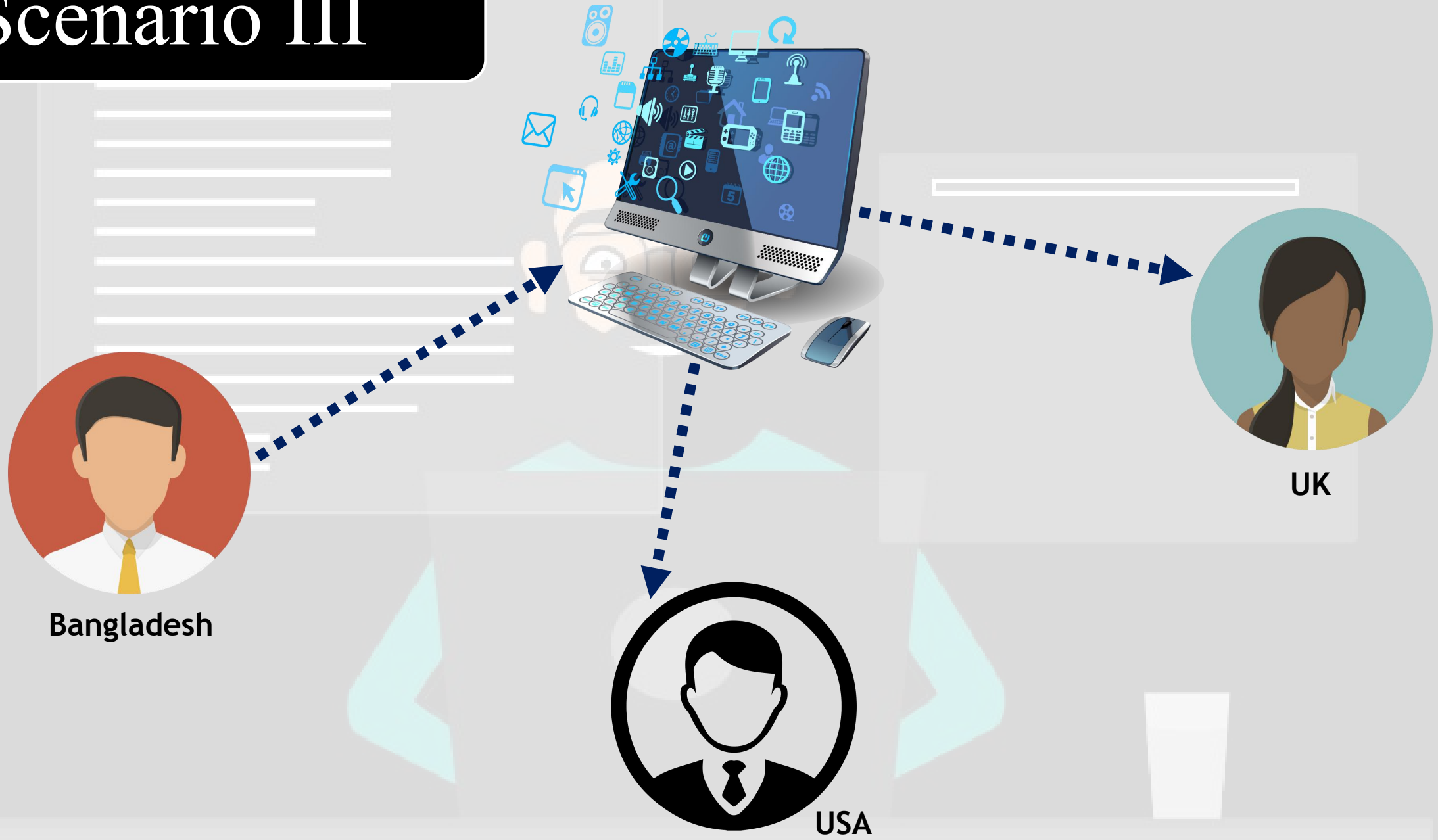
Scenario III

Bangladesh

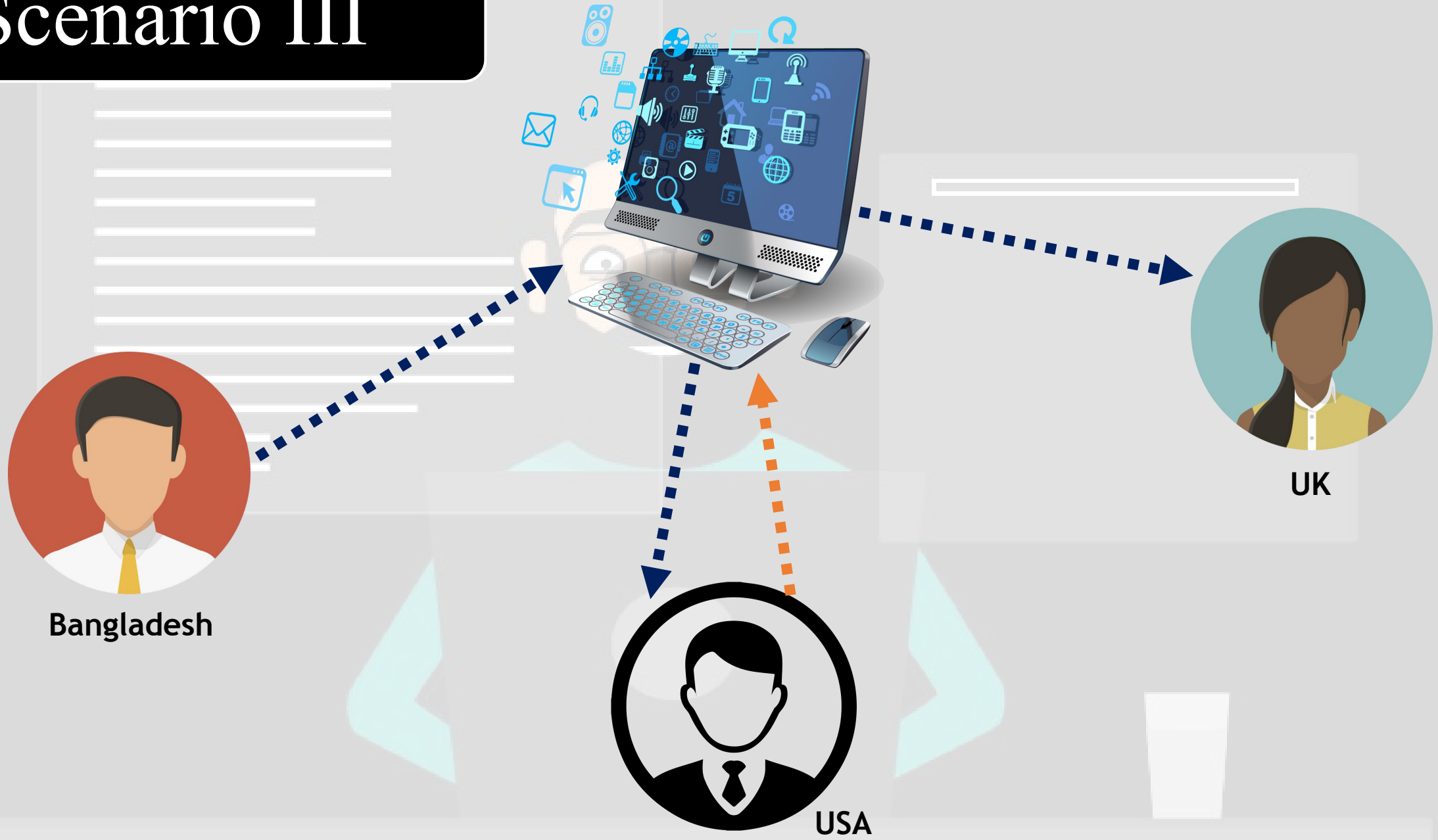
UK

USA

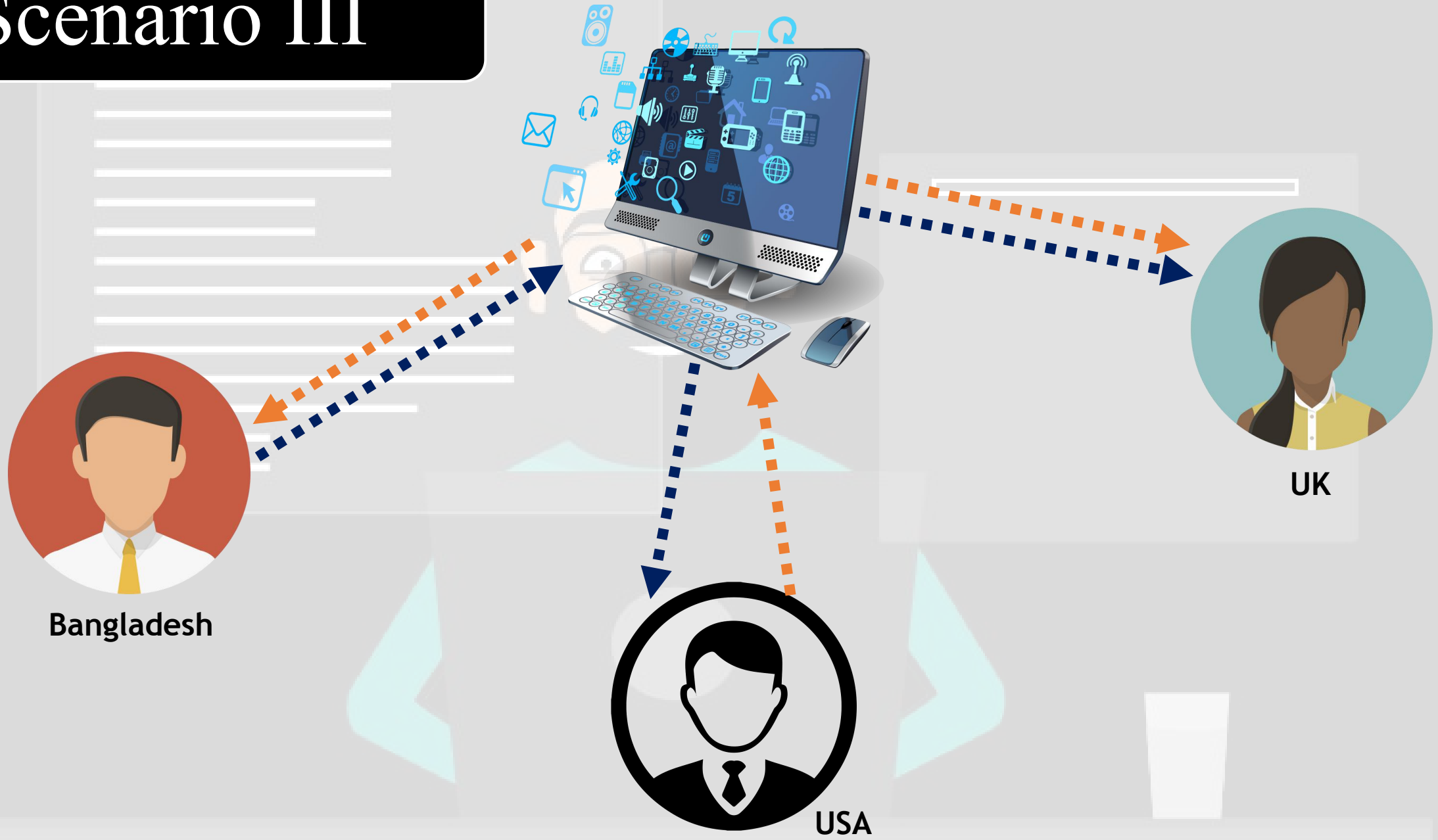
Scenario III



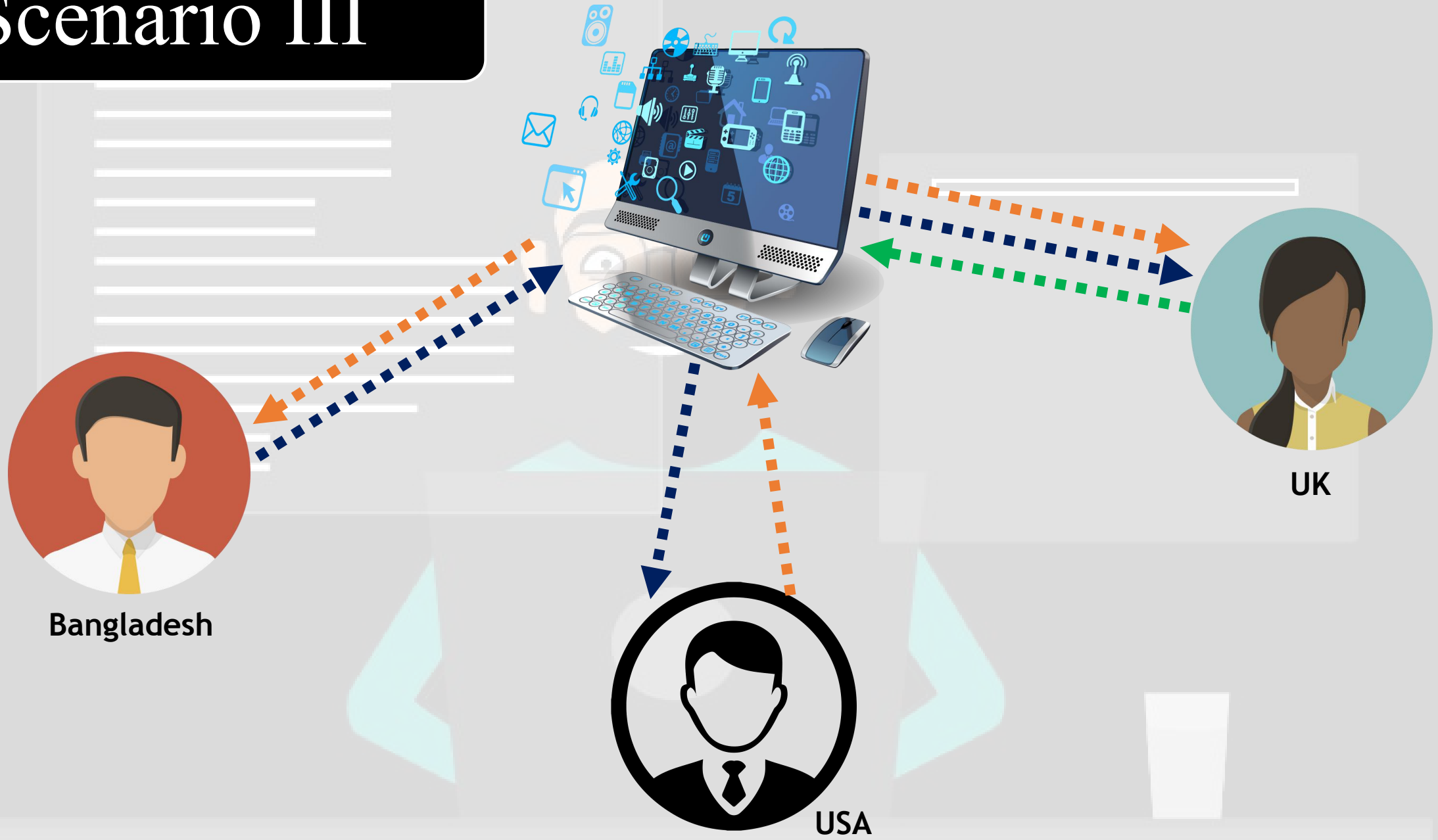
Scenario III



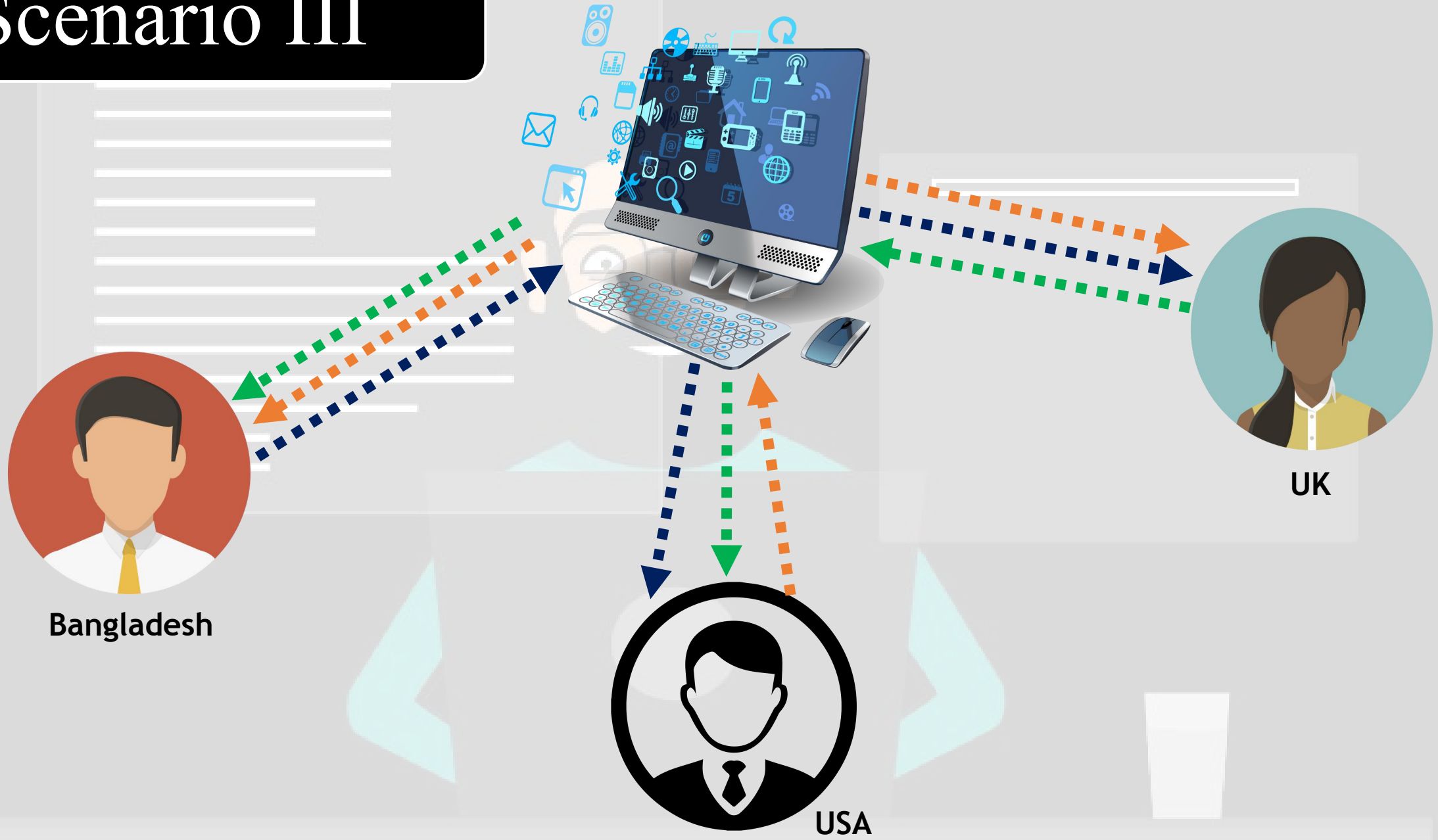
Scenario III



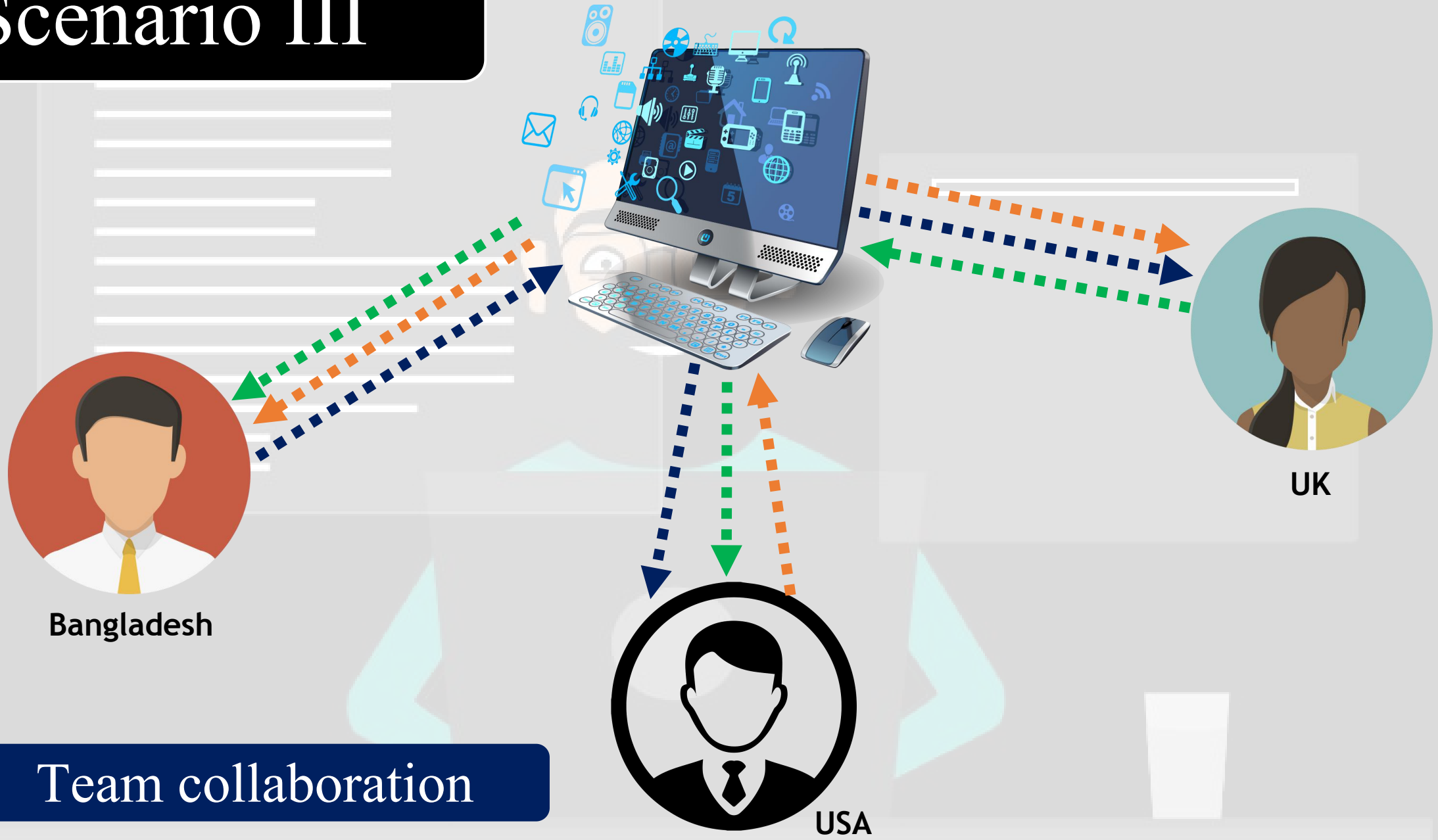
Scenario III



Scenario III



Scenario III



Team collaboration

Version Tracking

Changes Tracking

Merging Features

Team collaboration

.
.
.
.
.
.

Version Control System

Store all the records in cloud

Service

Host git repository/ folder/ source code

Provides GUI interface

Maintained on cloud / web

Tools/ software/ system

Change control/ version control.....

Provides CLI (Command Line interface Tools)

Installed and maintained locally

List of Version Control System

Azure DevOps Server

(160) **4.2** out of **5**

An enterprise-grade server for teams to share code, track work, and ship software — for any language, all in a single package. It's the perfect complement to your IDE.

Categories in common with Git:
Version Control Systems

Helix Core

(96) **4.2** out of **5**

Helix Core is an industrial-strength version control and collaboration platform that supports flexible workflows. Helix Core eliminates the complexity of large-scale, distributed product development – allowing you to build products faster while protecting your valuable IP.

Categories in common with Git:
Version Control Systems

AWS CodeCommit

(82) **4.2** out of **5**

AWS CodeCommit is a fully-managed source control service that makes it easy for companies to host secure and highly scalable private Git repositories. You can use CodeCommit to securely store anything from source code to binaries, and it works seamlessly with your existing Git tools.

Categories in common with Git:
Version Control Systems

Plastic SCM

(36) **4.2** out of **5**

Plastic SCM is a version control system that will help you develop new features in parallel, go distributed, merge in time, and never break a build again

Categories in common with Git:
Version Control Systems

Mercurial

(31) **4.2** out of **5**

Mercurial is a free, distributed source control management tool. It efficiently handles projects of any size and offers an easy and intuitive interface.

Categories in common with Git:
Version Control Systems

Micro Focus AccuRev

(20) **3.8** out of **5**

AccuRev is a software configuration management tool which addresses complex parallel and distributed development environments with its stream-based architecture, to accelerate your development process and improve asset reuse.

Categories in common with Git:
Version Control Systems

Subversion

(64) **3.9** out of **5**

Subversion is an open source version control system. Founded in 2000 by CollabNet, Inc., the Subversion project and software have seen incredible success over the past decade. Subversion has enjoyed and continues to enjoy widespread adoption in both the open source arena and the corporate world.

Categories in common with Git:
Version Control Systems

CVS

(13) **3.5** out of **5**

CVS is a version control system, an important component of Source Configuration Management (SCM). Using it, you can record the history of sources files, and documents. It fills a similar role to the free software RCS, PRCS, and Aegis packages.

Categories in common with Git:
Version Control Systems

List of Version Control System Host

Bitbucket

Buddy

★★★★½ (186) **4.7** out of **5**

Code, build and deploy with no effort

Categories in common with Bitbucket:
Version Control Hosting

GitLab

★★★★½ (755) **4.5** out of **5**

An open source web interface and source control platform based on Git.

Categories in common with Bitbucket:
Version Control Hosting | Peer Code Review

Azure DevOps Server

★★★★☆ (160) **4.2** out of **5**

An enterprise-grade server for teams to share code, track work, and ship software – for any language, all in a single package. It's the perfect complement to your IDE.

Categories in common with Bitbucket:
Version Control Hosting | Peer Code Review

Gerrit

★★★★☆ (29) **4.2** out of **5**

Gerrit Code Review

Gerrit provides web based code review and repository management for the Git version control system.

Categories in common with Bitbucket:
Version Control Hosting | Peer Code Review

Assembla

★★★★☆ (126) **4.2** out of **5**

Assembla is the secure solution for Git, SVN and Perforce source code repositories. For more than 10 years, organizations around the world have trusted Assembla to deliver high-performance version control with integrated project management, in the cloud or on their own infrastructure.

Categories in common with Bitbucket:
Version Control Hosting | Peer Code Review

Helix Core

★★★★☆ (96) **4.2** out of **5**

Helix Core is an industrial-strength version control and collaboration platform that supports flexible workflows. Helix Core eliminates the complexity of large-scale, distributed product development – allowing you to build products faster while protecting your valuable IP.

Categories in common with Bitbucket:
Version Control Hosting | Gaming Tools

Beanstalk

★★★★☆ (21) **4.1** out of **5**

Hassle-free, hosted version control service.

Categories in common with Bitbucket:
Version Control Hosting | Peer Code Review

Phabricator

★★★★½ (27) **4.3** out of **5**

Phabricator is a complete set of tools for developing software. Included apps help you manage tasks and sprints, review code, host git, svn, or mercurial repositories, build with continuous integration, review designs, discuss in internal chat channels, and much more. It's fast, scalable, and fully open source. Install it locally with no limitations, or have us host it for you.

Categories in common with Bitbucket:
Version Control Hosting | Peer Code Review

SourceForge

★★★★½ (59) **4.3** out of **5**

SourceForge is a web-based service that offers a source code repository, downloads mirrors, bug tracking and other features.

Categories in common with Bitbucket:
Version Control Hosting | Peer Code Review

G2 Deals

Buying software just got smarter. **And cheaper.**

See Exclusive Offers

https://www.g2.com/products/bitbucket/competitors/alternatives

```
shahan@shahan-HP-ProBook-450-G7 ~/Desktop/WORKSHOP (master) $ fi
```

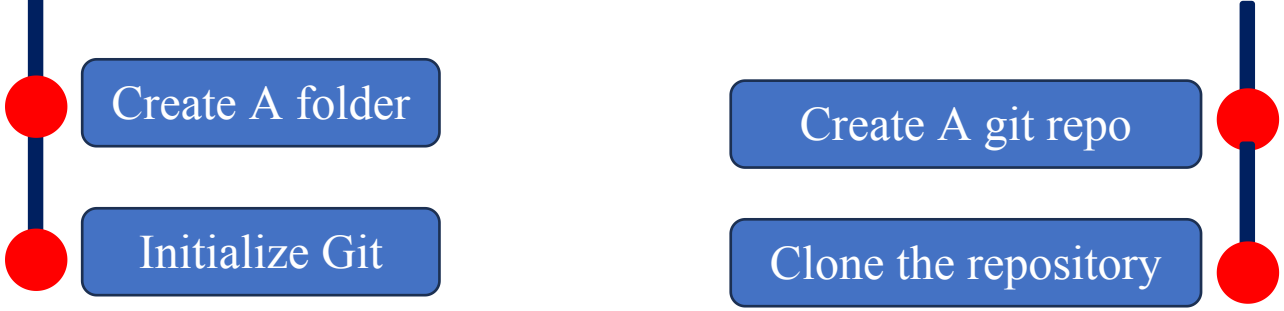
Create A folder

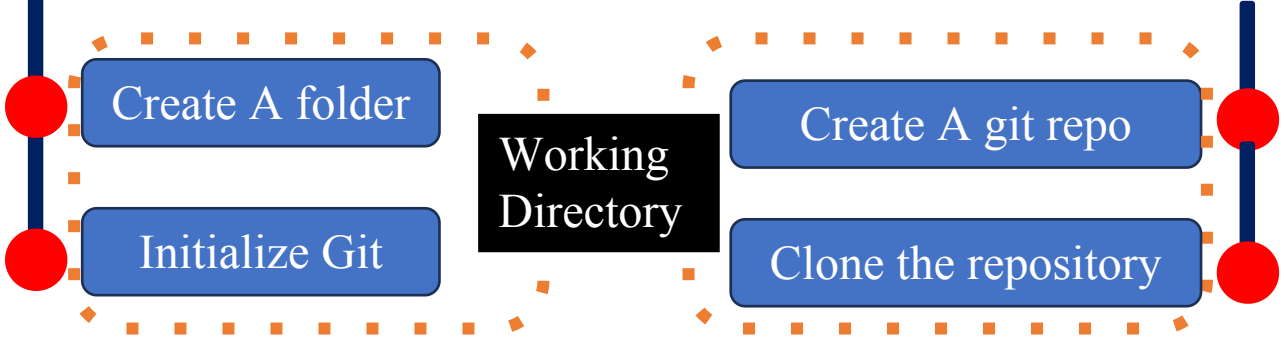
- Create A folder
- Initialize Git

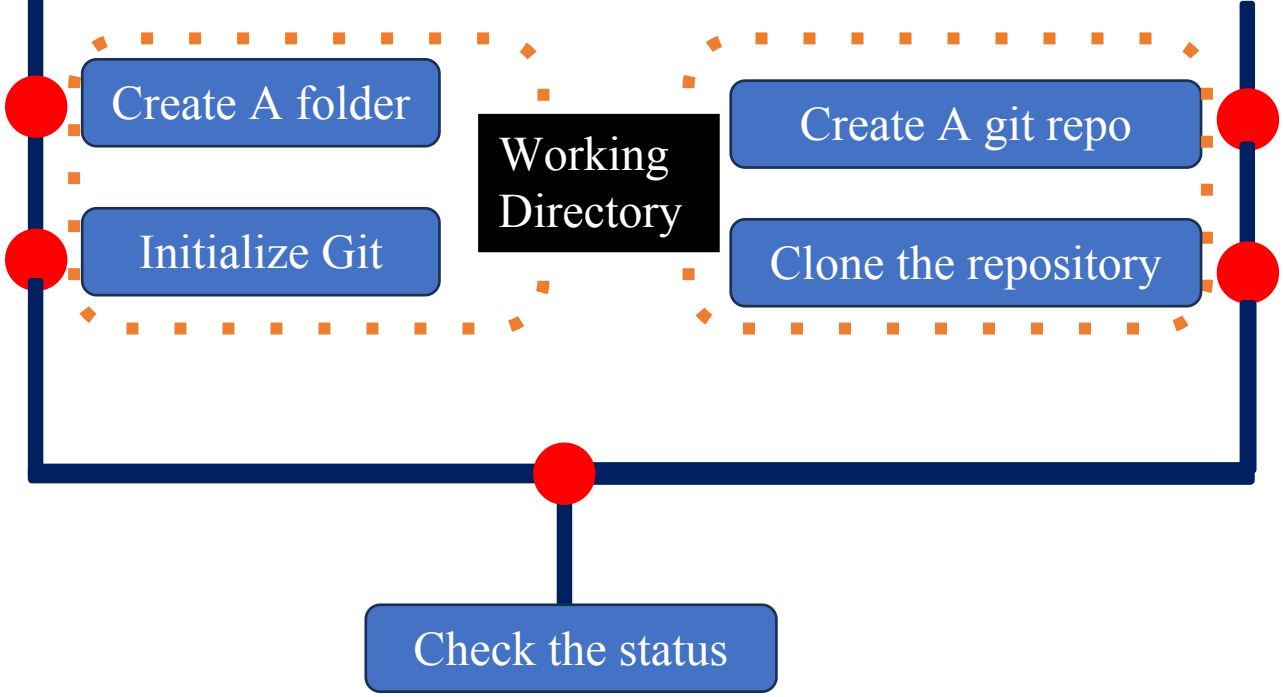
Create A folder

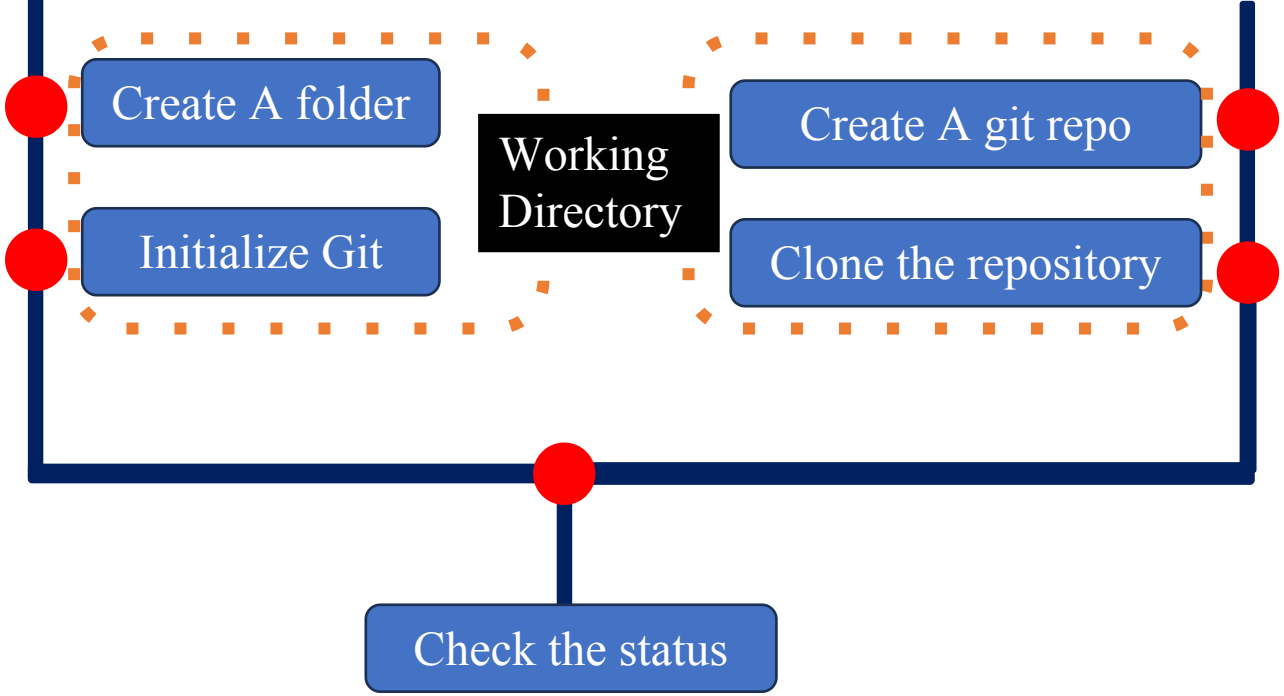
Initialize Git

Create A git repo

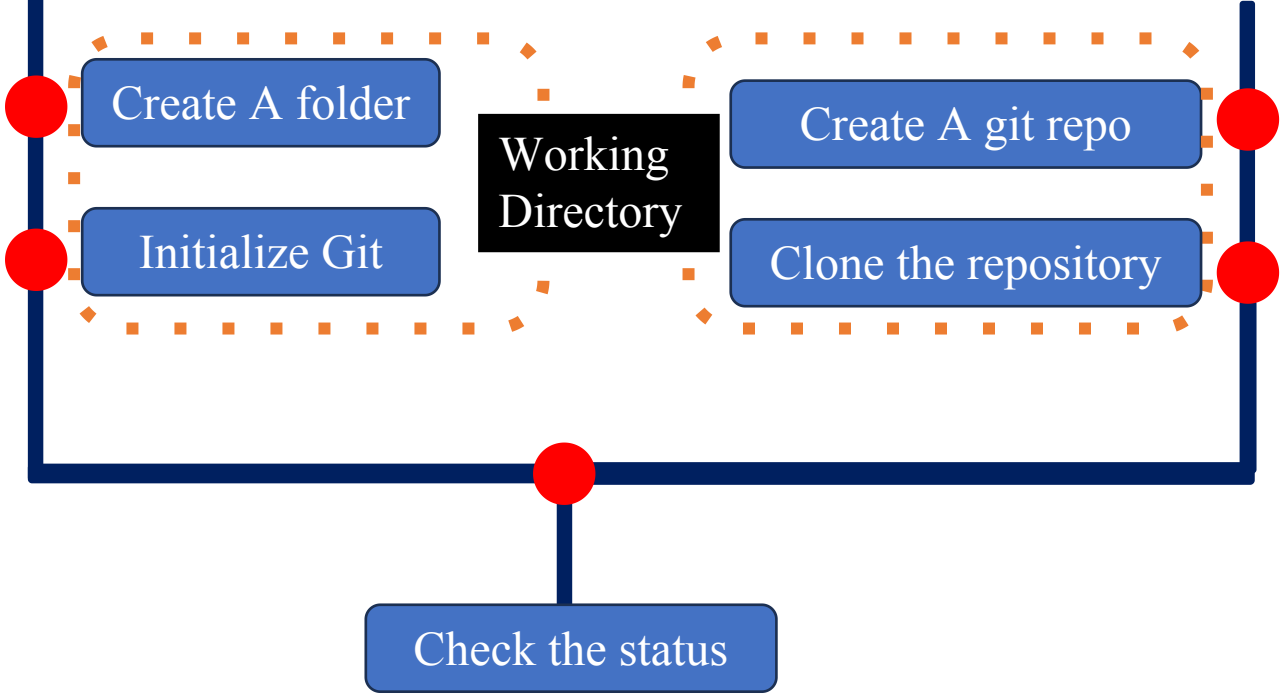

Create A folder
Initialize Git
Create A git repo
Clone the repository


Create A folder
Initialize Git
Working Directory
Create A git repo
Clone the repository


Create A folder
Working Directory
Create A git repo
Initialize Git
Clone the repository
Check the status


Create A folder
Working Directory
Create A git repo
Initialize Git
Clone the repository
Check the status

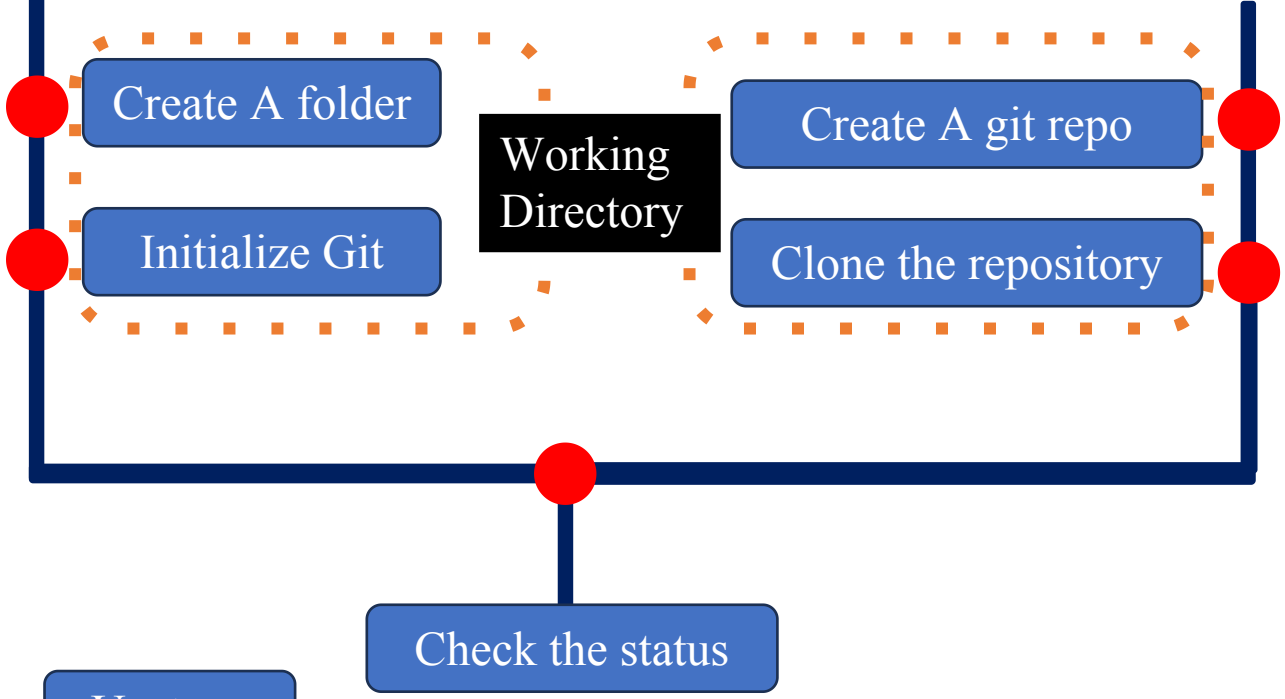
WORKSHOP



WORKSHOP

Demo.c

Demo.c
Sakdsajdasd
Sadasdsad
sadsadsd

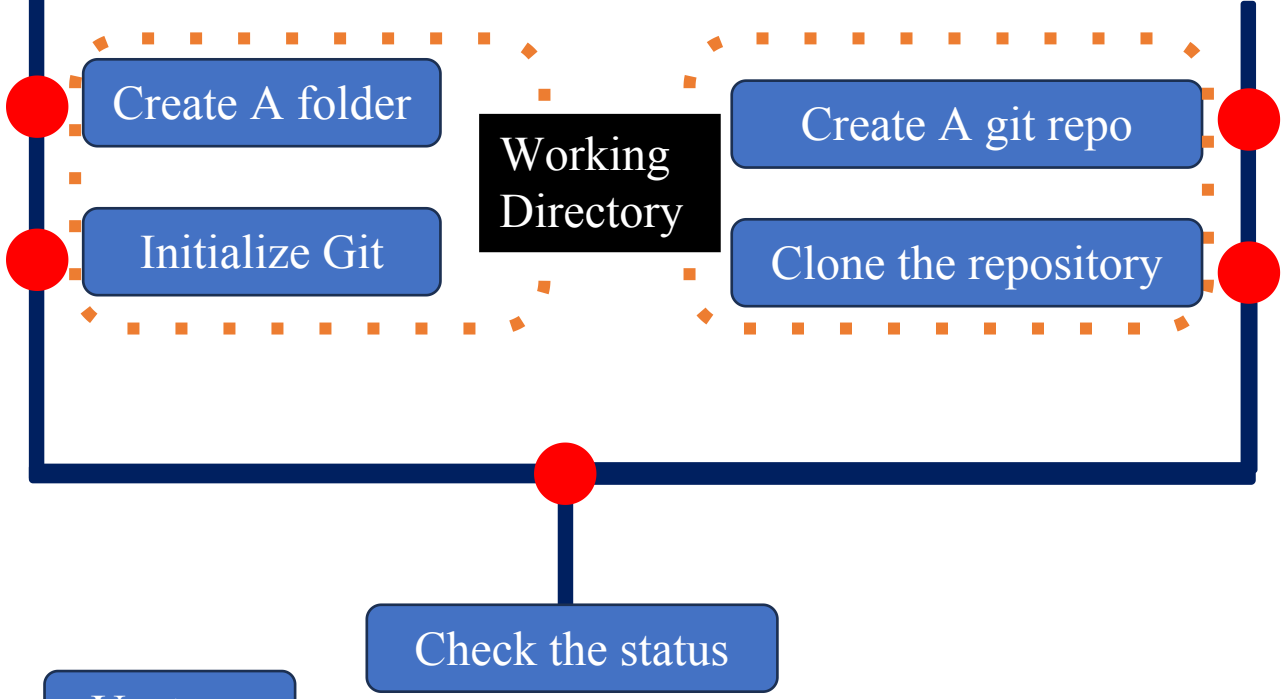


Unstage

WORKSHOP

Demo.c

Demo.c
Sakdsajdasd
Sadasdsad
sadsadsd



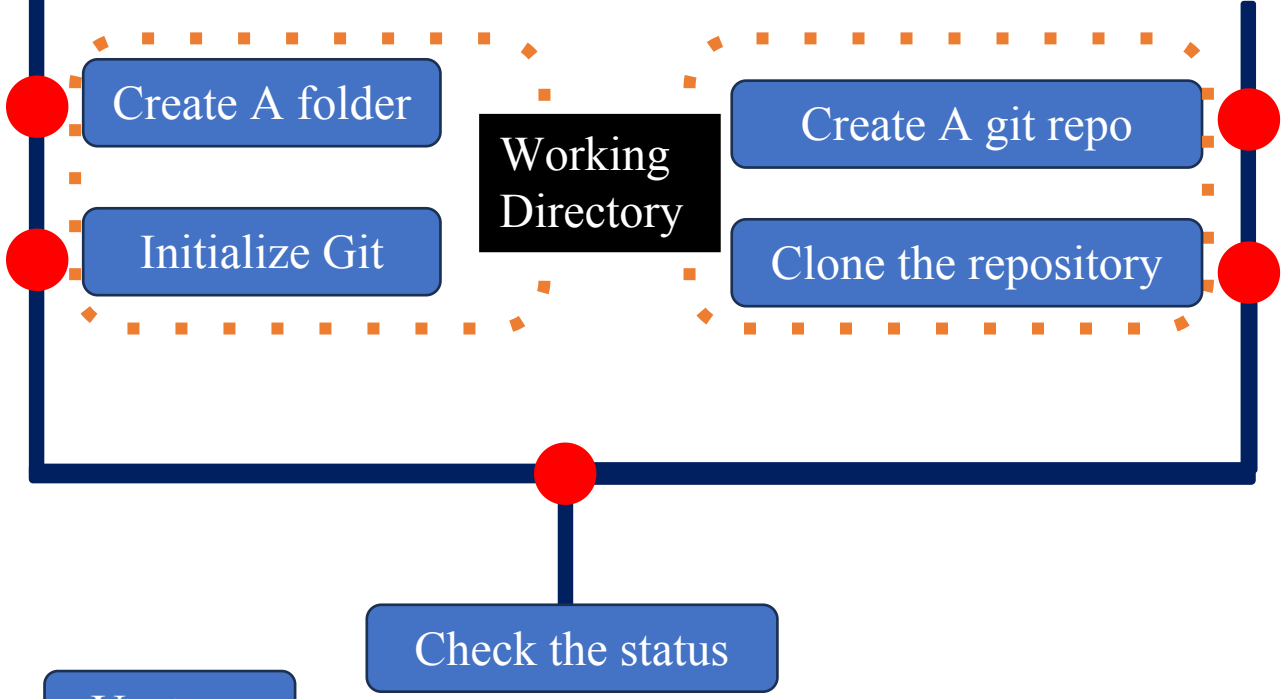
Unstage

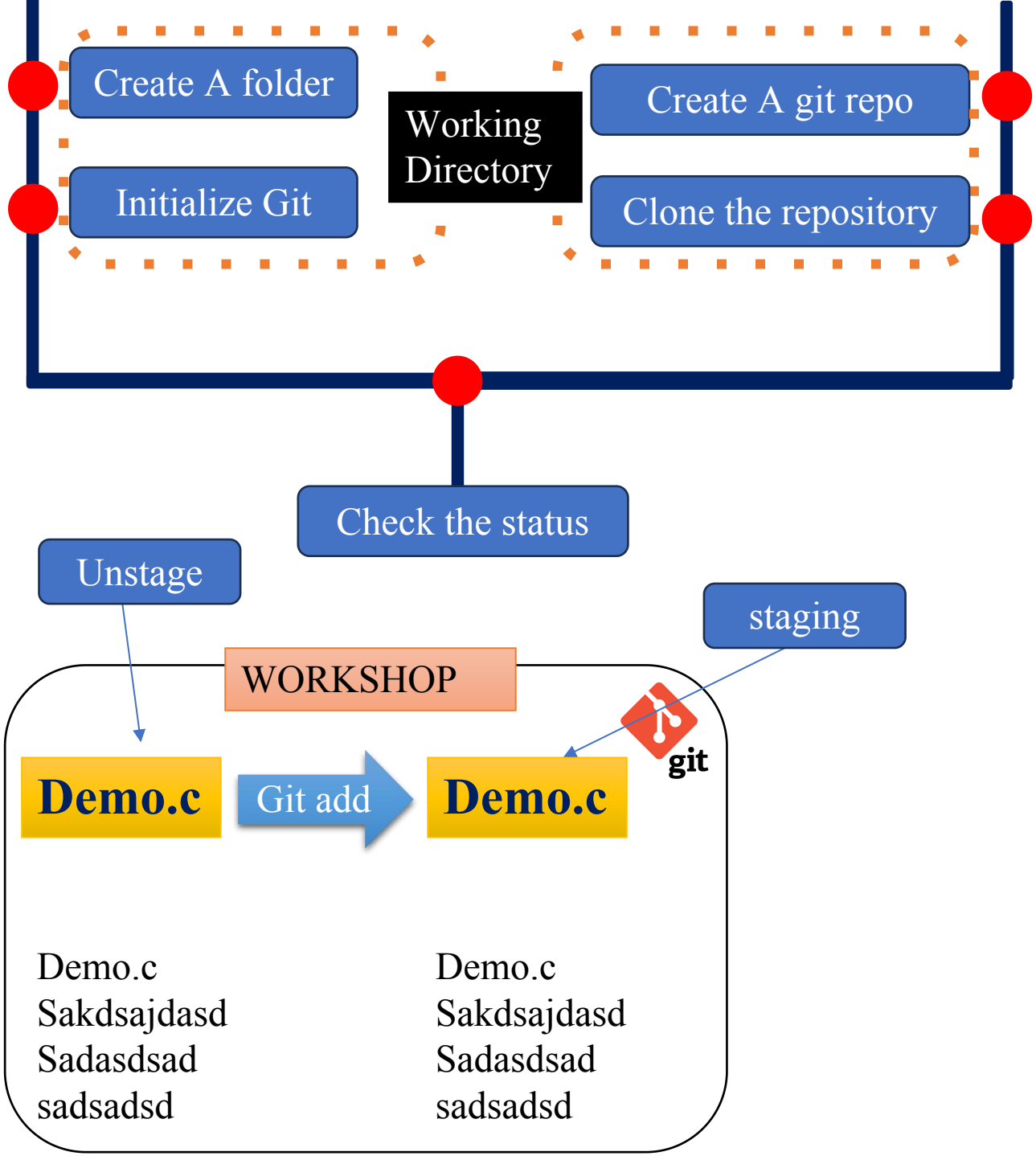
WORKSHOP

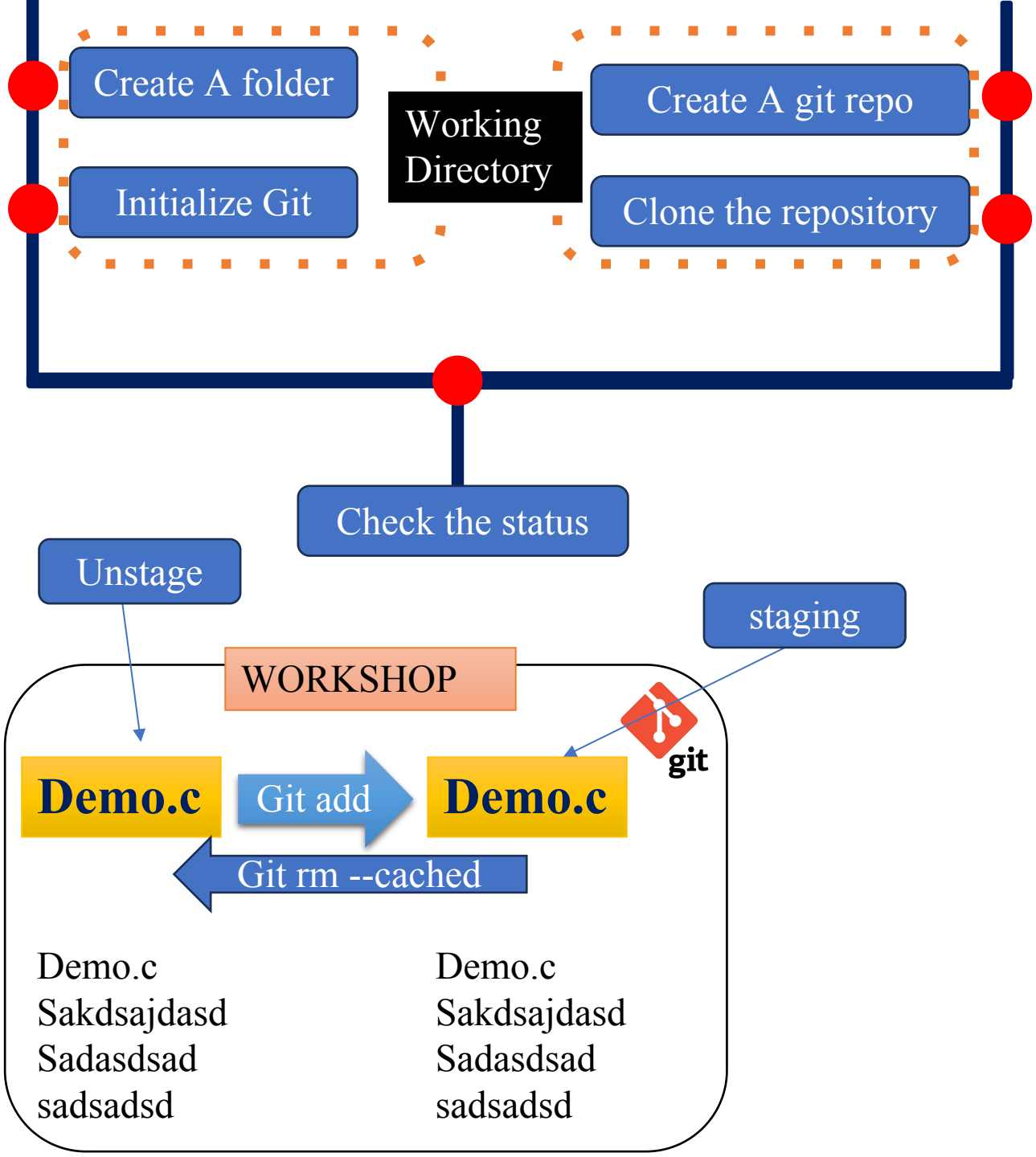
Demo.c

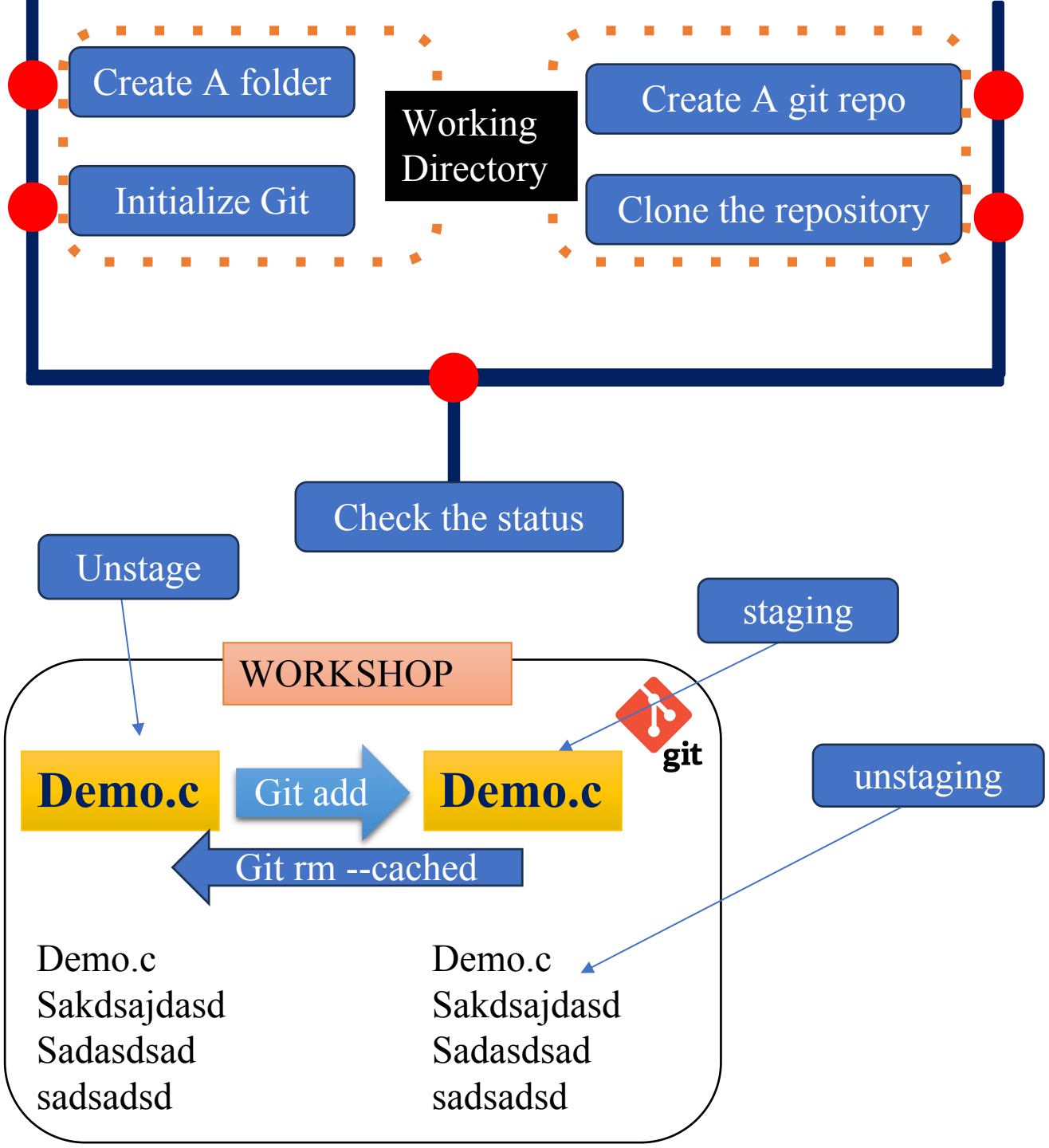
Git add

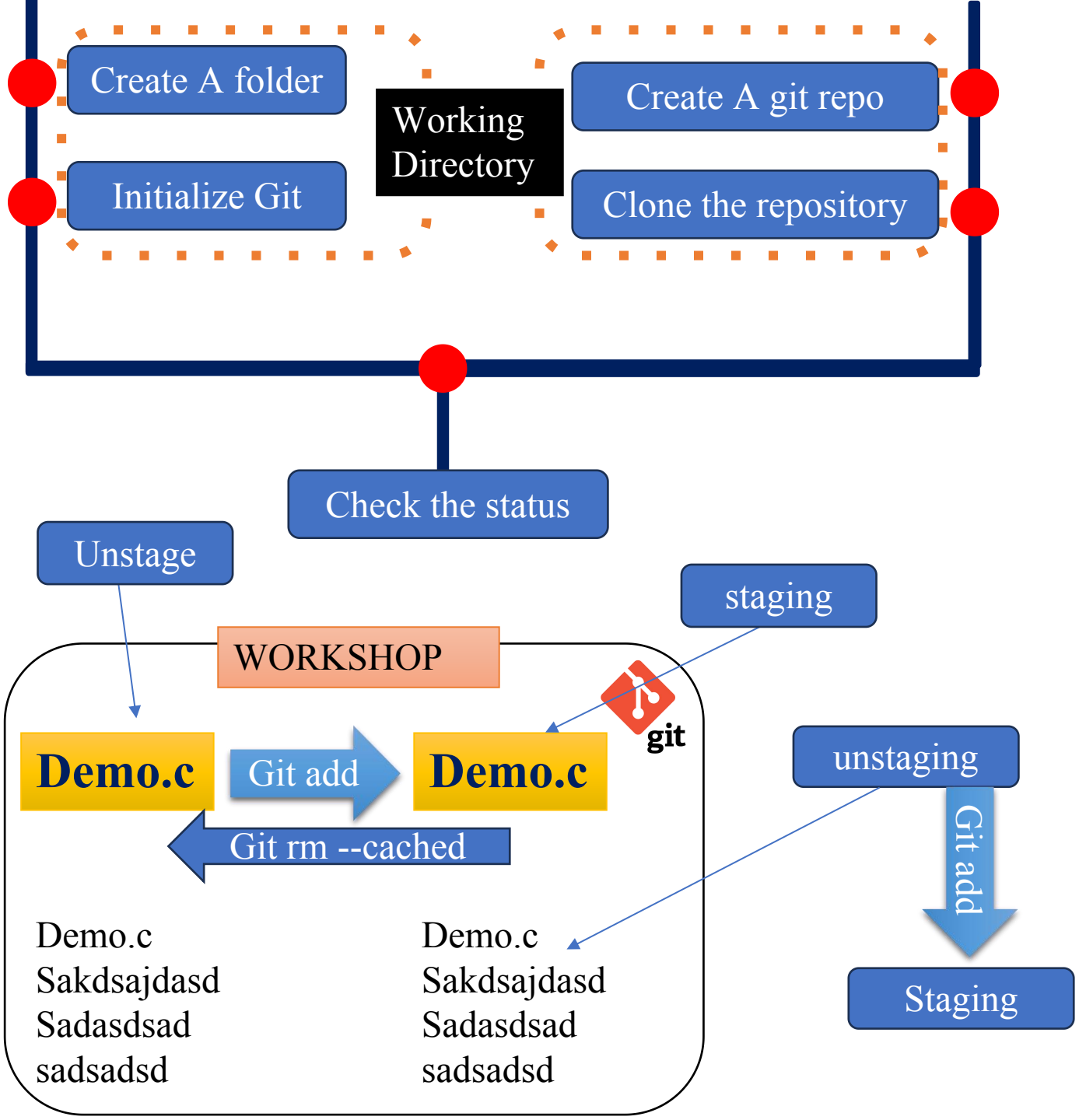
Demo.c
Sakdsajdasd
Sadasdsad
sadsadsd

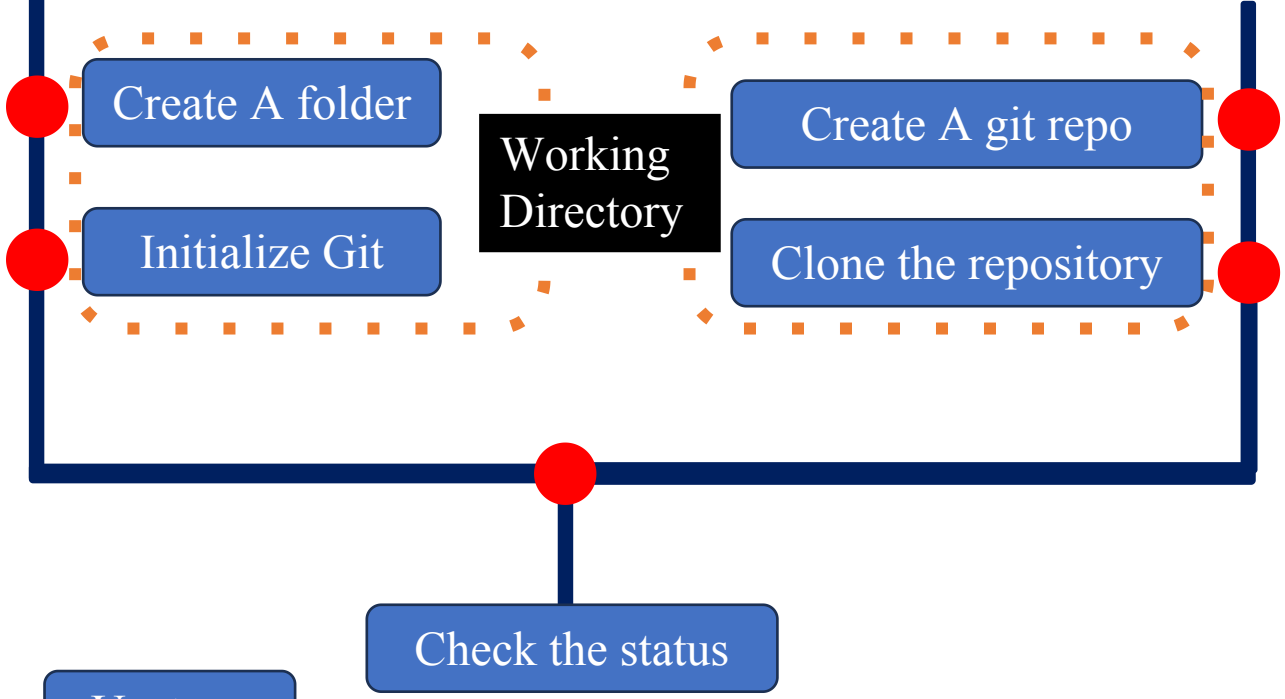

Create A folder
Working Directory
Create A git repo
Initialize Git
Clone the repository
Check the status
Unstage
WORKSHOP
git
Demo.c
Git add
Demo.c
Demo.c
Sakdsajdasd
Sadasdsad
sadsadsd
Demo.c
Sakdsajdasd
Sadasdsad
sadsadsd

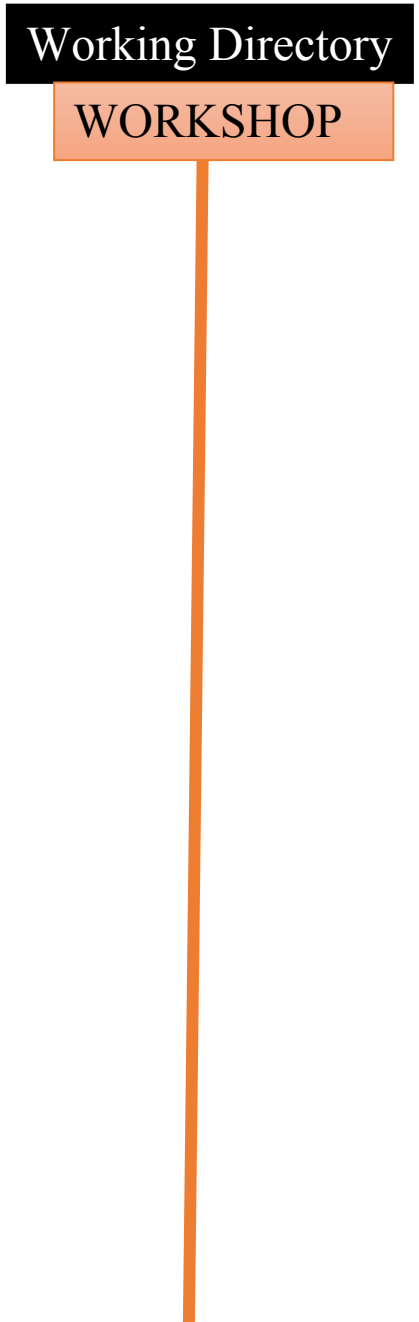

Create A folder
Working Directory
Create A git repo
Initialize Git
Clone the repository
Check the status
Unstage
staging
WORKSHOP
git
Demo.c
Git add
Demo.c
Demo.c
Sakdsajdasd
Sadasdsad
sadsadsd
Demo.c
Sakdsajdasd
Sadasdsad
sadsadsd

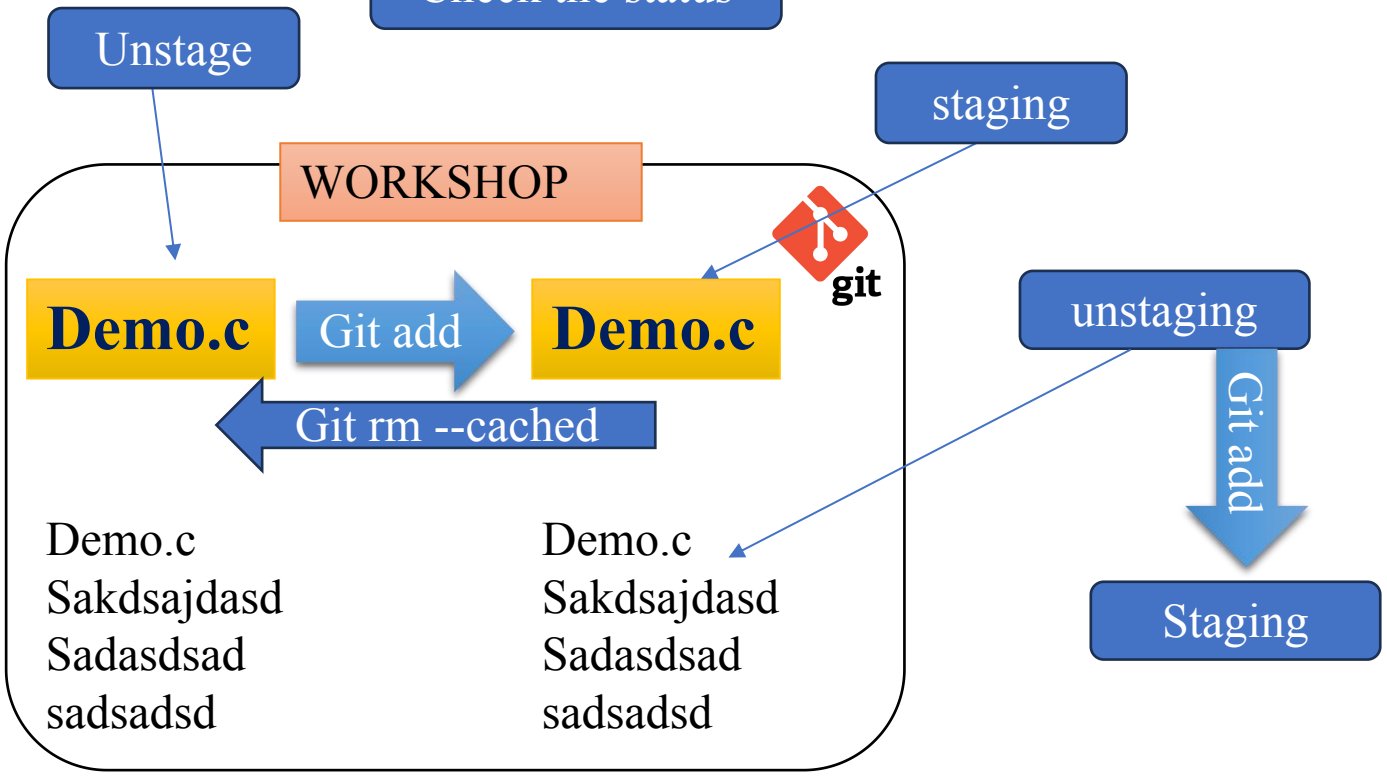

Create A folder
Working Directory
Create A git repo
Initialize Git
Clone the repository
Check the status
Unstage
staging
WORKSHOP
git
Demo.c
Git add
Demo.c
Git rm --cached
Demo.c
Sakdsajdasd
Sadasdsad
sadsadsd
Demo.c
Sakdsajdasd
Sadasdsad
sadsadsd

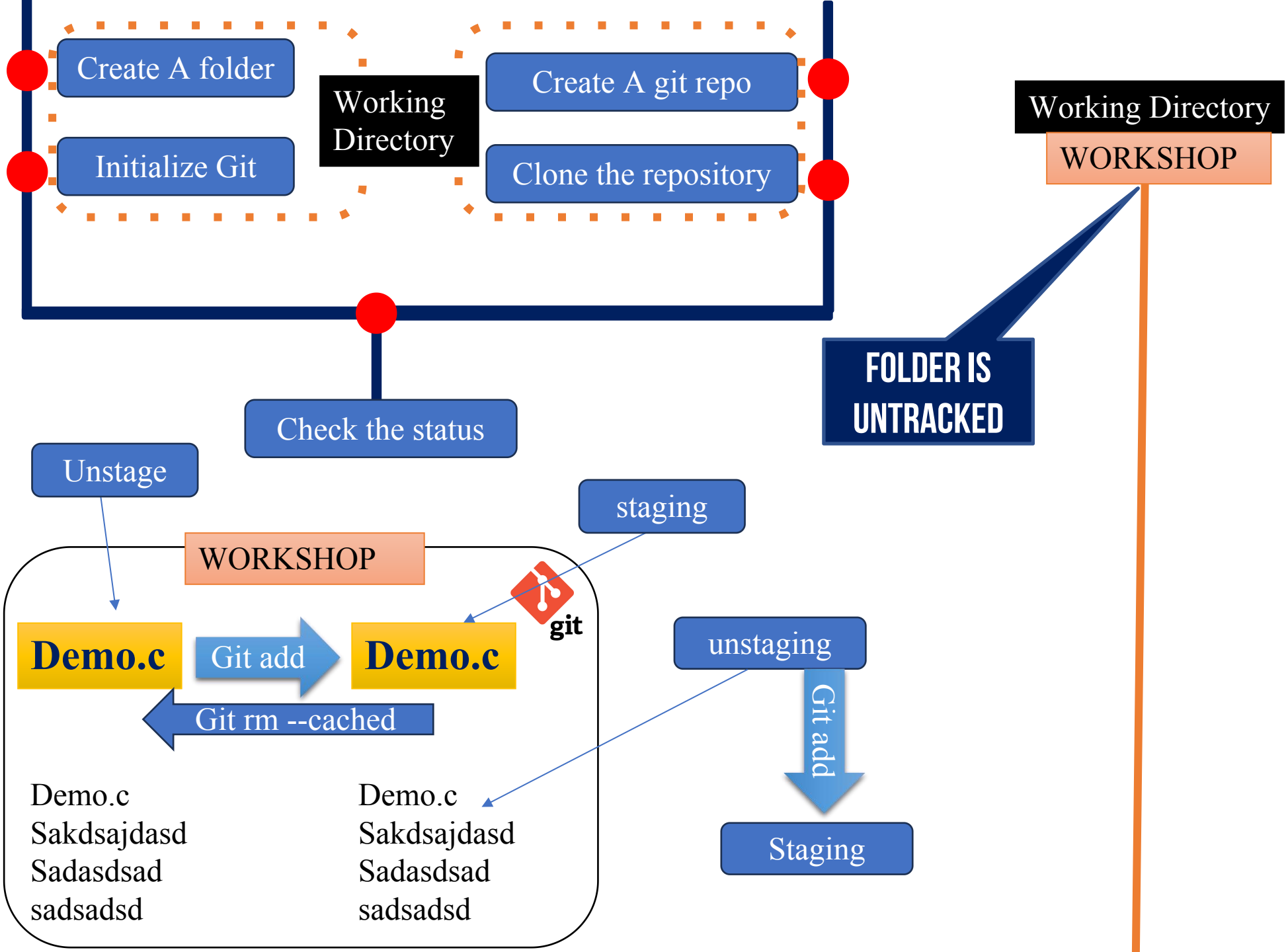

Create A folder
Working Directory
Create A git repo
Initialize Git
Clone the repository
Check the status
Unstage
staging
WORKSHOP
git
Demo.c
Git add
Demo.c
unstaging
Git rm --cached
Demo.c
Sakdsajdasd
Sadasdsad
sadsadsd
Demo.c
Sakdsajdasd
Sadasdsad
sadsadsd

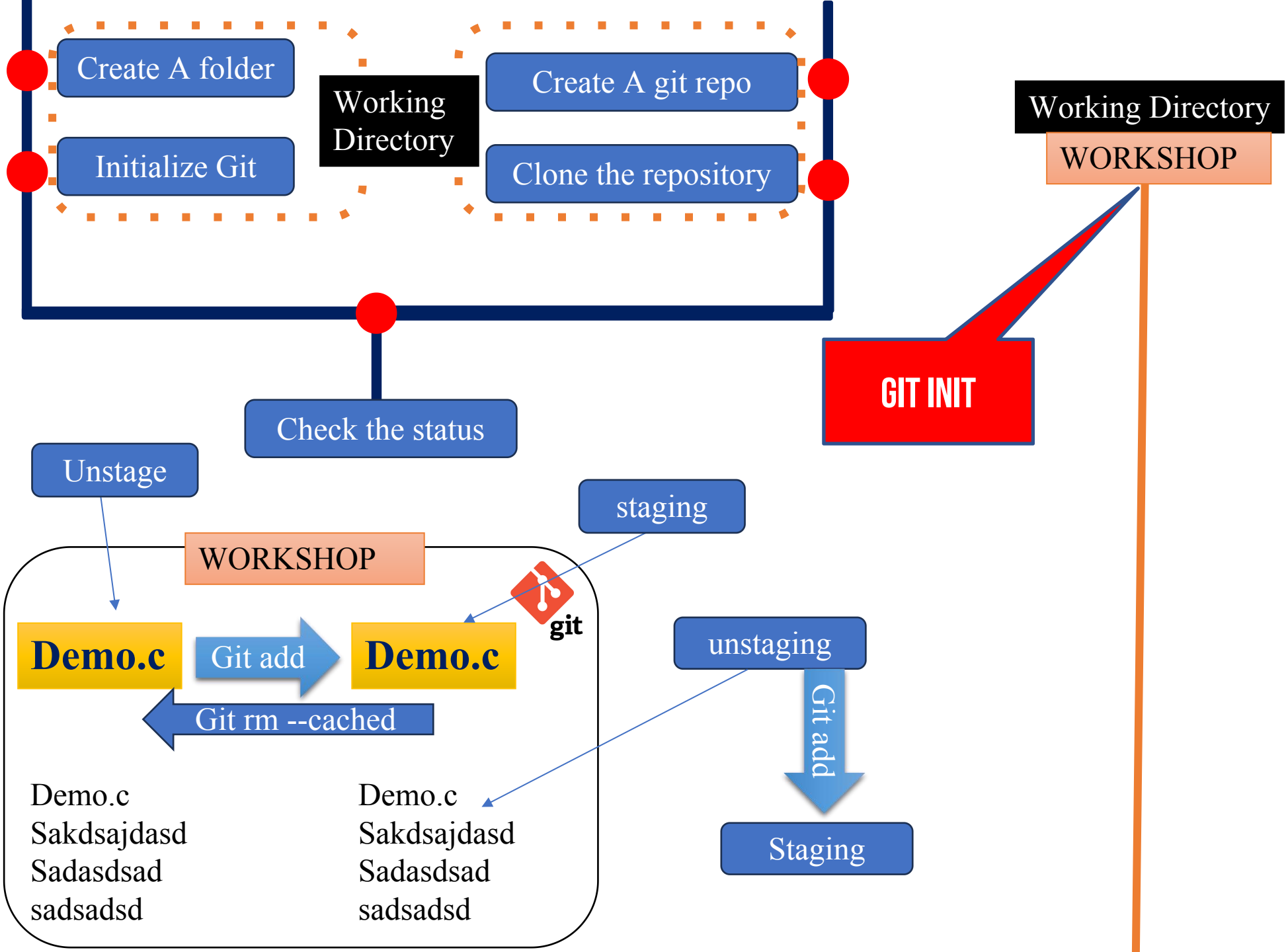

Create A folder
Working Directory
Create A git repo
Initialize Git
Clone the repository
Check the status
Unstage
staging
WORKSHOP
git
Demo.c
Git add
Demo.c
Git rm --cached
unstaging
Git add
Staging
Demo.c
Sakdsajdasd
Sadasdsad
sadsadsd
Demo.c
Sakdsajdasd
Sadasdsad
sadsadsd

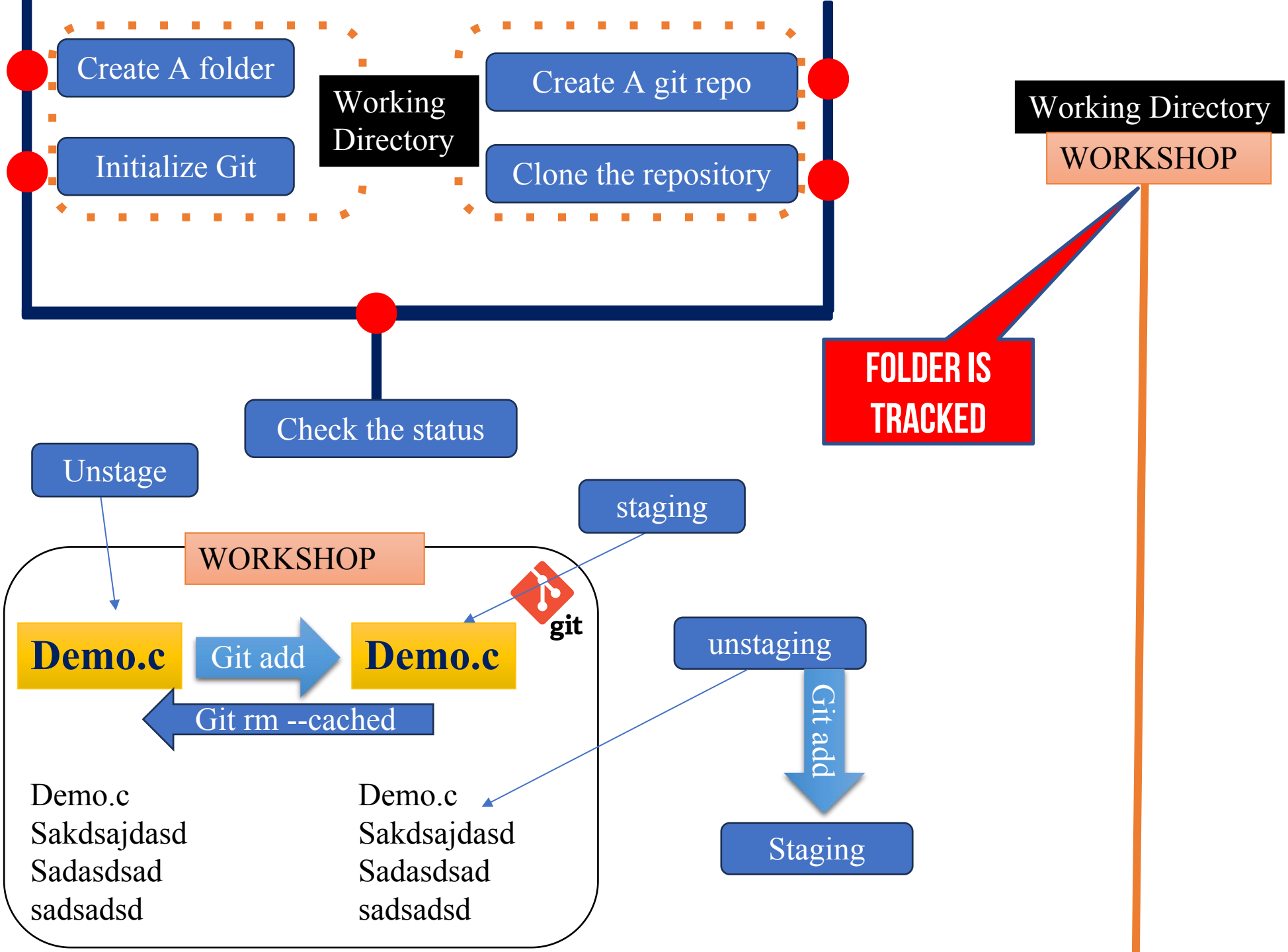

Create A folder
Working Directory
Create A git repo
Initialize Git
Clone the repository

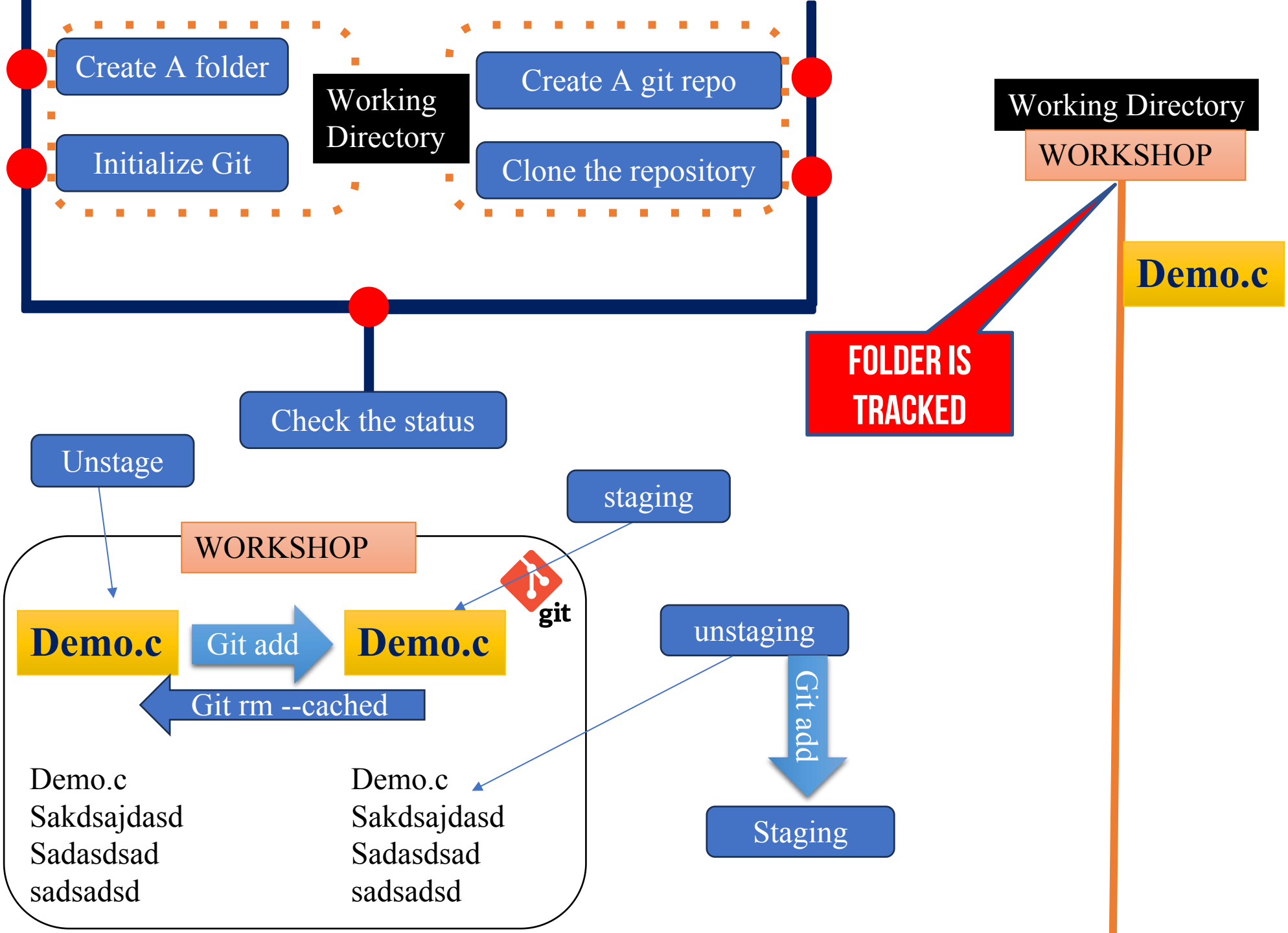

Working Directory
WORKSHOP

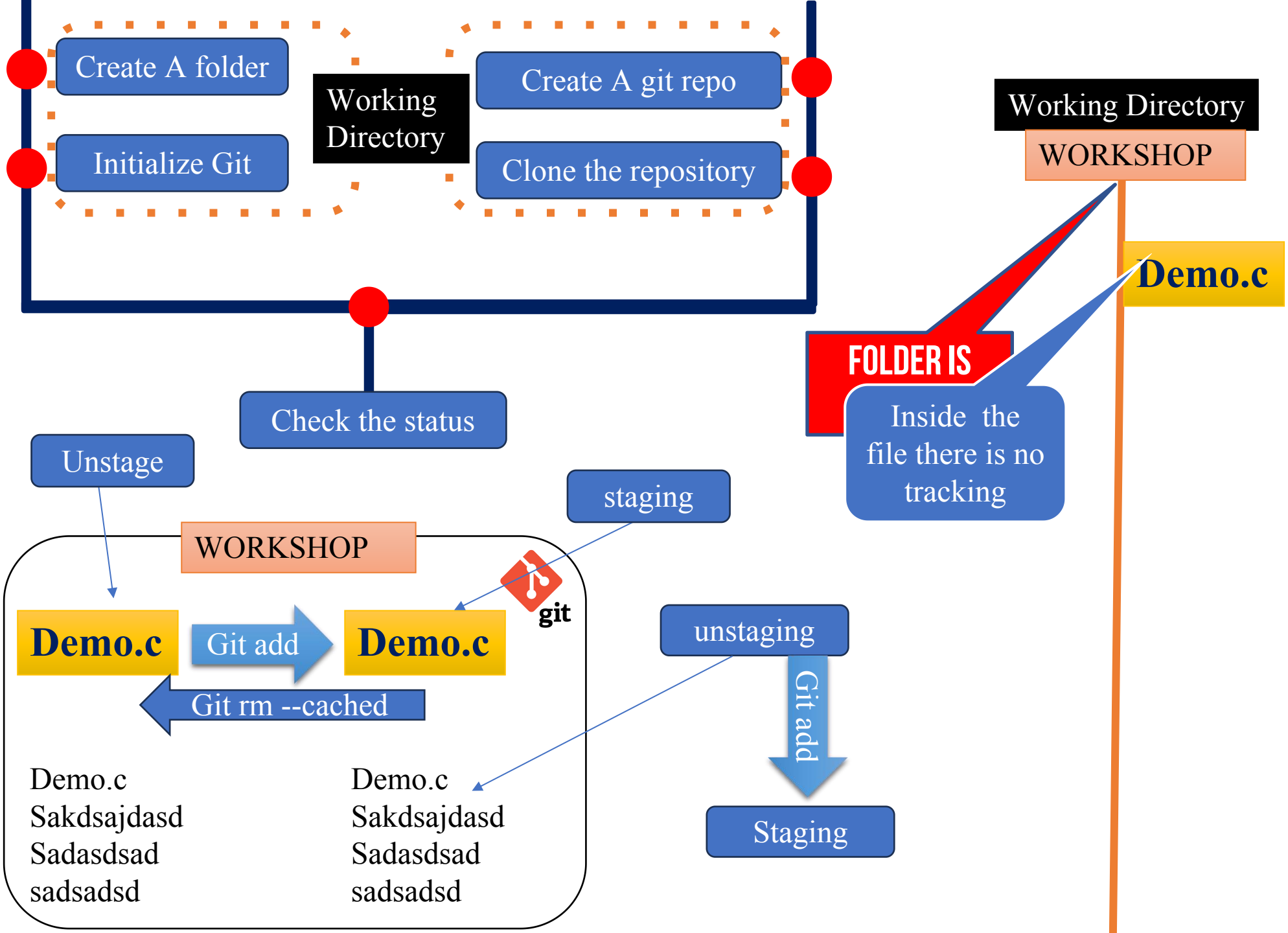

Check the status
Unstage
staging
WORKSHOP
git
Demo.c
Git add
Demo.c
Git rm --cached
unstaging
Git add
Staging
Demo.c
Sakdsajdasd
Sadasdsad
sadsadsd
Demo.c
Sakdsajdasd
Sadasdsad
sadsadsd

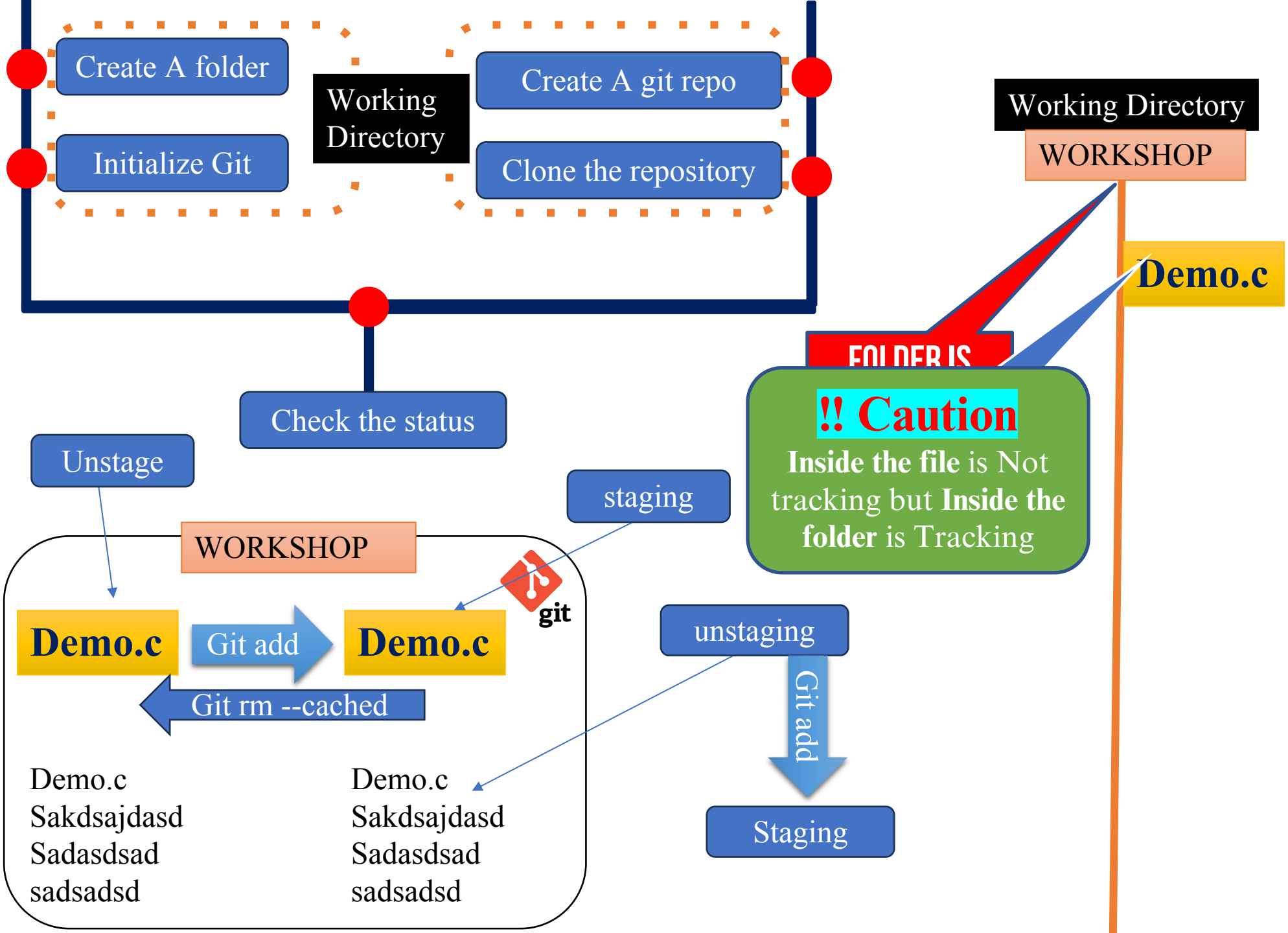

Create A folder
Working Directory
Create A git repo
Initialize Git
Clone the repository
Check the status
Working Directory
WORKSHOP
FOLDER IS UNTRACKED
Unstage
staging
WORKSHOP
git
Demo.c
Git add
Demo.c
Git rm --cached
unstaging
Git add
Staging
Demo.c
Sakdsajdasd
Sadasdsad
sadsadsd
Demo.c
Sakdsajdasd
Sadasdsad
sadsadsd

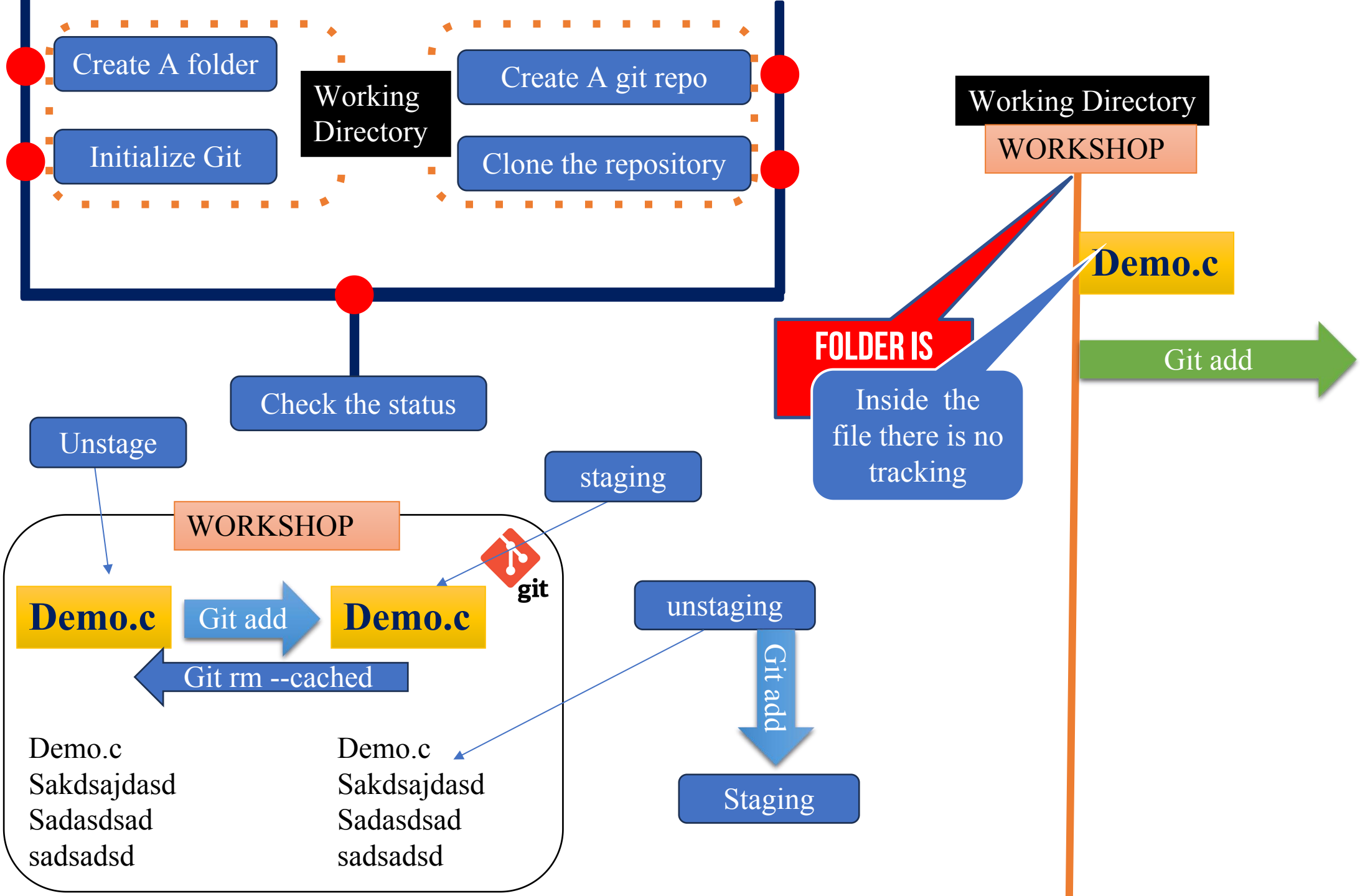

Create A folder
Working Directory
Create A git repo
Initialize Git
Clone the repository
Check the status
Working Directory
WORKSHOP
GIT INIT
Unstage
staging
WORKSHOP
Demo.c
Git add
Demo.c
git
Git rm --cached
unstaging
Git add
Staging
Demo.c
Sakdsajdasd
Sadasdsad
sadsadsd
Demo.c
Sakdsajdasd
Sadasdsad
sadsadsd

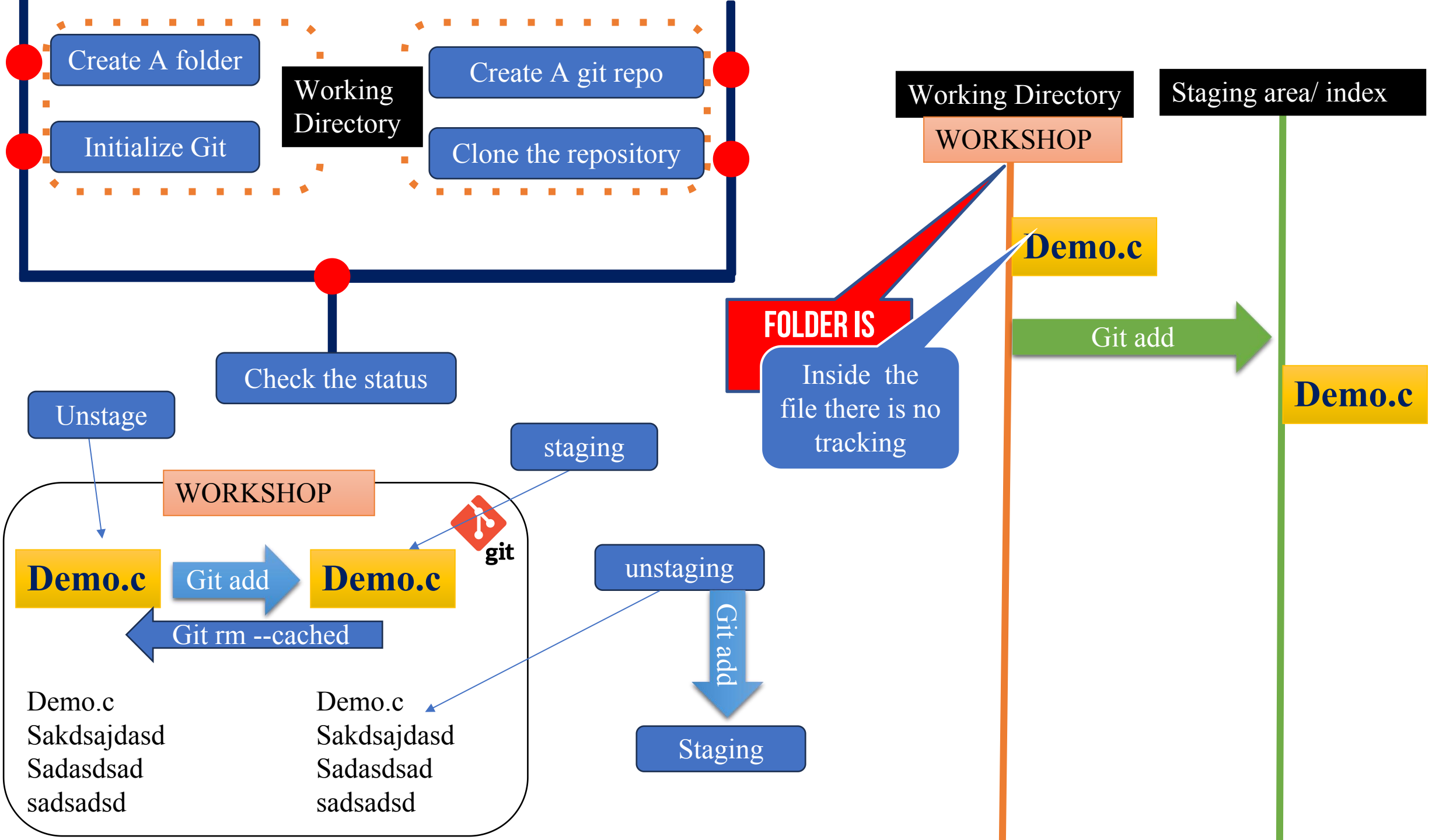

Create A folder
Working Directory
Create A git repo
Initialize Git
Clone the repository
Check the status
Working Directory
WORKSHOP
FOLDER IS TRACKED
Unstage
staging
WORKSHOP
git
Demo.c
Git add
Demo.c
Git rm --cached
unstaging
Git add
Staging
Demo.c
Sakdsajdasd
Sadasdsad
sadsadsd
Demo.c
Sakdsajdasd
Sadasdsad
sadsadsd

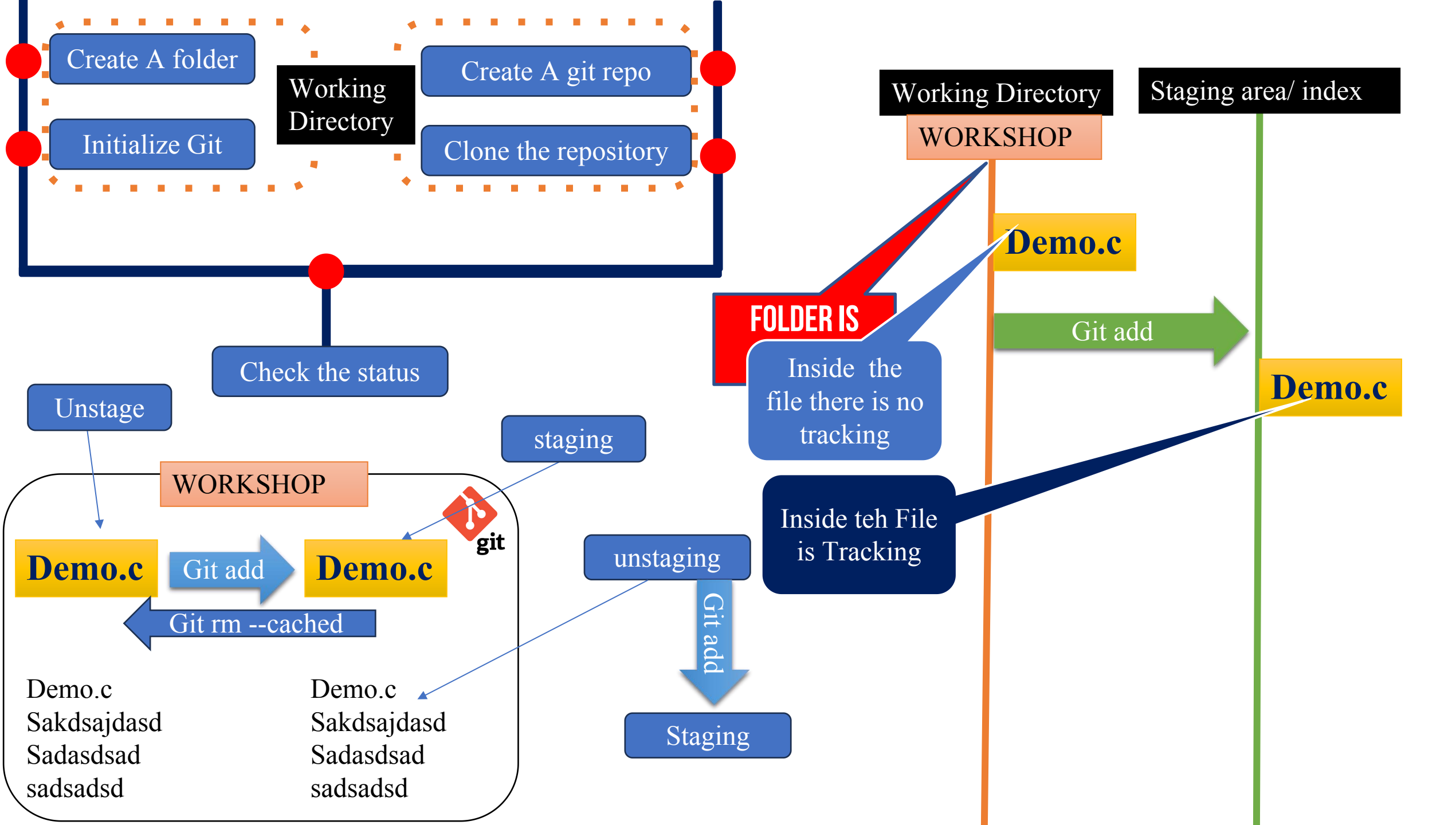

Create A folder
Working Directory
Create A git repo
Initialize Git
Clone the repository
Check the status
Working Directory
WORKSHOP
Demo.c
FOLDER IS TRACKED
Unstage
staging
WORKSHOP
git
Demo.c
Git add
Demo.c
Git rm --cached
unstaging
Git add
Staging
Demo.c
Sakdsajdasd
Sadasdsad
sadsadsd
Demo.c
Sakdsajdasd
Sadasdsad
sadsadsd

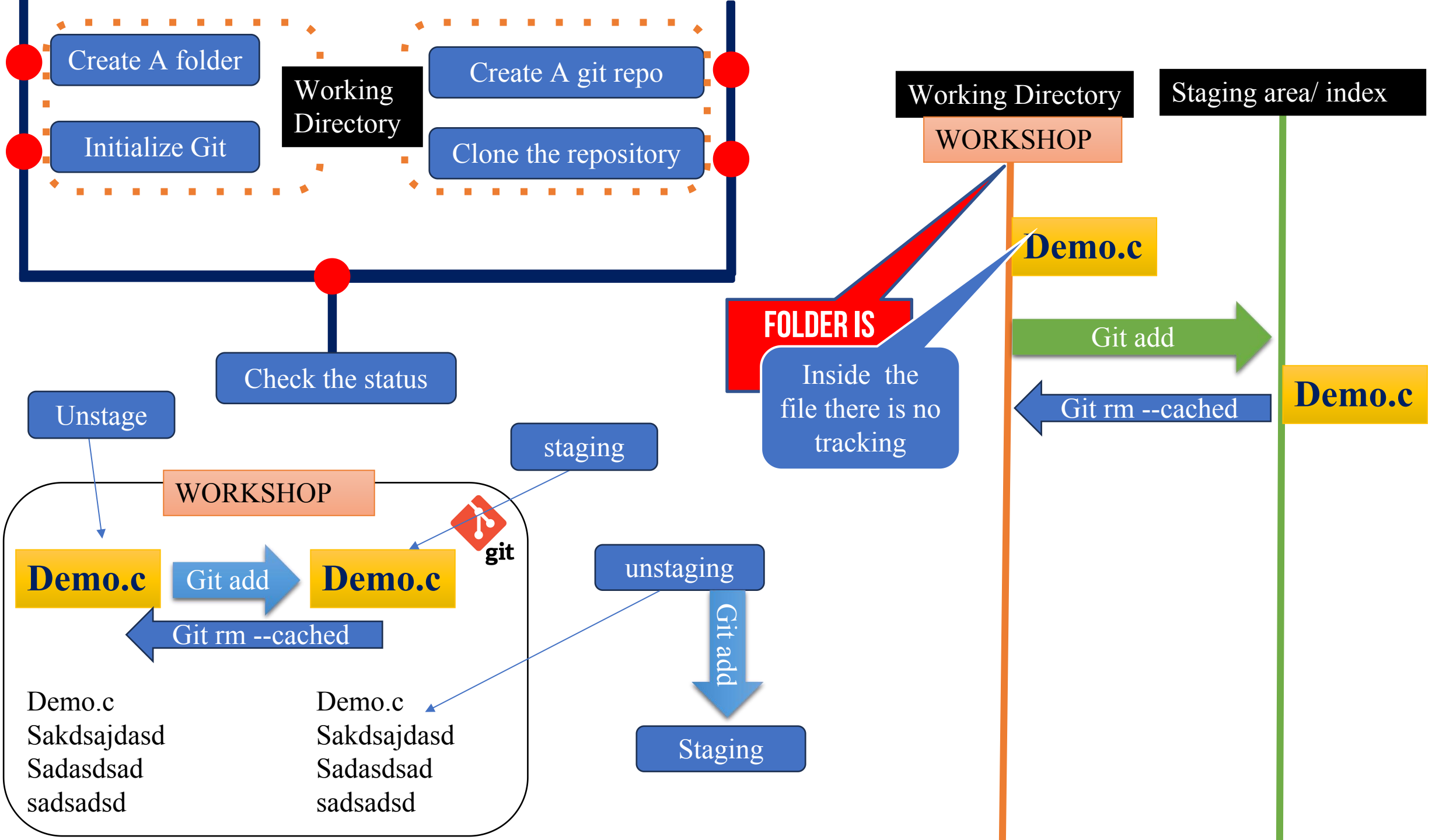

Create A folder
Working Directory
Create A git repo
Initialize Git
Clone the repository
Check the status
Working Directory
WORKSHOP
Demo.c
FOLDER IS
Inside the file there is no tracking
Unstage
staging
WORKSHOP
git
Demo.c
Git add
Demo.c
Git rm --cached
unstaging
Git add
Staging
Demo.c
Sakdsajdasd
Sadasdsad
sadsadsd
Demo.c
Sakdsajdasd
Sadasdsad
sadsadsd

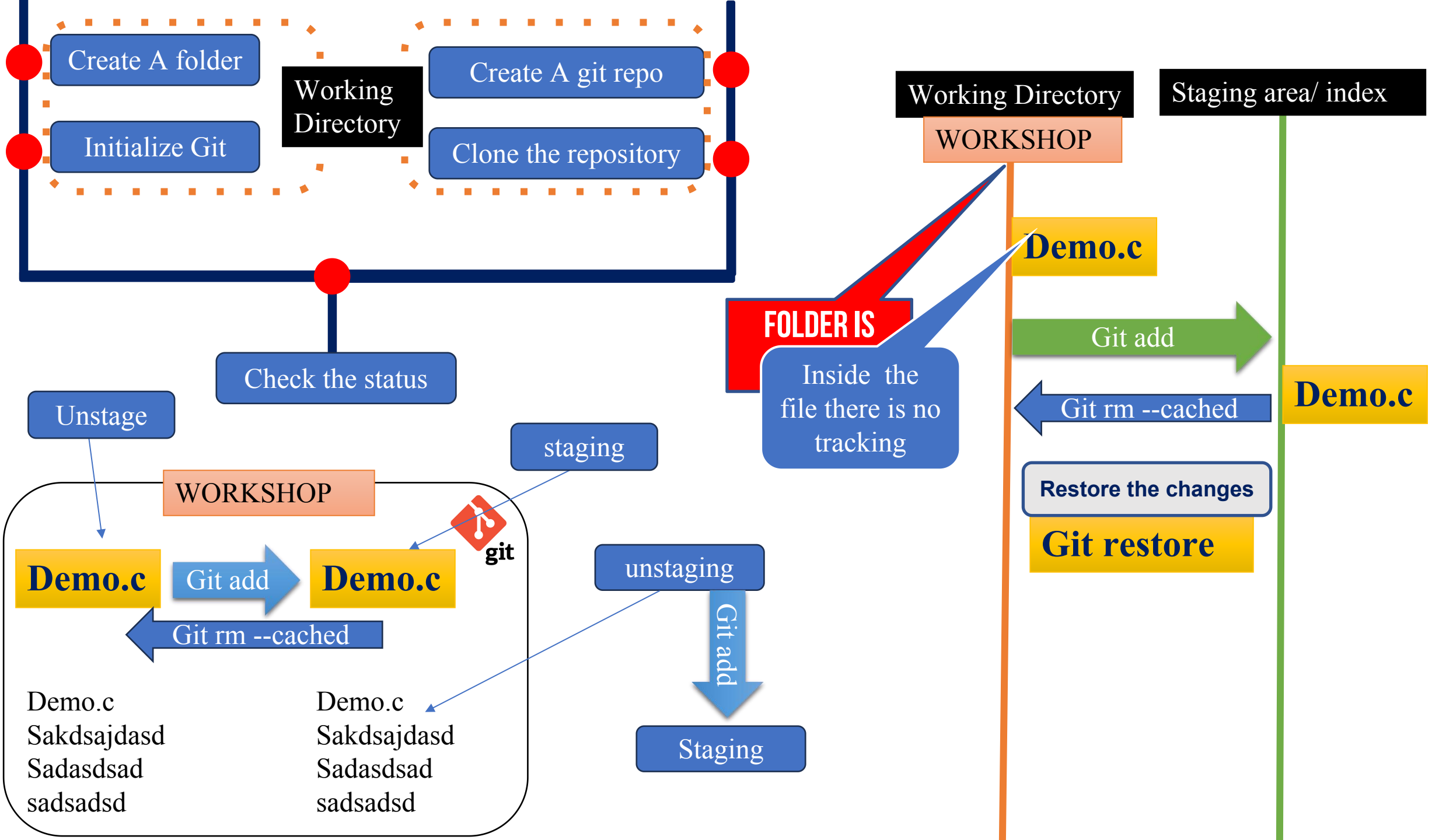

Create A folder
Working Directory
Create A git repo
Initialize Git
Clone the repository
Check the status
Working Directory
WORKSHOP
Demo.c
!! Caution
Inside the file is Not tracking but Inside the folder is Tracking
Unstage
staging
WORKSHOP
git
Demo.c
Git add
Demo.c
Git rm --cached
unstaging
Git add
Staging
Demo.c
Sakdsajdasd
Sadasdsad
sadsadsd
Demo.c
Sakdsajdasd
Sadasdsad
sadsadsd


Create A folder
Working Directory
Create A git repo
Initialize Git
Clone the repository
Check the status
Working Directory
WORKSHOP
Demo.c
FOLDER IS
Inside the file there is no tracking
Git add
Unstage
staging
WORKSHOP
git
Demo.c
Git add
Demo.c
Git rm --cached
unstaging
Git add
Staging
Demo.c
Sakdsajdasd
Sadasdsad
sadsadsd
Demo.c
Sakdsajdasd
Sadasdsad
sadsadsd

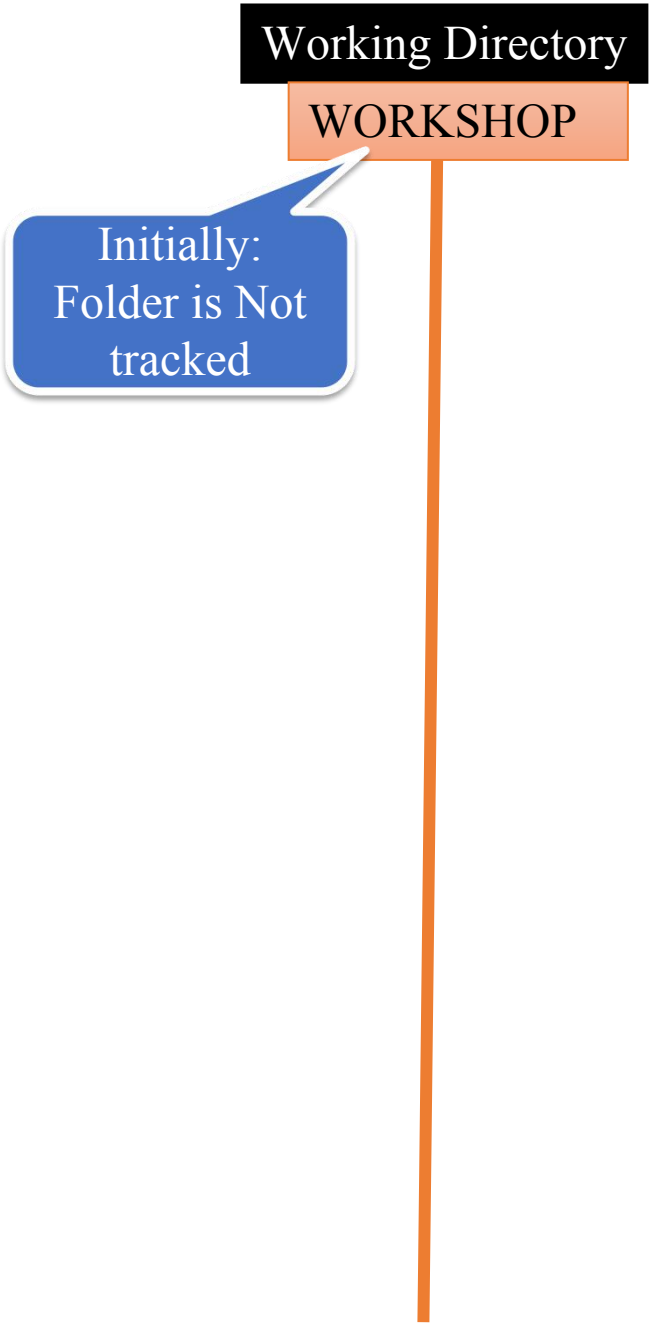

Create A folder
Working Directory
Create A git repo
Initialize Git
Clone the repository
Check the status
Working Directory
WORKSHOP
Staging area/ index
Demo.c
FOLDER IS
Inside the file there is no tracking
Git add
Demo.c
Unstage
staging
WORKSHOP
Demo.c
Git add
Demo.c
git
Git rm --cached
unstaging
Git add
Staging
Demo.c
Sakdsajdasd
Sadasdsad
sadsadsd
Demo.c
Sakdsajdasd
Sadasdsad
sadsadsd

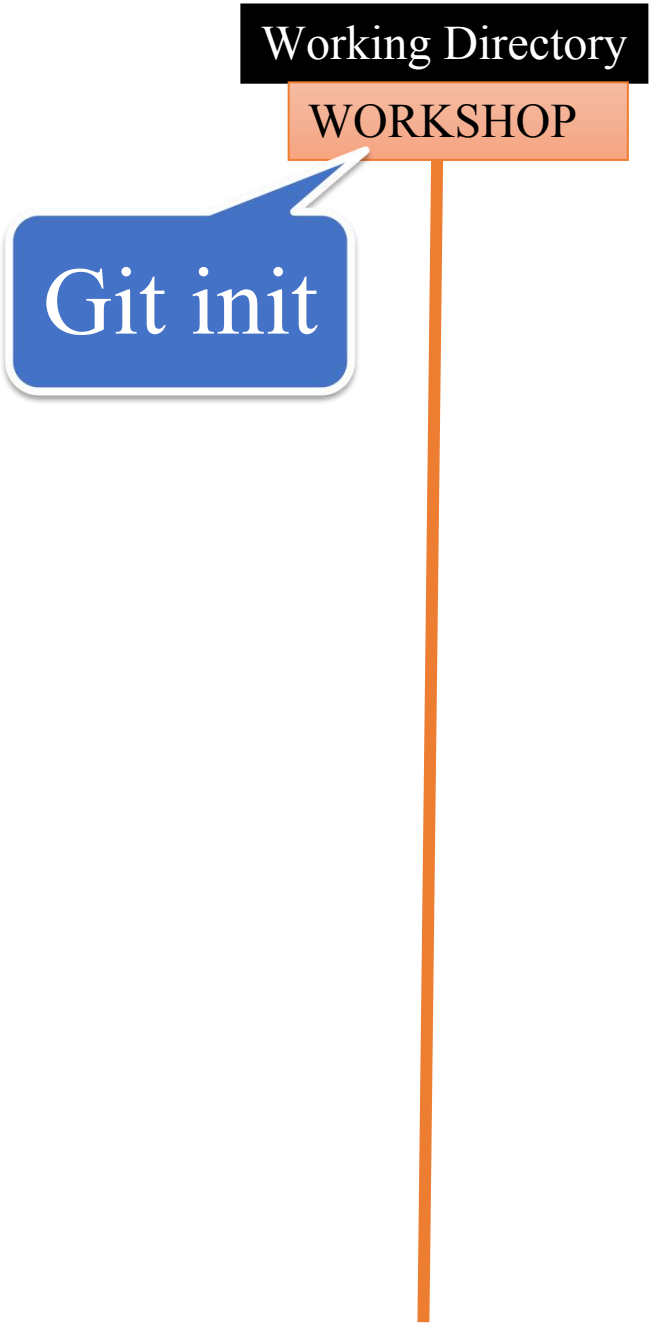

Create A folder
Working Directory
Create A git repo
Initialize Git
Clone the repository
Check the status
Working Directory
WORKSHOP
Staging area/ index
Demo.c
FOLDER IS
Inside the file there is no tracking
Git add
Demo.c
Unstage
staging
WORKSHOP
Inside teh File is Tracking
Demo.c
Git add
Demo.c
git
unstaging
Git add
Git rm --cached
Staging
Demo.c
Sakdsajdasd
Sadasdsad
sadsadsd
Demo.c
Sakdsajdasd
Sadasdsad
sadsadsd

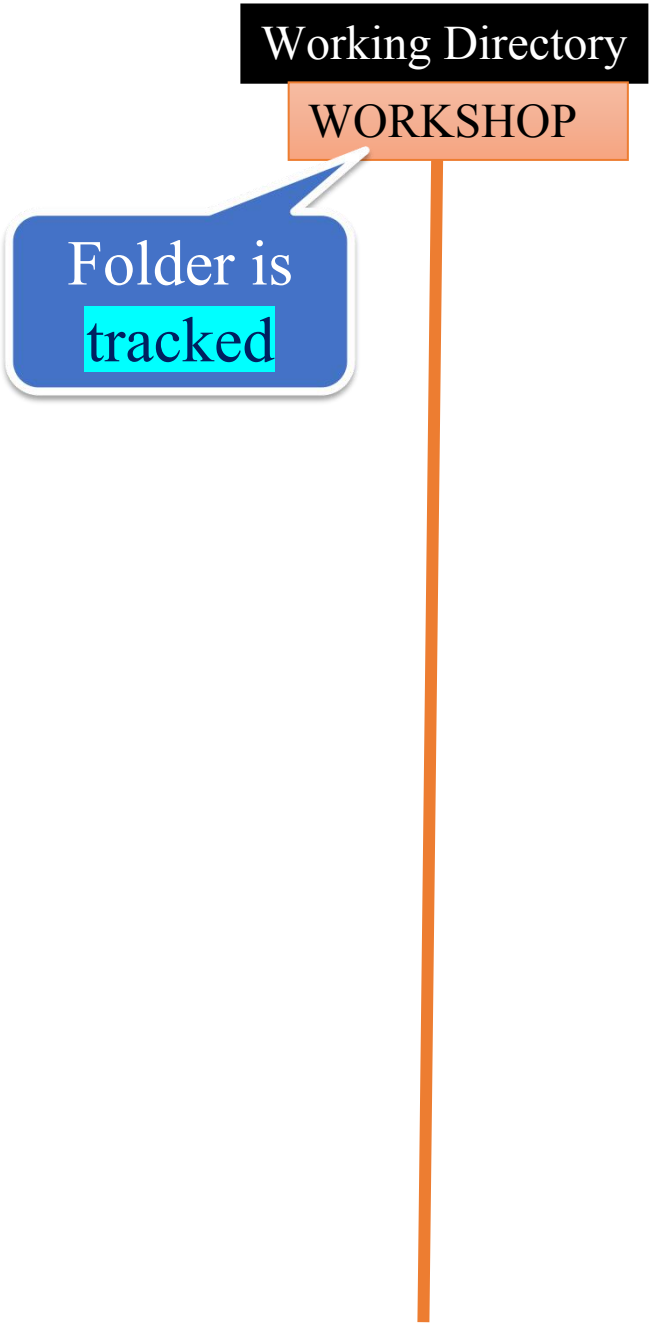

Create A folder
Working Directory
Create A git repo
Initialize Git
Clone the repository
Check the status
Working Directory
Staging area/ index
WORKSHOP
Demo.c
FOLDER IS
Inside the file there is no tracking
Git add
Git rm --cached
Demo.c
Unstage
staging
WORKSHOP
Demo.c
Git add
Demo.c
git
Git rm --cached
unstaging
Git add
Staging
Demo.c
Sakdsajdasd
Sadasdsad
sadsadsd
Demo.c
Sakdsajdasd
Sadasdsad
sadsadsd

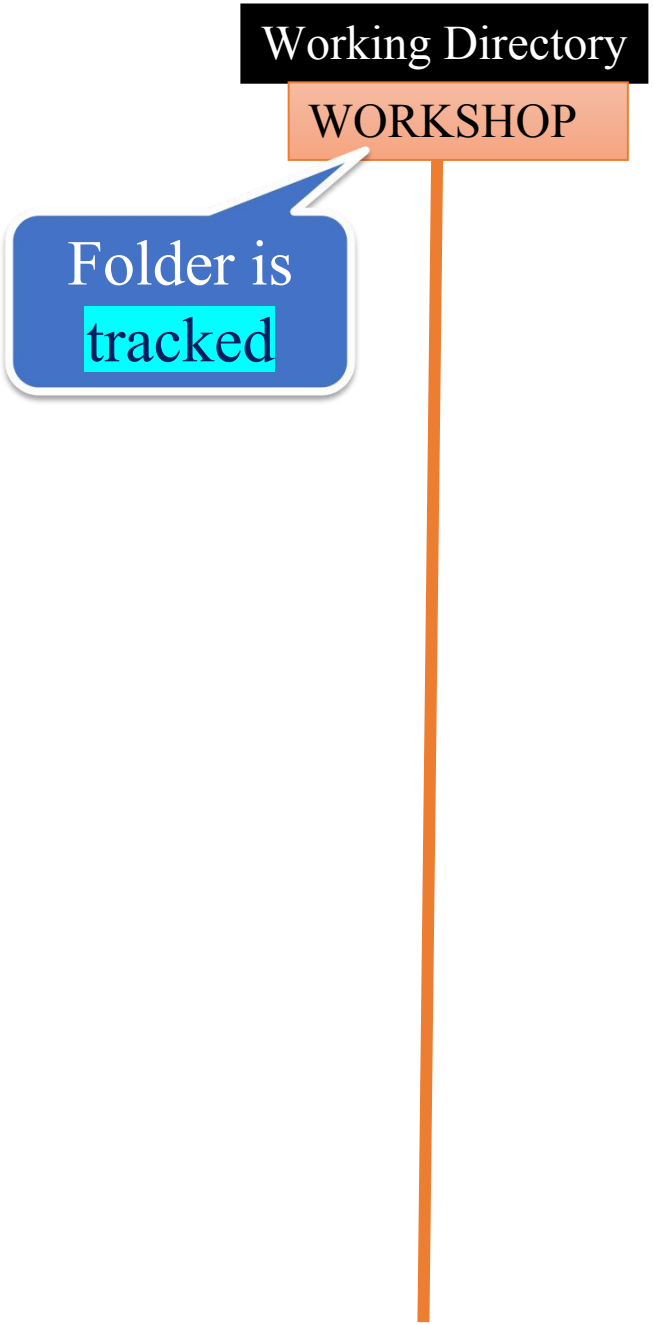

Create A folder
Working Directory
Create A git repo
Initialize Git
Clone the repository
Check the status
Working Directory
Staging area/ index
WORKSHOP
Demo.c
FOLDER IS
Inside the file there is no tracking
Git add
Demo.c
Git rm --cached
Restore the changes
Git restore
Unstage
staging
WORKSHOP
Demo.c
Git add
Demo.c
Git rm --cached
git
unstaging
Git add
Staging
Demo.c
Sakdsajdasd
Sadasdsad
sadsadsd
Demo.c
Sakdsajdasd
Sadasdsad
sadsadsd

RECAP WITH ME

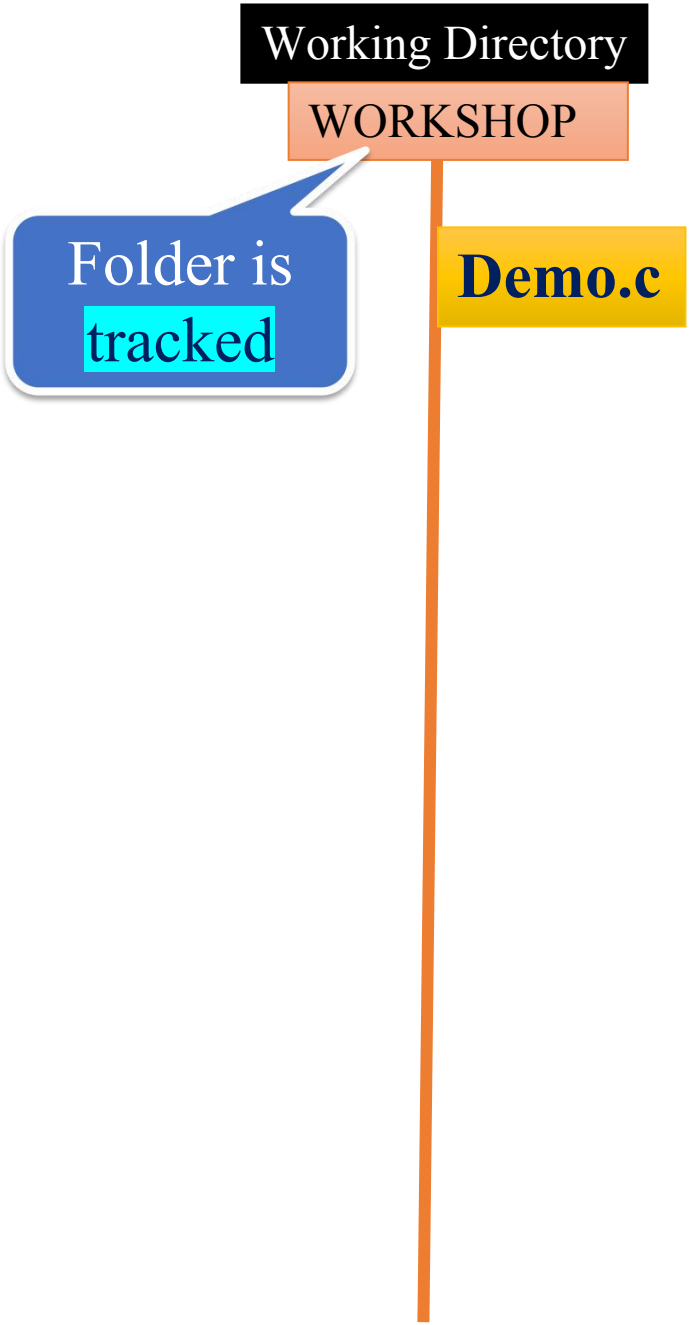

Working Directory
WORKSHOP
Initially:
Folder is Not
tracked

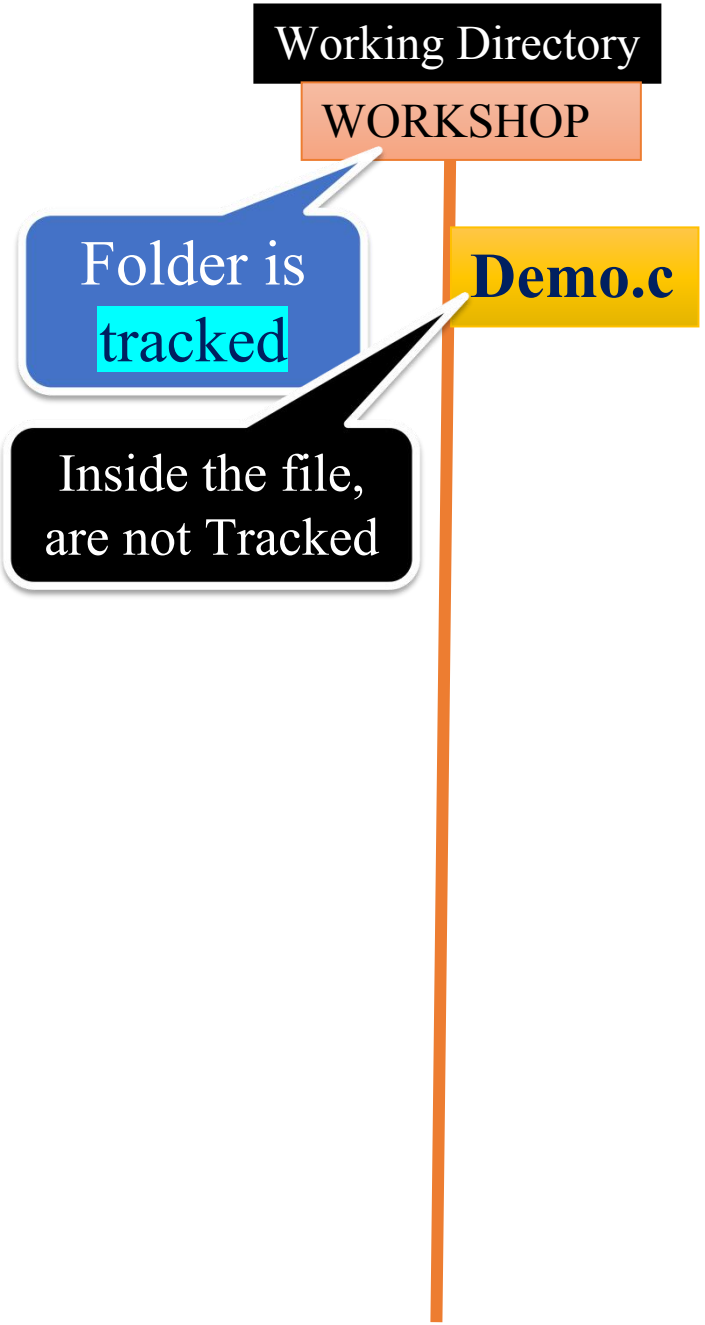

Working Directory
WORKSHOP
Git init

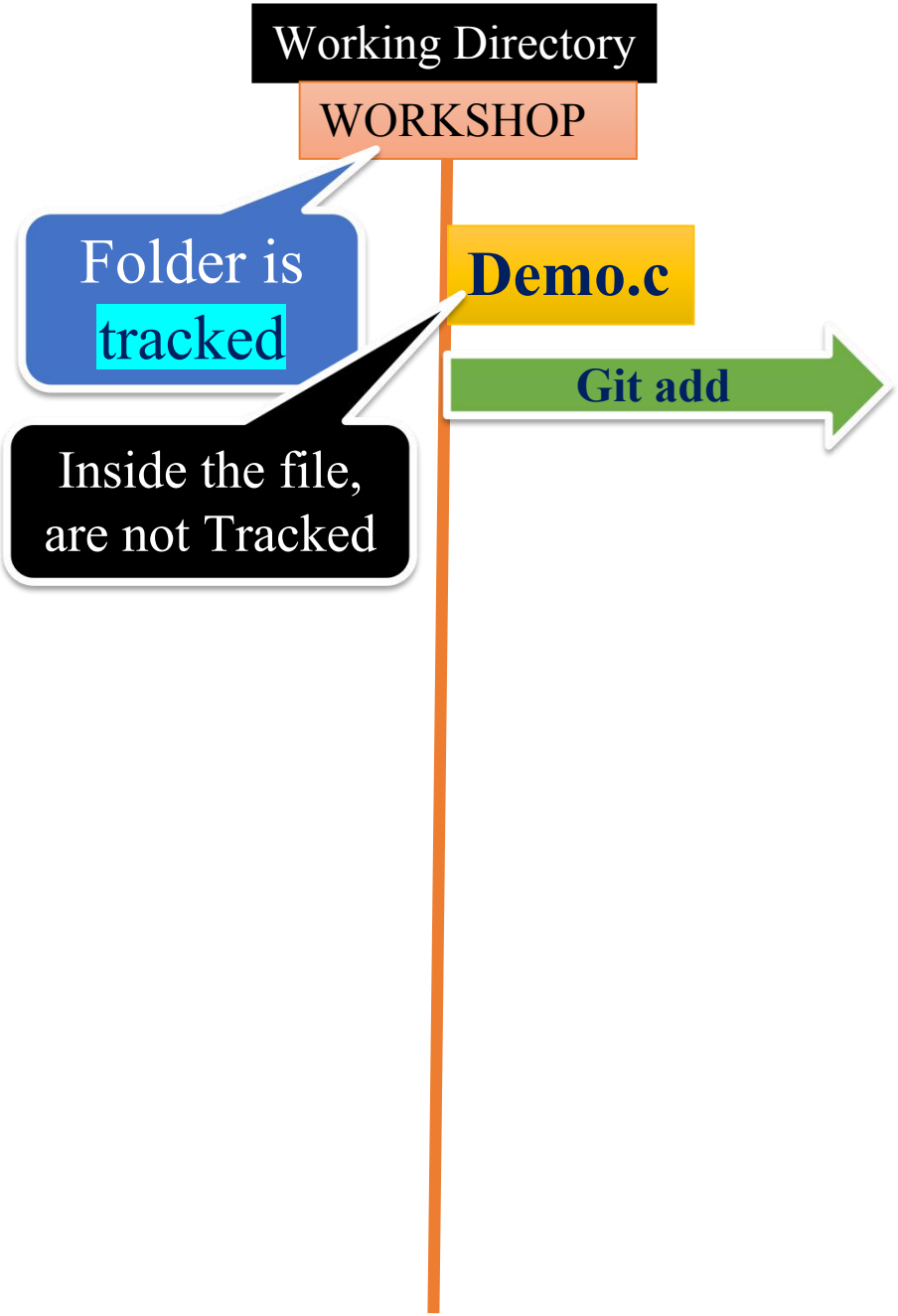

Working Directory
WORKSHOP
Folder is
tracked

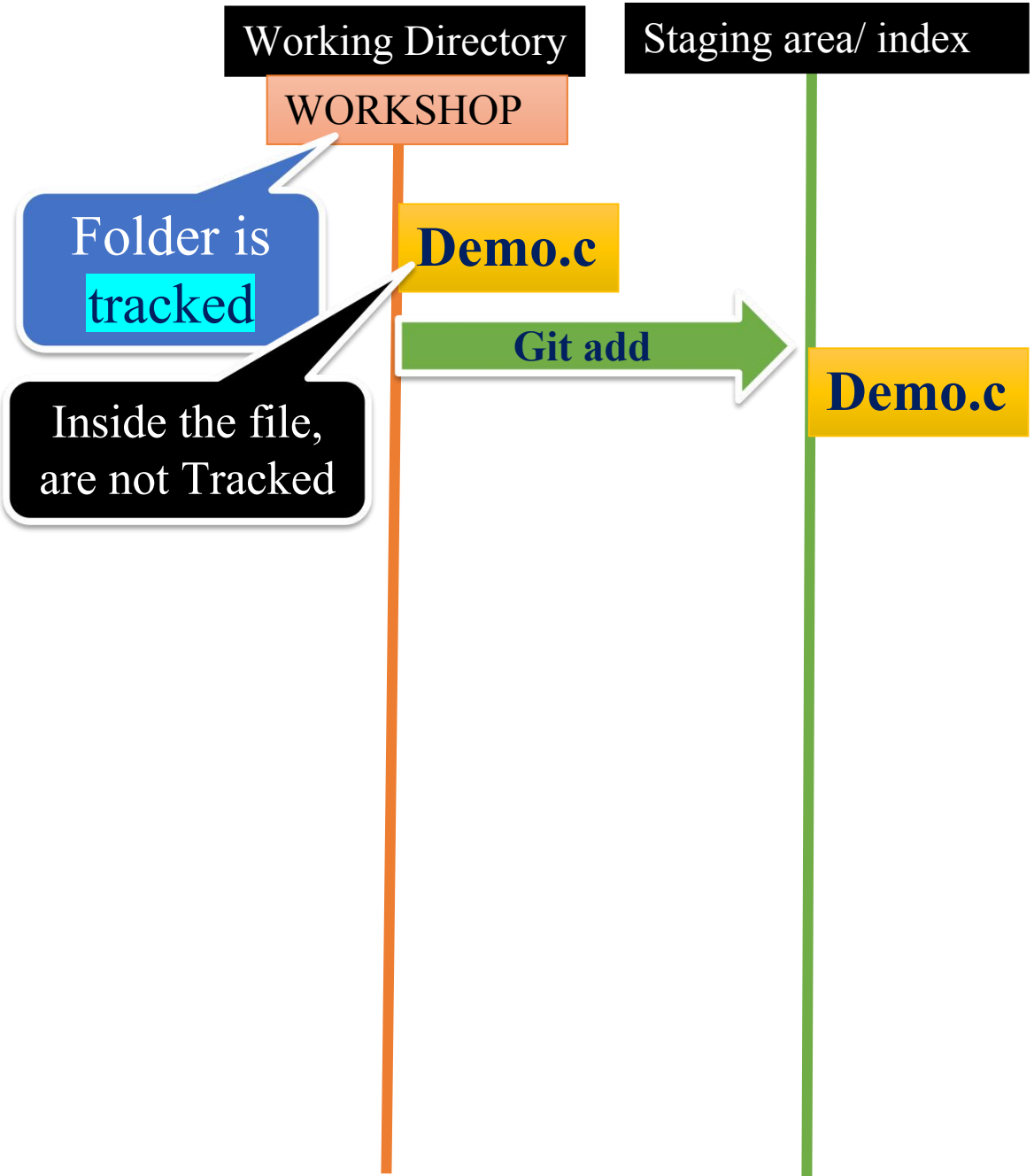


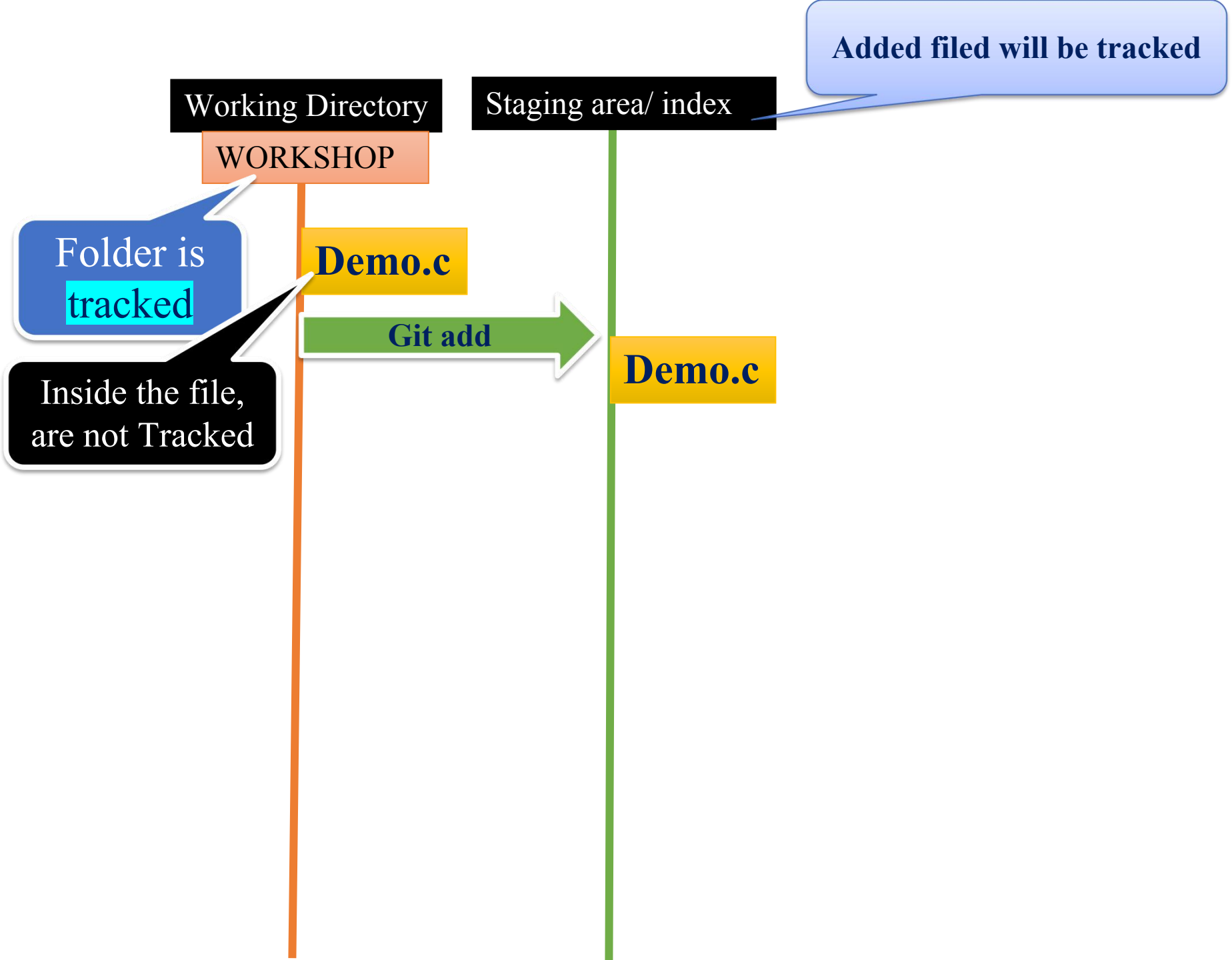
IF YOU CLONE A GITHUB REPO, IT WILL AUTOMATICALLY TRACKED, MEANS IT'S BY DEFAULT INITIALIZEED GIT

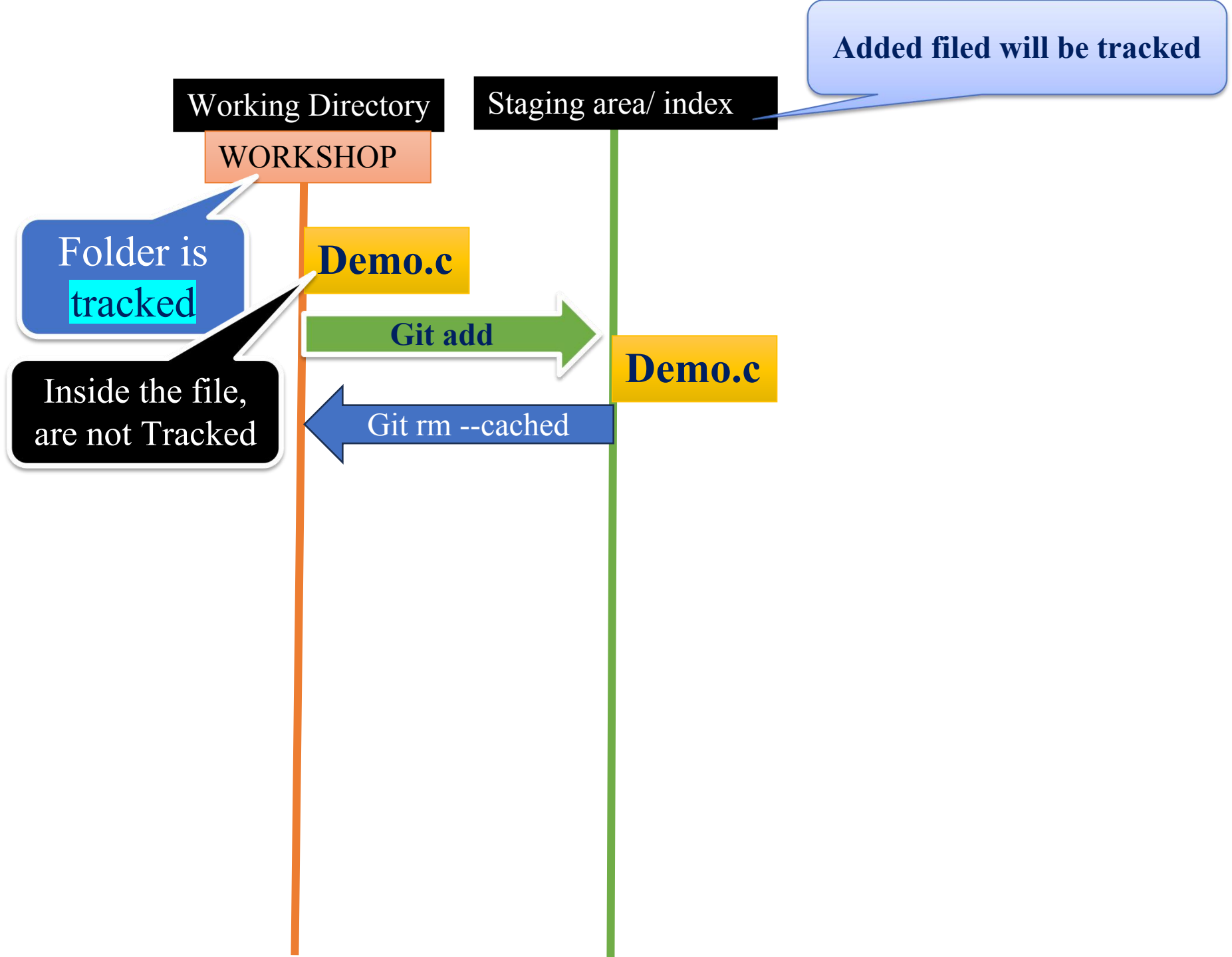

Working Directory
WORKSHOP
Folder is tracked
Demo.c

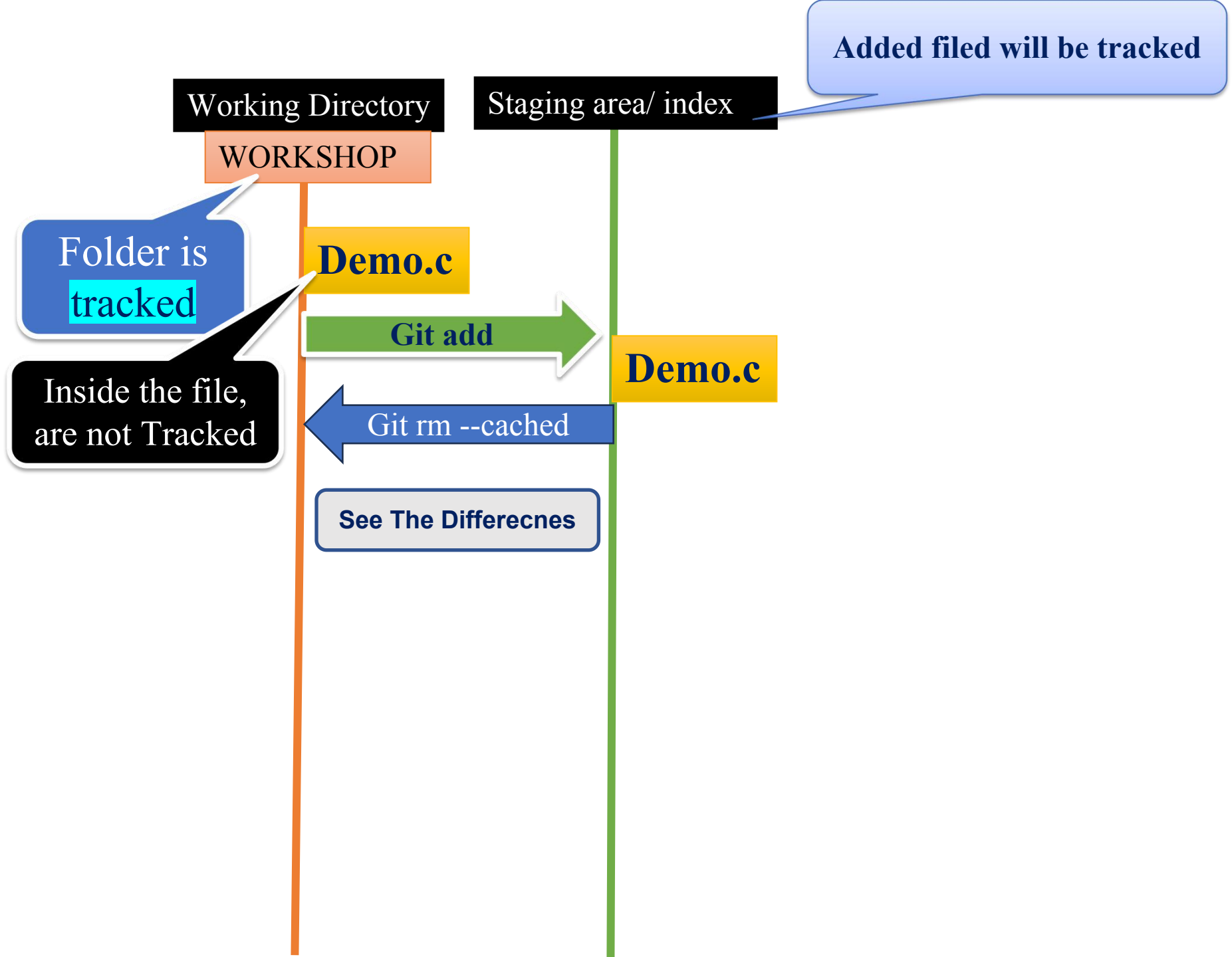

Working Directory
WORKSHOP
Folder is tracked
Demo.c
Inside the file, are not Tracked

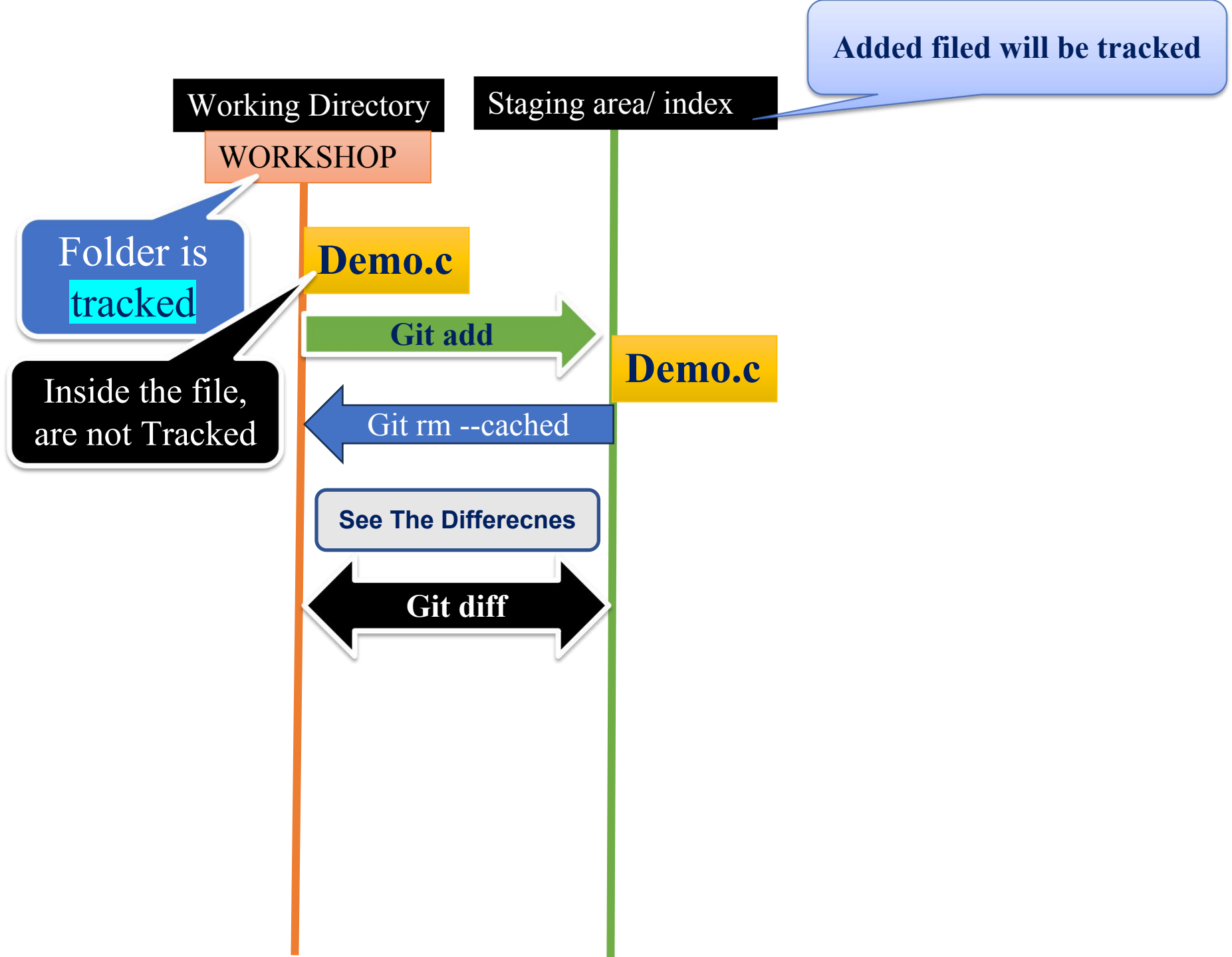

Working Directory
WORKSHOP
Folder is tracked
Demo.c
Git add
Inside the file, are not Tracked

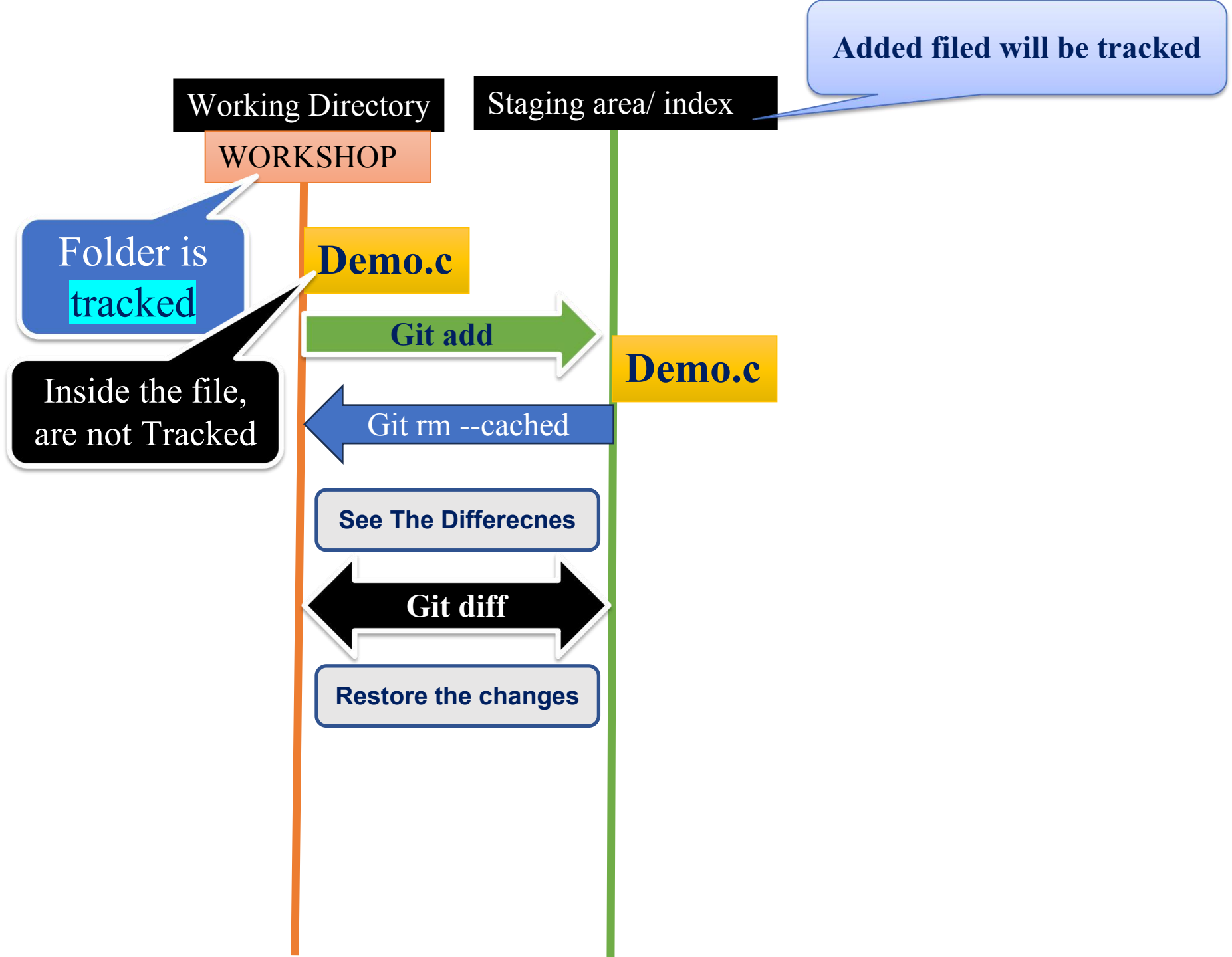

Working Directory
Staging area/ index
WORKSHOP
Folder is tracked
Demo.c
Git add
Demo.c
Inside the file, are not Tracked

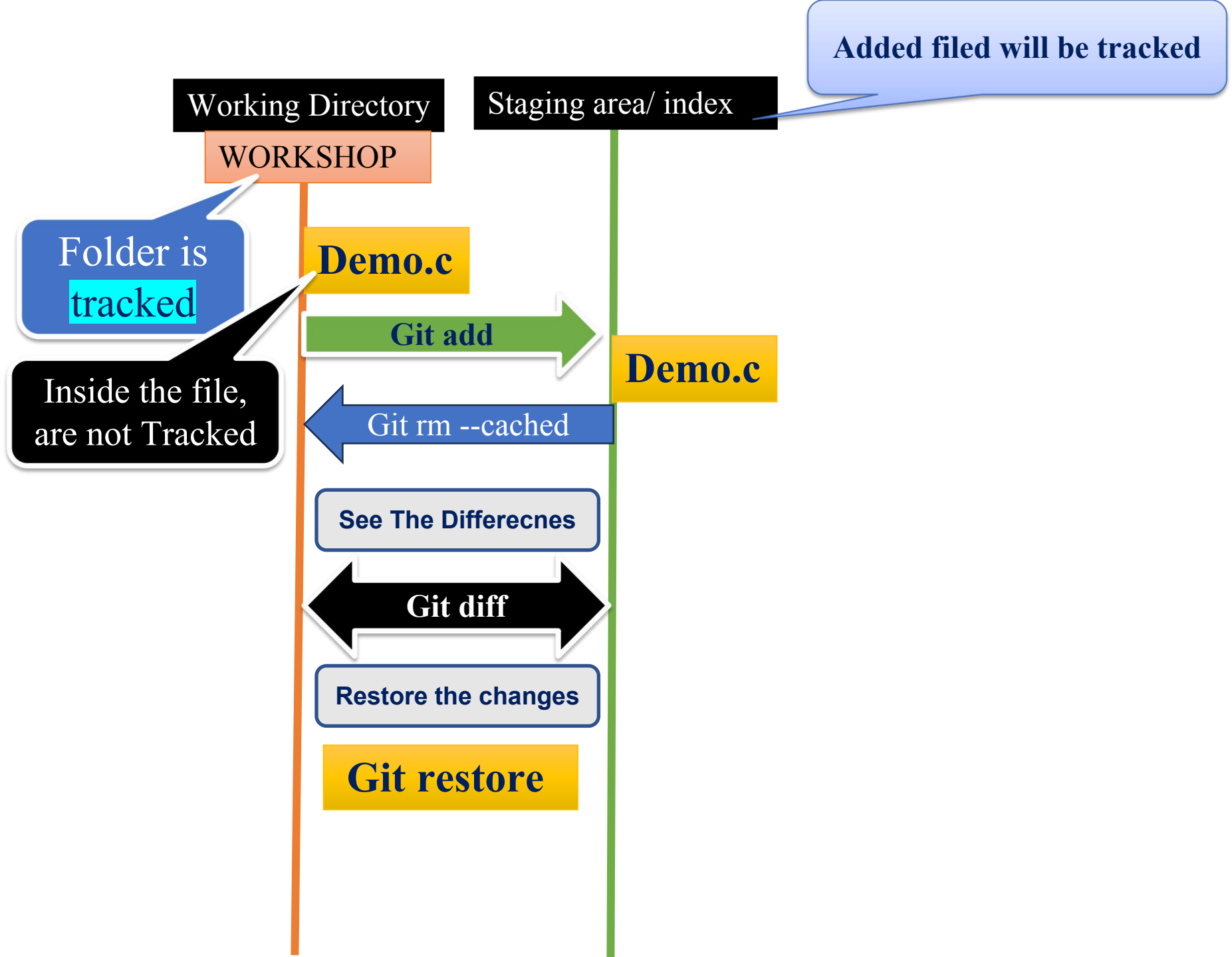

Added filed will be tracked
Working Directory
Staging area/ index
WORKSHOP
Folder is tracked
Demo.c
Git add
Demo.c
Inside the file, are not Tracked

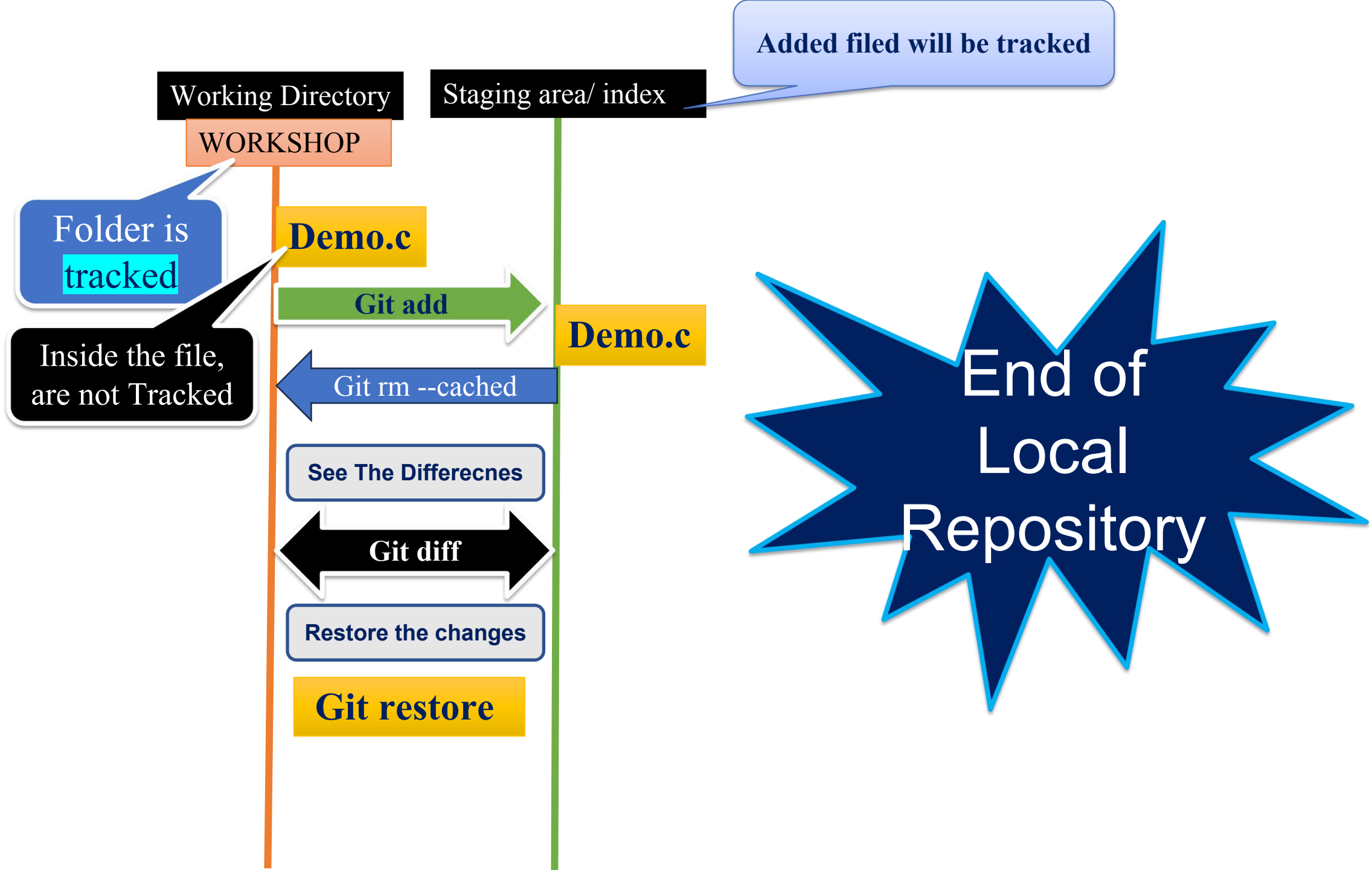

Added filed will be tracked
Working Directory
Staging area/ index
WORKSHOP
Folder is tracked
Demo.c
Git add
Demo.c
Inside the file, are not Tracked
Git rm --cached

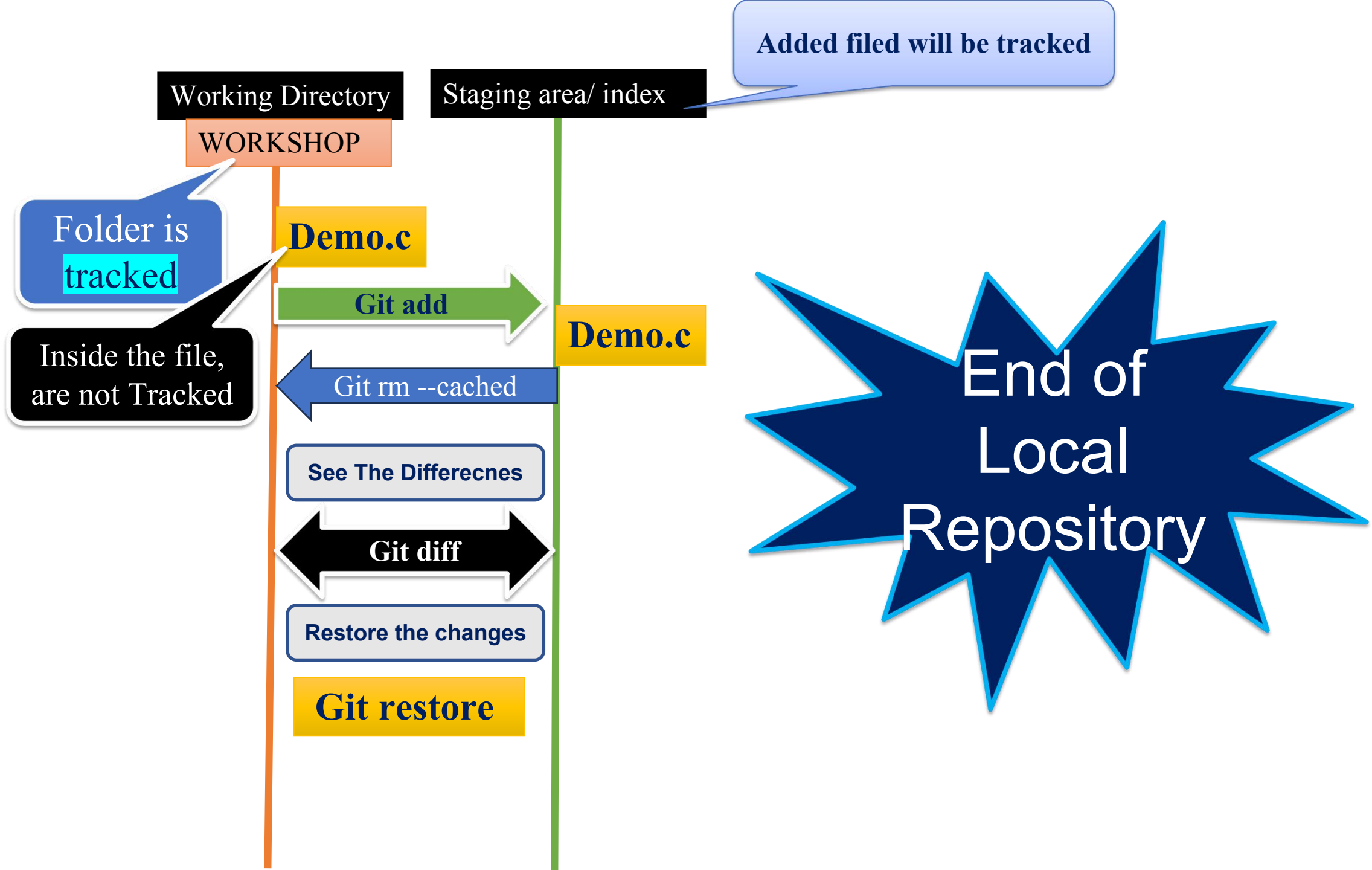

Added filed will be tracked
Working Directory
Staging area/ index
WORKSHOP
Folder is tracked
Demo.c
Git add
Demo.c
Inside the file, are not Tracked
Git rm --cached
See The Differecnes


Added filed will be tracked
Working Directory
Staging area/ index
WORKSHOP
Folder is tracked
Demo.c
Git add
Demo.c
Inside the file, are not Tracked
Git rm --cached
See The Differecnes
Git diff


Added filed will be tracked
Working Directory
Staging area/ index
WORKSHOP
Folder is tracked
Demo.c
Git add
Demo.c
Inside the file, are not Tracked
Git rm --cached
See The Differecnes
Git diff
Restore the changes


Added filed will be tracked
Working Directory
Staging area/ index
WORKSHOP
Folder is tracked
Demo.c
Git add
Demo.c
Inside the file, are not Tracked
Git rm --cached
See The Differecnes
Git diff
Restore the changes
Git restore


Added filed will be tracked
Working Directory
Staging area/ index
WORKSHOP
Folder is tracked
Demo.c
Git add
Demo.c
Inside the file, are not Tracked
Git rm --cached
See The Differecnes
Git diff
Restore the changes
Git restore
End of Local Repository


Added filed will be tracked
Working Directory
Staging area/ index
WORKSHOP
Folder is tracked
Demo.c
Git add
Demo.c
Inside the file, are not Tracked
Git rm --cached
See The Differecnes
Git diff
Restore the changes
Git restore
End of Local Repository

Added filed will be tracked

Working Directory

Staging area/ index

WORKSHOP

Folder is tracked

Demo.c

Git add

Demo.c

Inside the file, are not Tracked

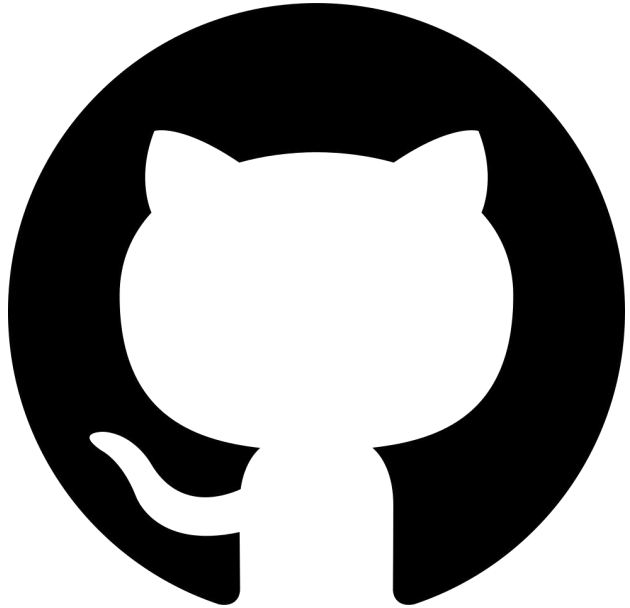
Git rm --cached

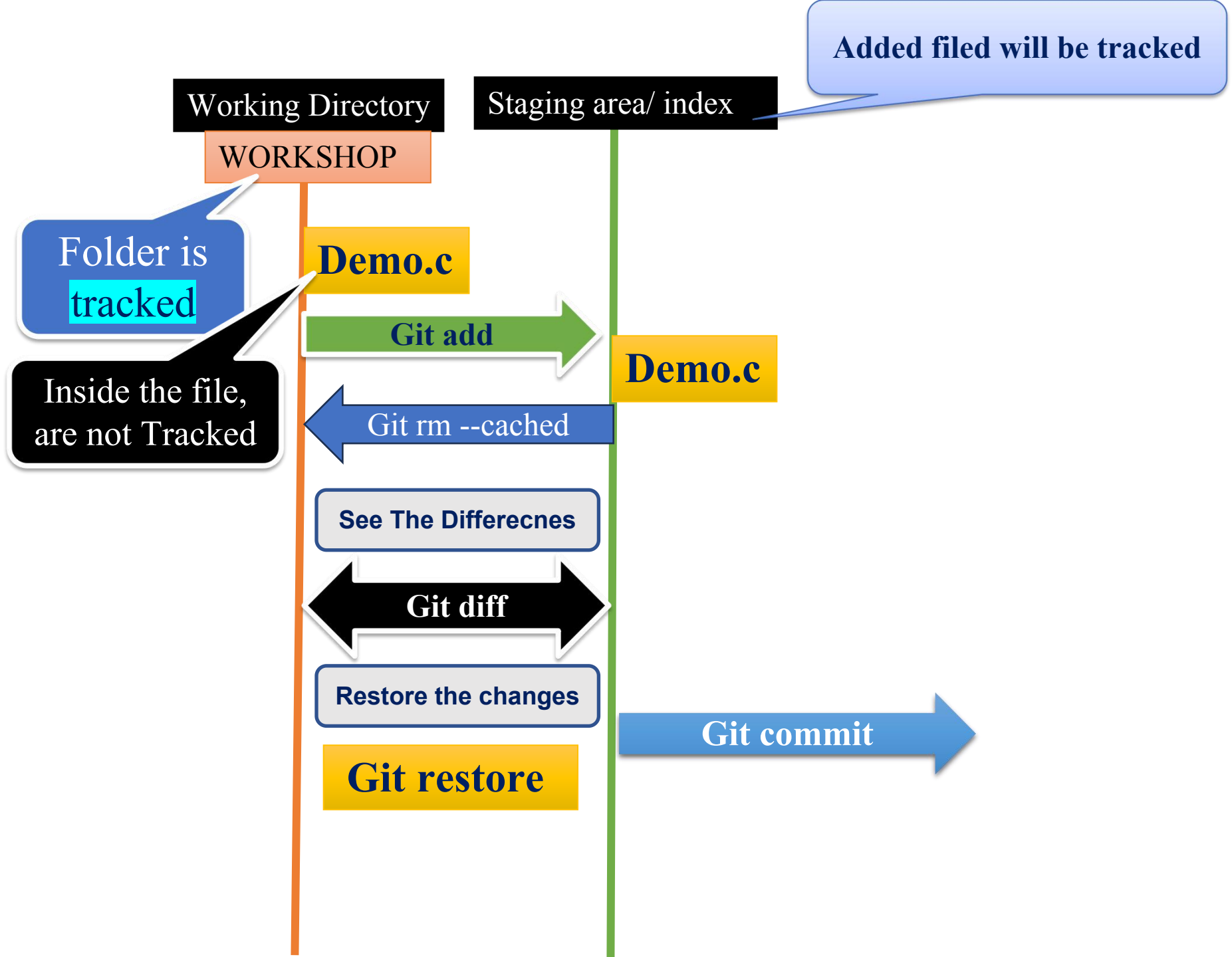
See The Differecnes

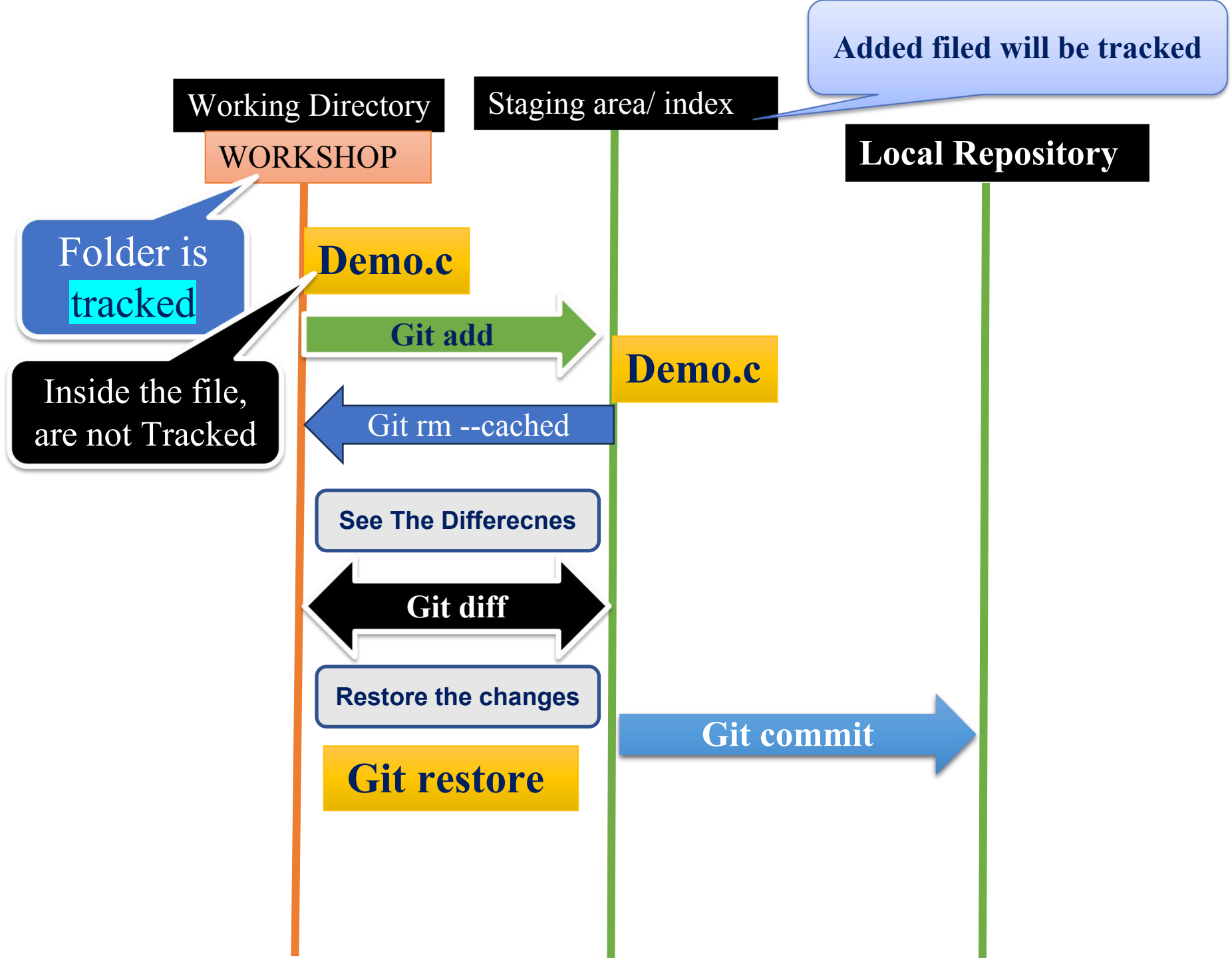
Git diff

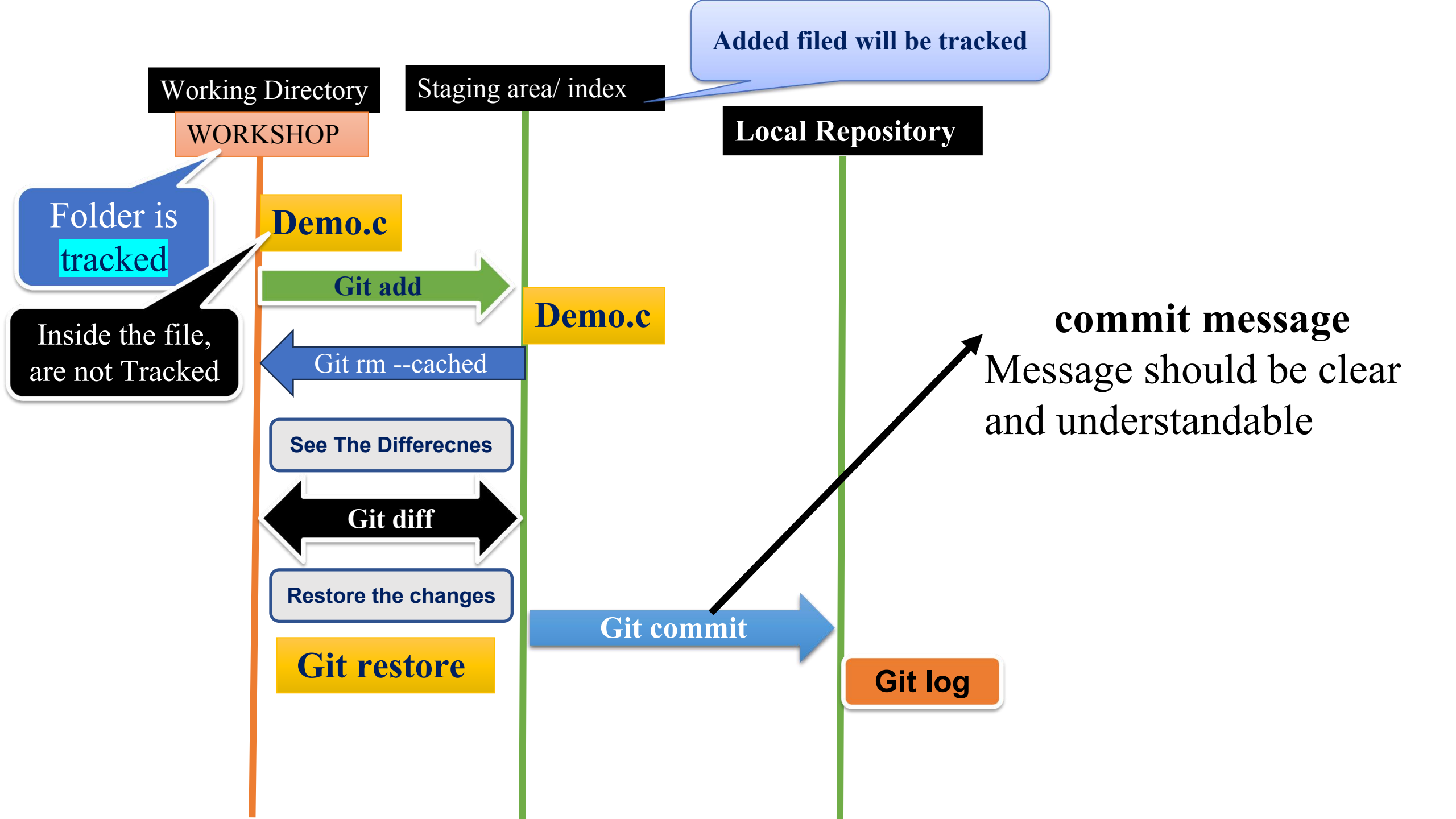
Restore the changes

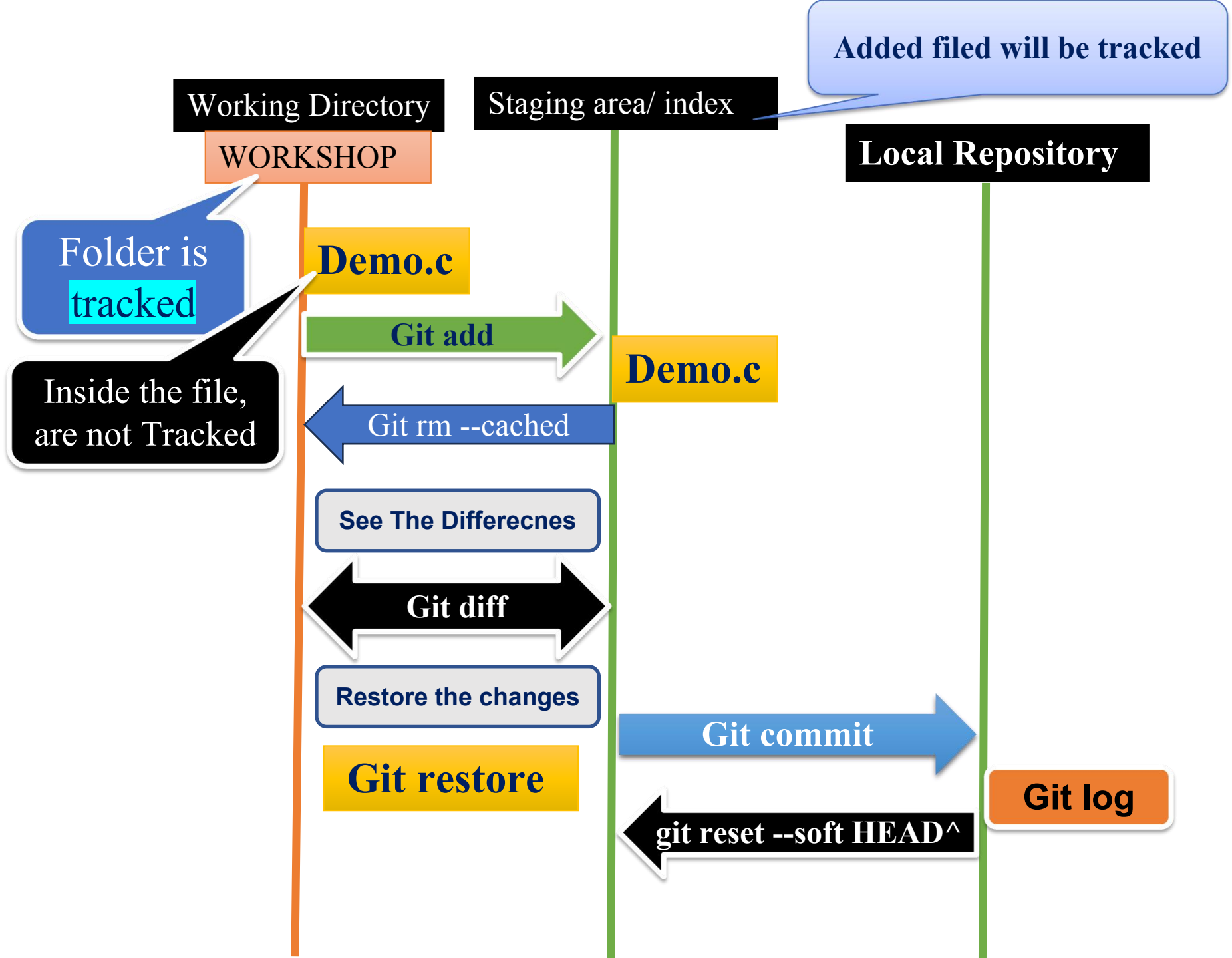
Git restore

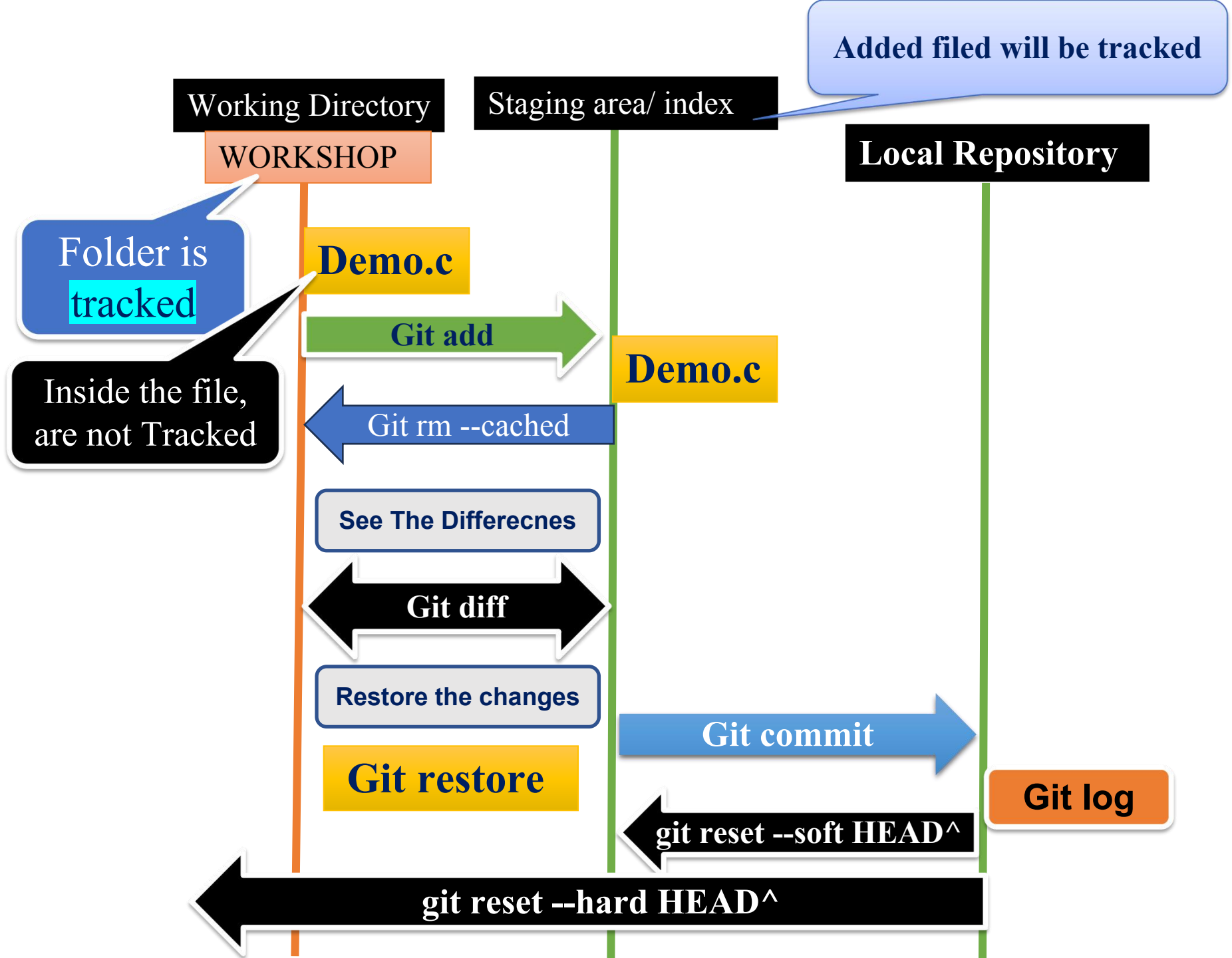


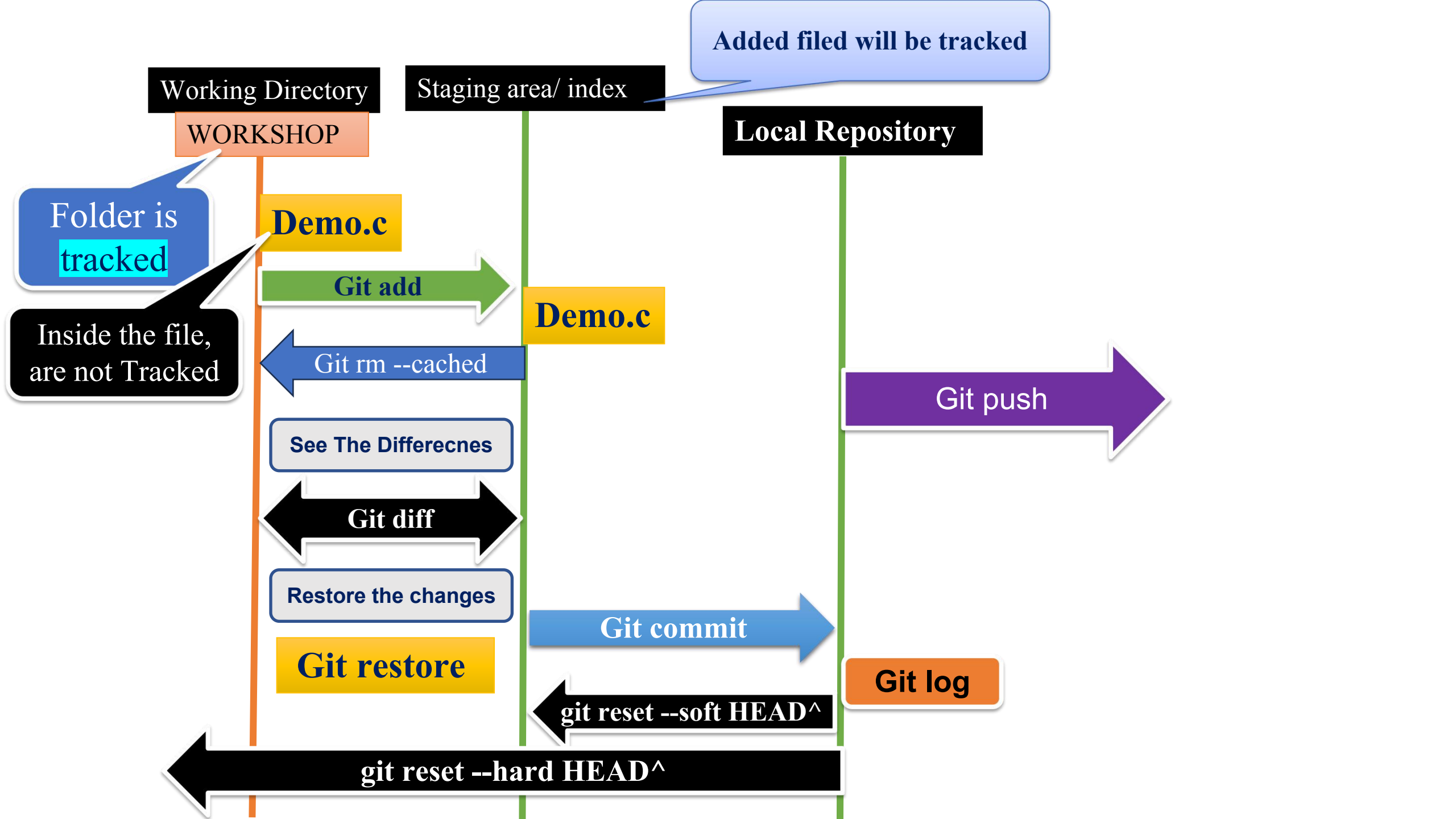

Added filed will be tracked
Working Directory
Staging area/ index
WORKSHOP
Folder is tracked
Demo.c
Git add
Demo.c
Inside the file, are not Tracked
Git rm --cached
See The Differecnes
Git diff
Restore the changes
Git commit
Git restore

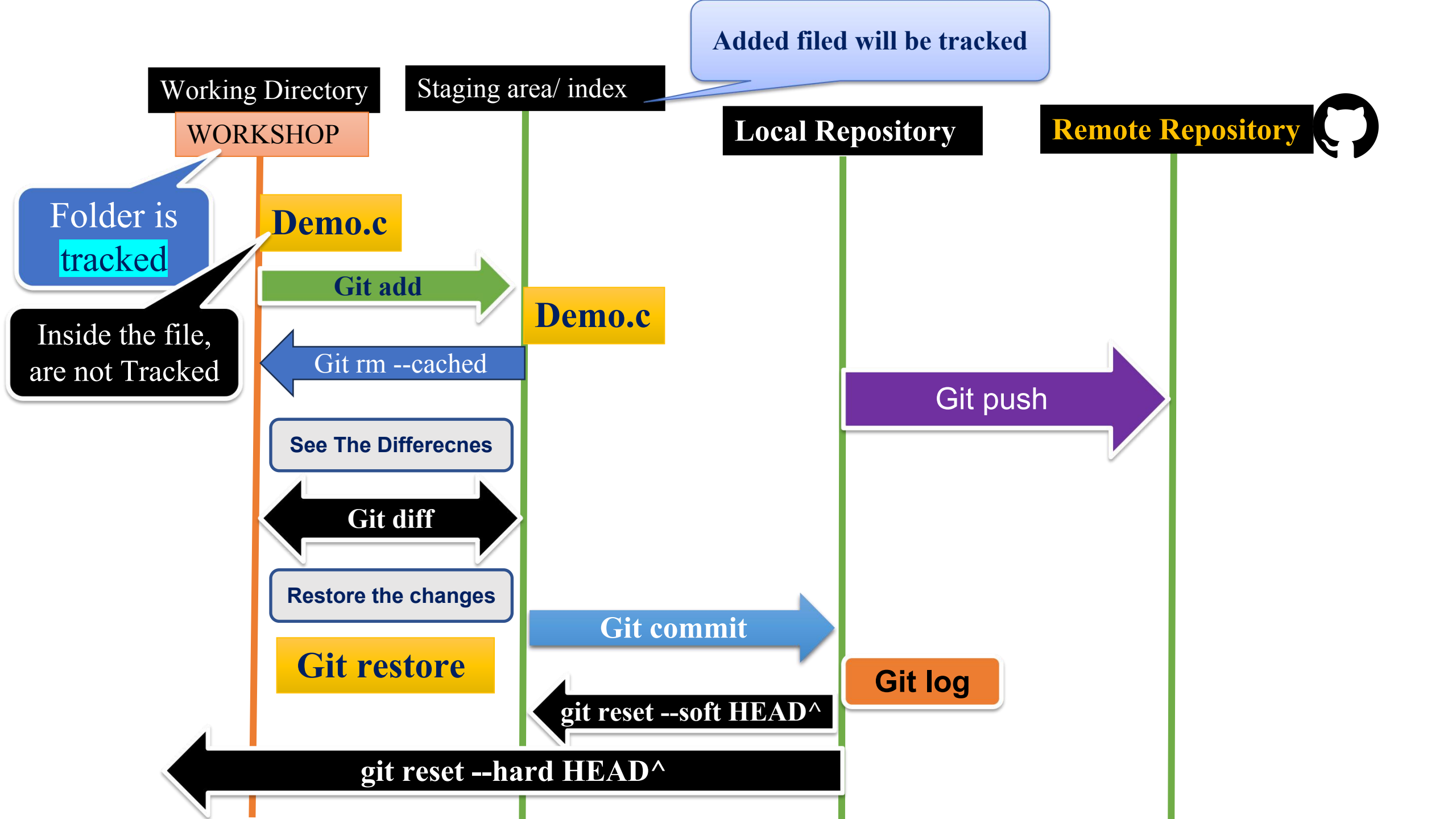

Added filed will be tracked
Working Directory
Staging area/ index
Local Repository
WORKSHOP
Folder is tracked
Demo.c
Git add
Demo.c
Inside the file, are not Tracked
Git rm --cached
See The Differecnes
Git diff
Restore the changes
Git commit
Git restore

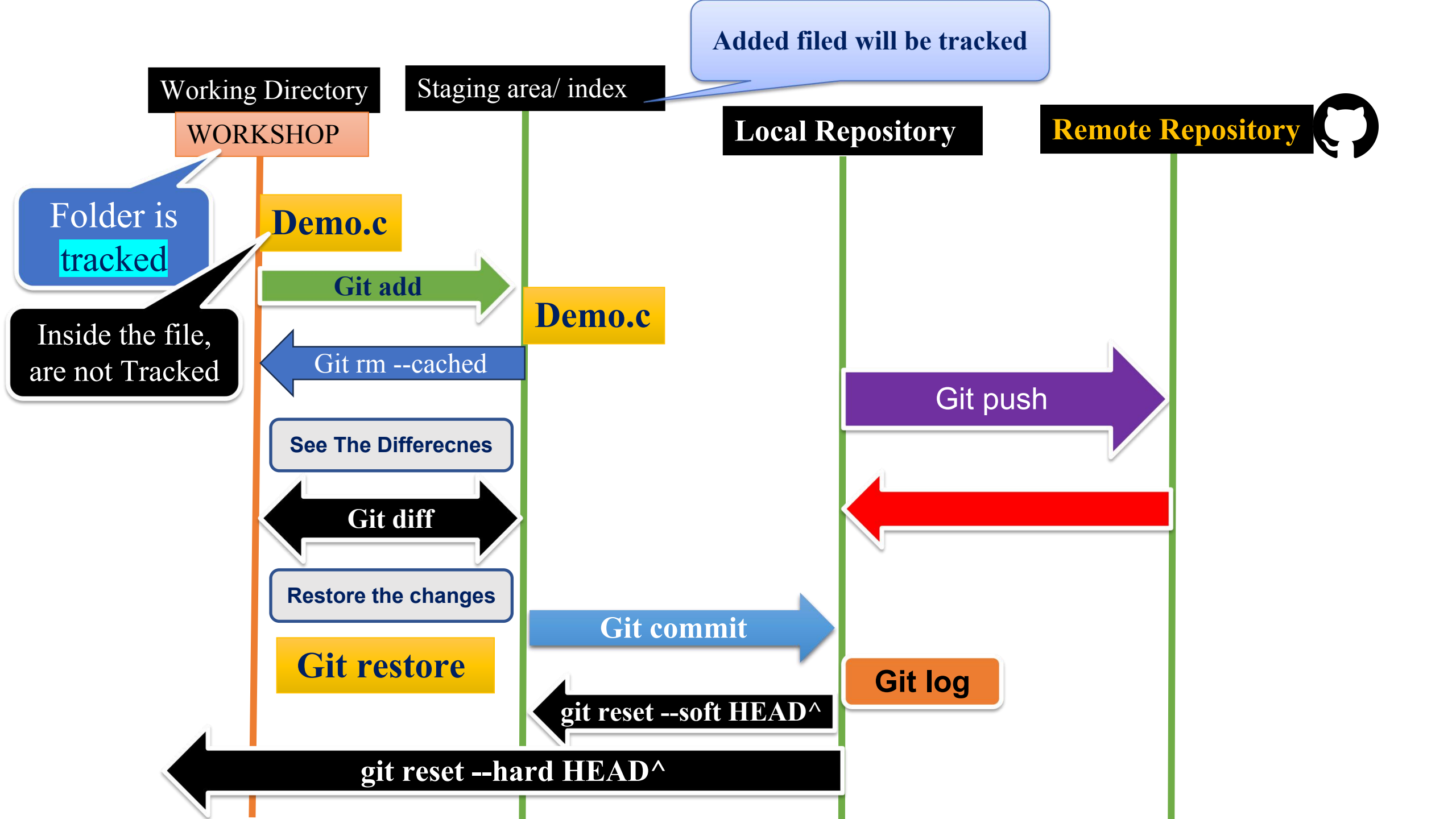

Added filed will be tracked
Working Directory
Staging area/ index
Local Repository
WORKSHOP
Folder is tracked
Demo.c
Git add
Demo.c
Inside the file, are not Tracked
Git rm --cached
See The Differecnes
Git diff
Restore the changes
Git restore
Git commit
Git log
commit message
Message should be clear and understandable

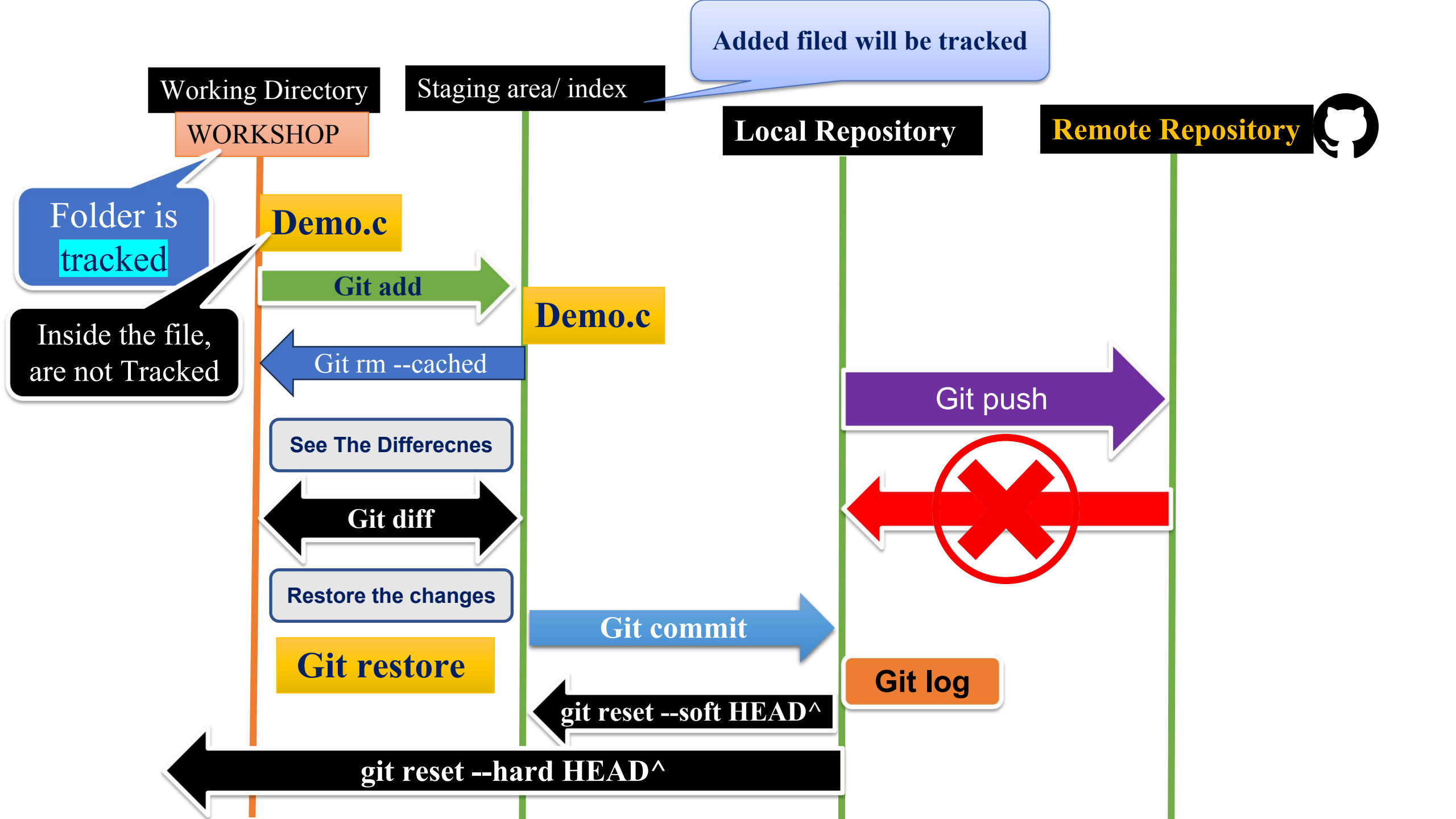

Added filed will be tracked
Working Directory
Staging area/ index
Local Repository
WORKSHOP
Folder is tracked
Demo.c
Git add
Demo.c
Inside the file, are not Tracked
Git rm --cached
See The Differecnes
Git diff
Restore the changes
Git commit
Git restore
Git log
git reset --soft HEAD^


Added filed will be tracked
Working Directory
Staging area/ index
Local Repository
WORKSHOP
Folder is tracked
Demo.c
Git add
Demo.c
Inside the file, are not Tracked
Git rm --cached
See The Differecnes
Git diff
Restore the changes
Git commit
Git restore
Git log
git reset --soft HEAD^
git reset --hard HEAD^


Added filed will be tracked
Working Directory
Staging area/ index
Local Repository
WORKSHOP
Folder is tracked
Demo.c
Git add
Demo.c
Inside the file, are not Tracked
Git rm --cached
Git push
See The Differecnes
Git diff
Restore the changes
Git commit
Git restore
Git log
git reset --soft HEAD^
git reset --hard HEAD^


Added filed will be tracked
Working Directory
Staging area/ index
Local Repository
Remote Repository
WORKSHOP
Folder is tracked
Demo.c
Git add
Demo.c
Inside the file, are not Tracked
Git rm --cached
Git push
See The Differecnes
Git diff
Restore the changes
Git commit
Git restore
Git log
git reset --soft HEAD^
git reset --hard HEAD^


Added filed will be tracked
Working Directory
Staging area/ index
Local Repository
Remote Repository
WORKSHOP
Folder is tracked
Demo.c
Git add
Demo.c
Inside the file, are not Tracked
Git rm --cached
Git push
See The Differecnes
Git diff
Restore the changes
Git commit
Git restore
Git log
git reset --soft HEAD^
git reset --hard HEAD^


Added filed will be tracked
Working Directory
Staging area/ index
Local Repository
Remote Repository
WORKSHOP
Folder is tracked
Demo.c
Git add
Demo.c
Inside the file, are not Tracked
Git rm --cached
Git push
See The Differecnes
Git diff
Restore the changes
Git commit
Git restore
Git log
git reset --soft HEAD^
git reset --hard HEAD^

Master Branch

```
#include <stdio.h>

int main()
{
    int a, b, sum, sub, mu
    scanf("%d",&a);
    scanf("%d",&b);

    sum=a+b;
    sub=a-b;
    mult=a*b;
    div=a/b;
    rem=a%b;

    printf("%d\n",sum);
    printf("%d\n",sub);
    printf("%d\n",mult);
    printf("%d\n",div);
    printf("%d\n",rem);
}
```

Debug Branch

```
#include <stdio.h>

int main()
{
    int a, b, sum, sub, mult, div, rem;
    scanf("%d",&a);
    scanf("%d",&b);

    sum=a+b;
    sub=a-b;
    mult=a*b;
    if(b==0)
    {
    printf("Denominator (b) cannot be zero");
    }
    else
    {
            div=a/b;
    rem=a%b;
    printf("%d\n",div);
    printf("%d\n",rem);
    }
}
```

Prompt Branch

```
#include <stdio.h>
int main()

                                ult, div, rem;
                                 1: ");

                                 2: ");

    rem=a%b;
    printf("Sum = %d\n",sum);
    printf("Sub = %d\n",sub);
    printf("Mult = %d\n",mult);
    printf("Div = %d\n",div);
    printf("Remainder = %d\n",rem);
}
```

Merge