Customization Server

coolOrange s.r.l

Aug 11, 2023

CUSTOMIZATION SERVER

1 Installation **1**
- 1.1 Requirements . . . 1
- 1.2 Setup . . . 1
- 1.3 Updates . . . 1
- 1.4 Uninstall . . . 2

2 Getting started **3**
- 2.1 Publish customizations easily . . . 3
- 2.2 Publish using Git or Visual Studio Code . . . 4
- 2.3 View customizations using Bonobo Git Server . . . 11

3 Change logs **15**
- 3.1 Customization Server v2 . . . 15
- 3.2 Customization Share v1 . . . 15

CHAPTER

ONE

INSTALLATION

1.1 Requirements

The Customization Server should be installed on the same machine as the Autodesk Data Management Server and is fully compatible with its system requirements.

Operating System: 64-bit only

- Microsoft Windows Server 2012
- Microsoft Windows Server 2016
- Microsoft Windows Server 2019
- Microsoft Windows Server 2022

.NET Framework: 4.7 or higher

1.2 Setup

The following installer automates the necessary steps for installing the latest version of Bonobo Git Server on the IIS and creates the necessary Git repository for Client Customizations to be distributed to all workstations.

Download: `coolOrange Customization Server`

The setup is delivered as an executable and accepts the standard windows installer arguments documented here.
To accept the products EULA when starting the setup in silent mode pass the **ACCEPT_EULA=1** argument.

1.3 Updates

To install a newer version of the product just execute the setup file of the new version.
This will keep your customizations in the existing repositories.

1.4 Uninstall

In case you want to remove the product from your Autodesk Data Management Server you can:

- Execute the setup file again. This will give you the option to repair or remove the Customization Server. Click on "Remove" to uninstall the program.
- Go to "Control Panel - Programs and Features", find "coolOrange Customization Server" and run "Uninstall".

Back up customizations

All the currently published customizations are also removed and are then only available on synchronized workstations. Therefore, please make sure whether you want to **back up** your data in `C:\inetpub\wwwroot\coolOrange\customizations\App_Data` beforehand.

CHAPTER

TWO

GETTING STARTED

Once the CustomizationServer is installed on the **Autodesk Data Management Server**, the system is ready to distribute customizations effortlessly.
No additional configuration is required to initiate the process.

2.1 Publish customizations easily

Your files only need to be uploaded to its repository so that they will be automatically distributed to all workstations connected to the Vault Server.
Regardless of whether the machines are running or turned off, or if new environments are added over time, the customizations are then synchronized every time a Vault application is launched.
This ensures that all Vault applications in your organization always use up-to-date customizations.

Currently, the distribution exclusively supports the distribution of customizations for Vault Client and CAD applications.
These include:

- powerEvents workflows, guardrails, automations, tabs and other enhancements
- powerJobs Client standard scripts
- powerGate ERP Integrations

The corresponding products must be installed in order to publish.

Limitations

Please note that distribution is limited to the customizations and products mentioned above.
Customizations for other products are not yet supported for automatic distribution.

All their customization files are located in the directory `C:\ProgramData\coolOrange\Client Customizations`.

After they have been **tested** on your environment, the easiest way to **publish** them is to use the Publish-Customizations script.
It requires no Git installation or knowledge and all the steps described below are set up for you.

2.2 Publish using Git or Visual Studio Code

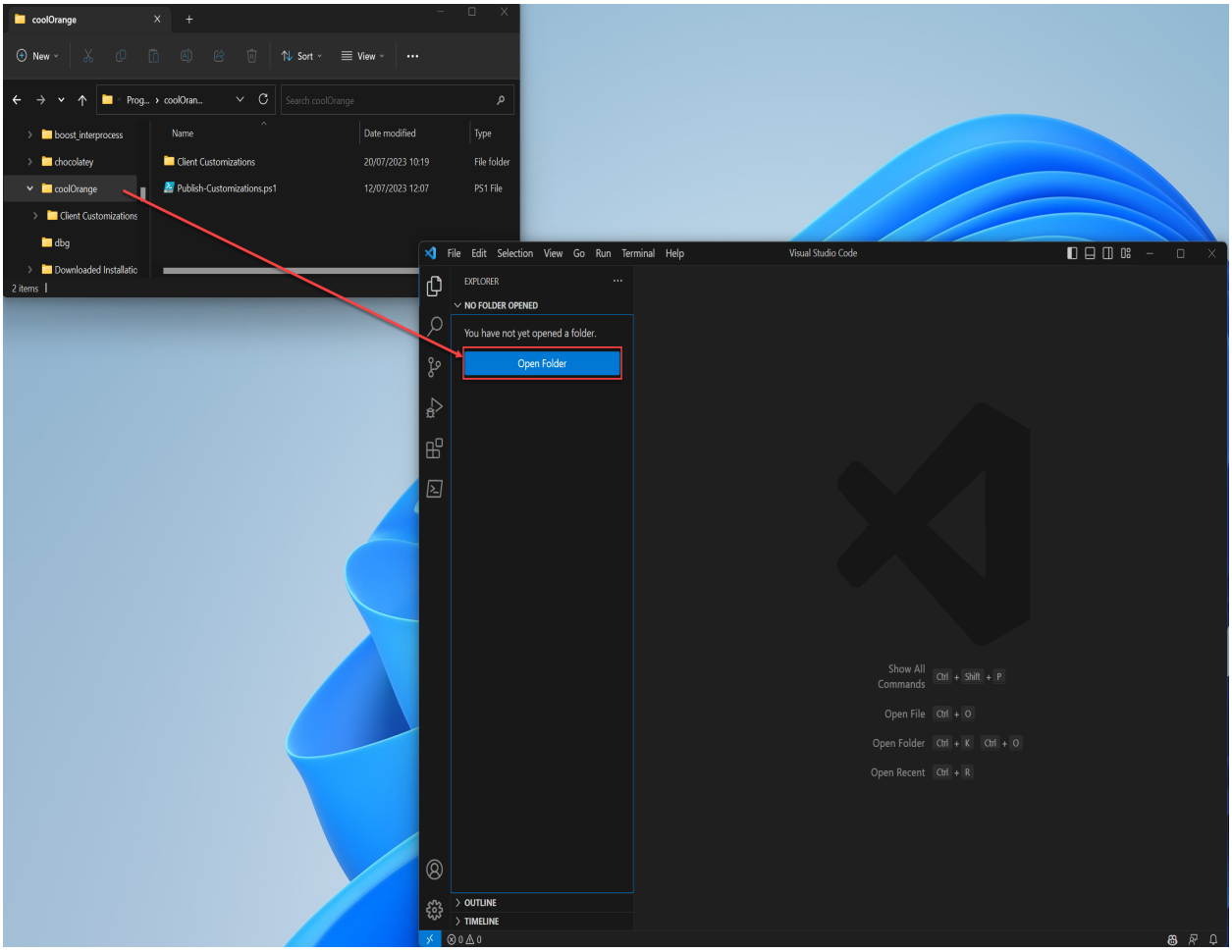
Git experts already have a Git client installed on their development environment.
Especially when several team members work on the same customizations on different computers and publis fixes or enhancements, tools like Visual Studio Code can be very helpful in resolving possible conflicts.

To publish the local files which are already located in the `C:\ProgramData\coolOrange` directory, after installing the CustomizationServer on the Vault Server either:

- run the Publish-Customizations script **for the very first publication** (recommended)
- or restart the Vault application and logon to your Vault Server

In both situations, the local directory is automatically linked to the Git repository http://{Your_Vault_Server}/customizations/coolOrange.git.
From this point on, an additional **remote** was set up in the local git repository, which now points to the respective Vault Server.

In the following example, our `C:\ProgramData\coolOrange` directory is even already managed in another Git repository (cloned from GitHub).



To see a list of all remotes (e.g. if multiple ADMS servers are used) simply use the following command:

```
git remote -v
```

(continues on next page)

(continued from previous page)

```
#origin  https://github.com/coolOrangeLabs/powerGateErpIntegration.git (fetch)
#origin  https://github.com/coolOrangeLabs/powerGateErpIntegration.git (push)
#localhost       http://localhost/customizations/coolOrange.git (fetch)        <-- New
#localhost       http://localhost/customizations/coolOrange.git (push)         <-- New
```

Update Master branch

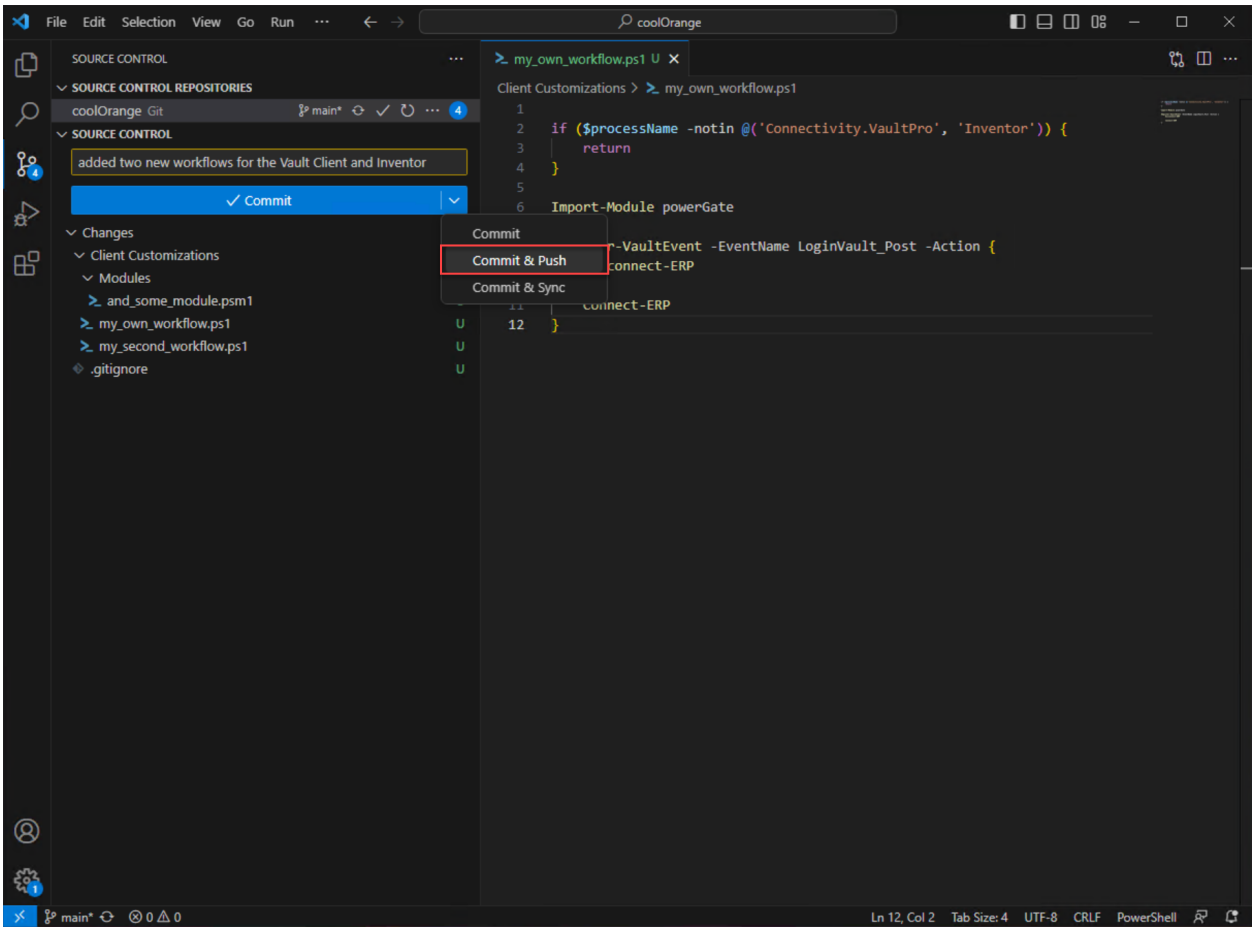
Please note that we will work on a "master" branch (not main). You may want to rename the *main* branch to *master* using this command:

```
git branch -m master
```

All customization files in the local *master* branch are committed with a meaningful commit message and pushed to GitHub (our *origin*):

```
git add "Client Customizations"
git commit -m "added two new workflows for the Vault Client and Inventor"
git push origin master
```

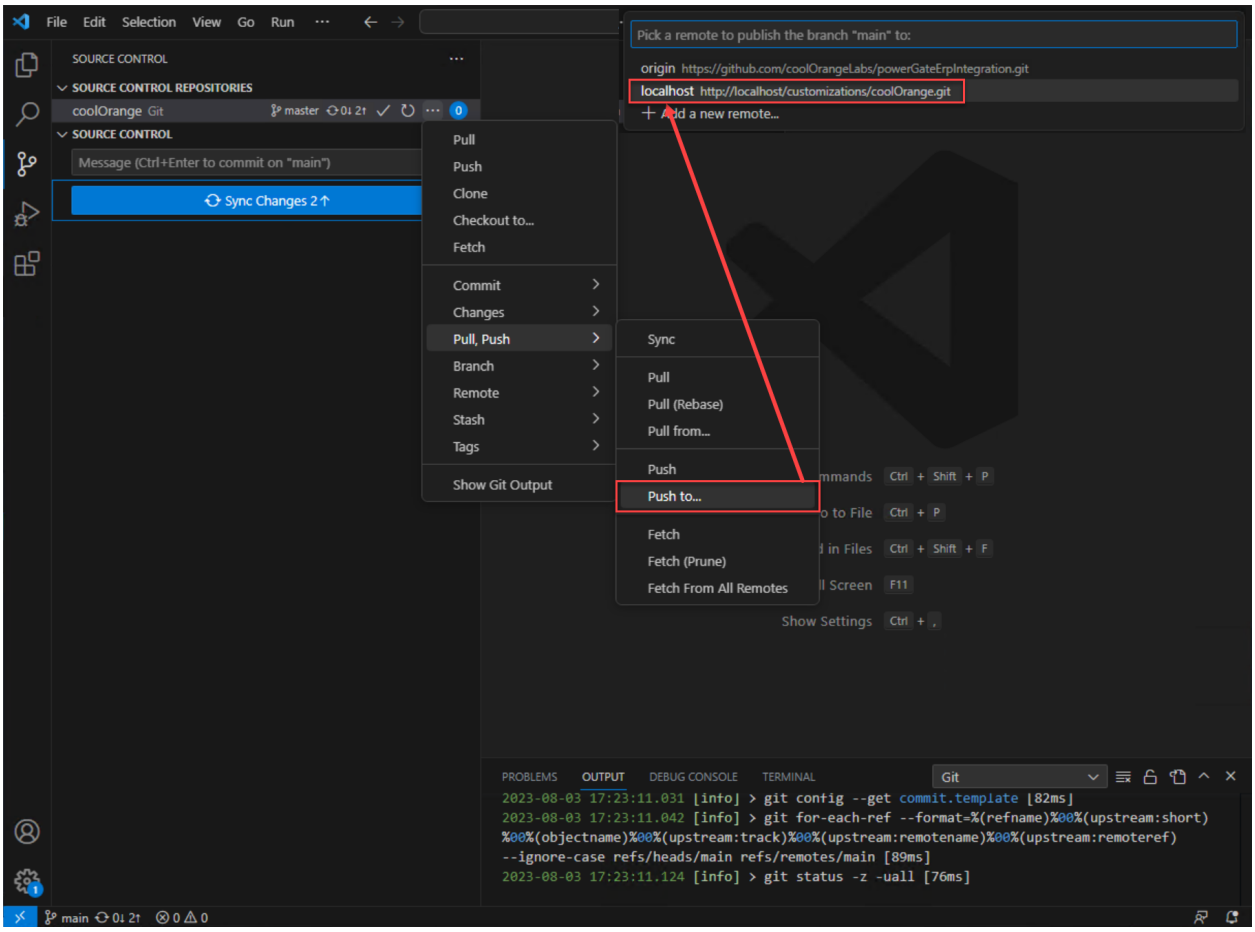
In Visual Studio Code a commit looks like this, followed by a click on the *Commit & Push* button:



After several commits, when our customizations are in a finished state that has been successfully tested, we want to

distribute them to all workstations.
For the go-live we now only need to **push** all commits to the Vault Server:

```
git push {Your_Vault_Server} master
```



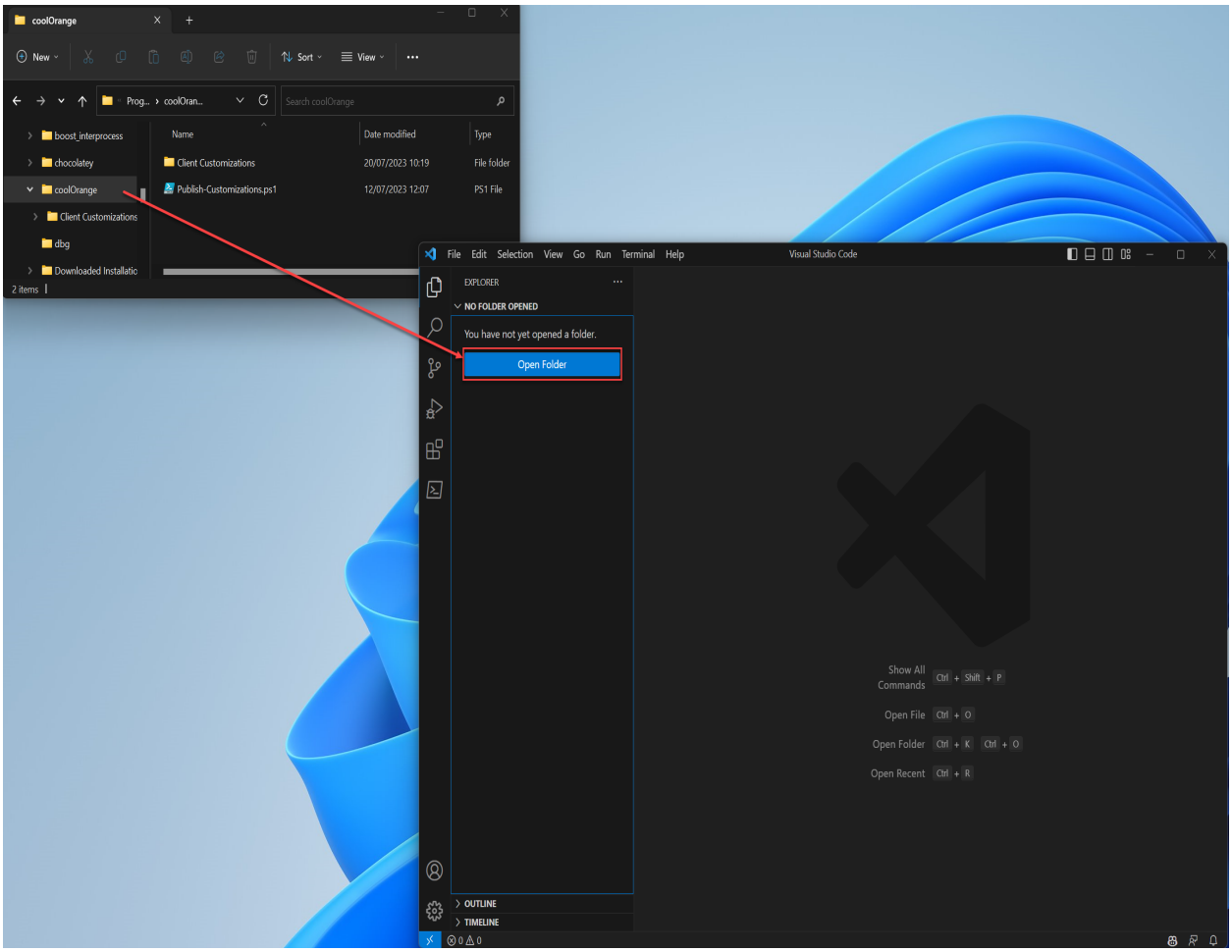
Your customizations can now be used productively on all workstation environments which are part of this Vault site.

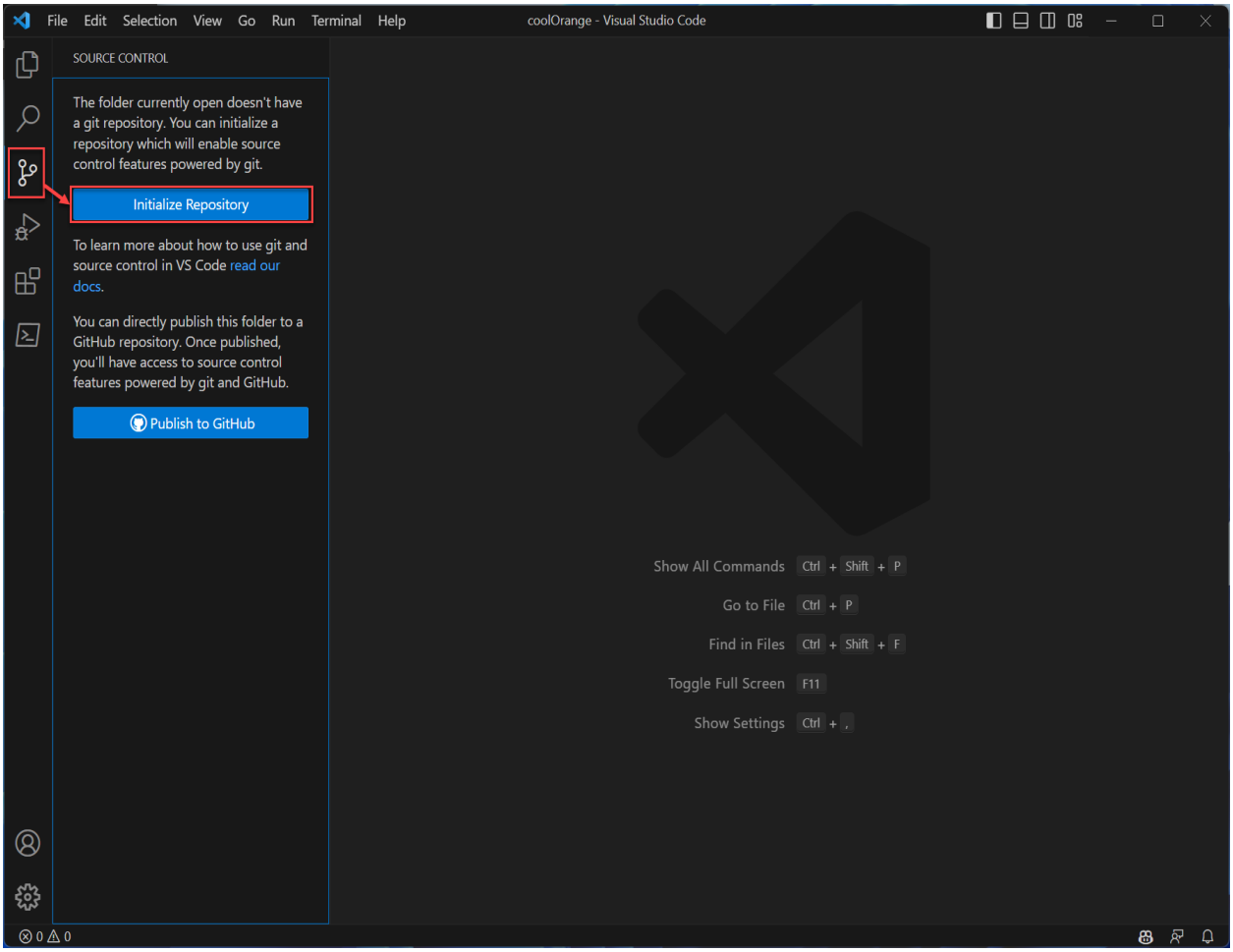
Git experts: Steps to configure the local git repository manually (if no other git system is used)

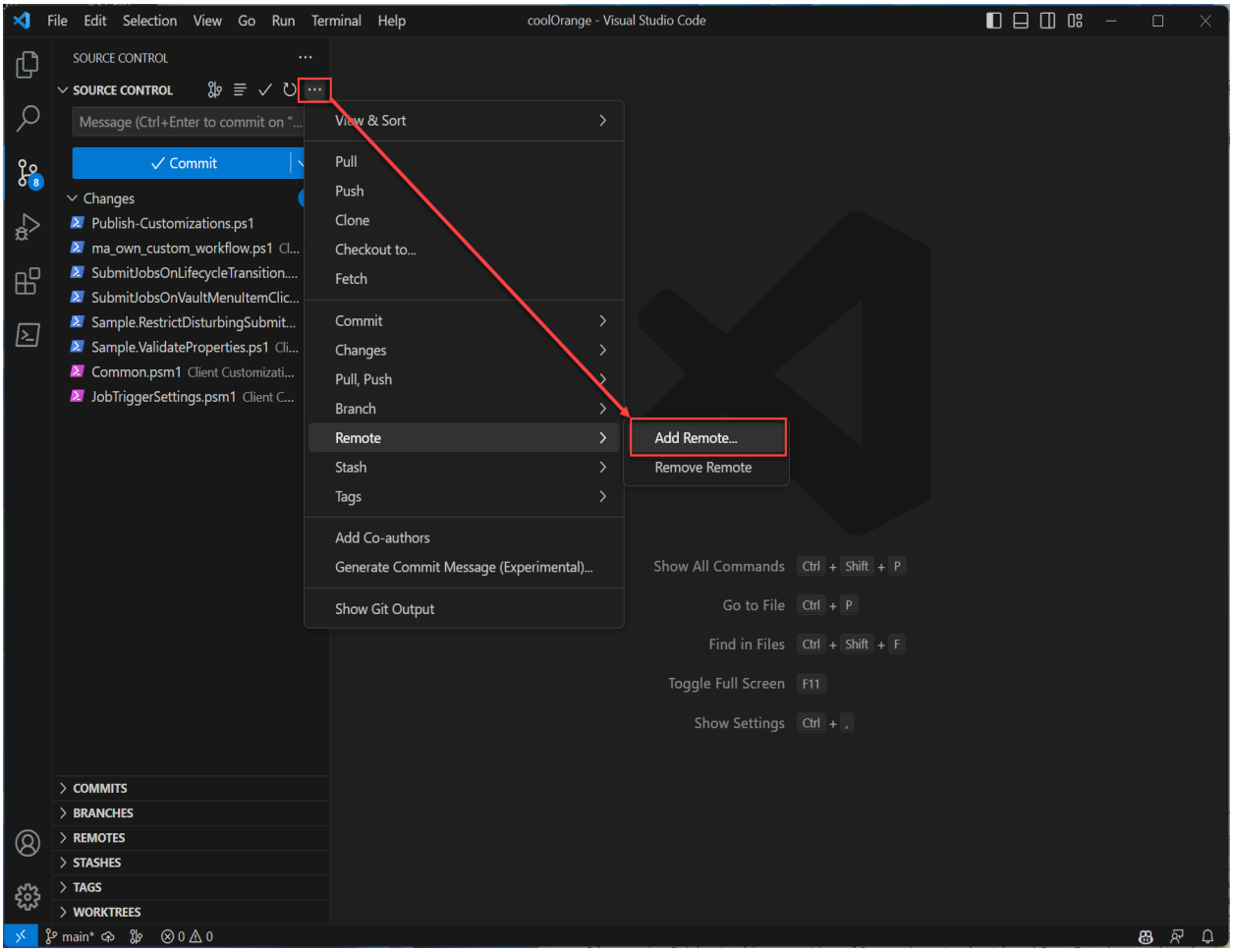
The repository coolOrange.git can not simply be cloned into a non-empty `C:\ProgramData\coolOrange\` directory. However it is possible to **initialize** the current directory as local repository in the same way as the *Publish-Customizations* script does.
First add the repository coolOrange.git as **remote** with the name of your Vault Server:

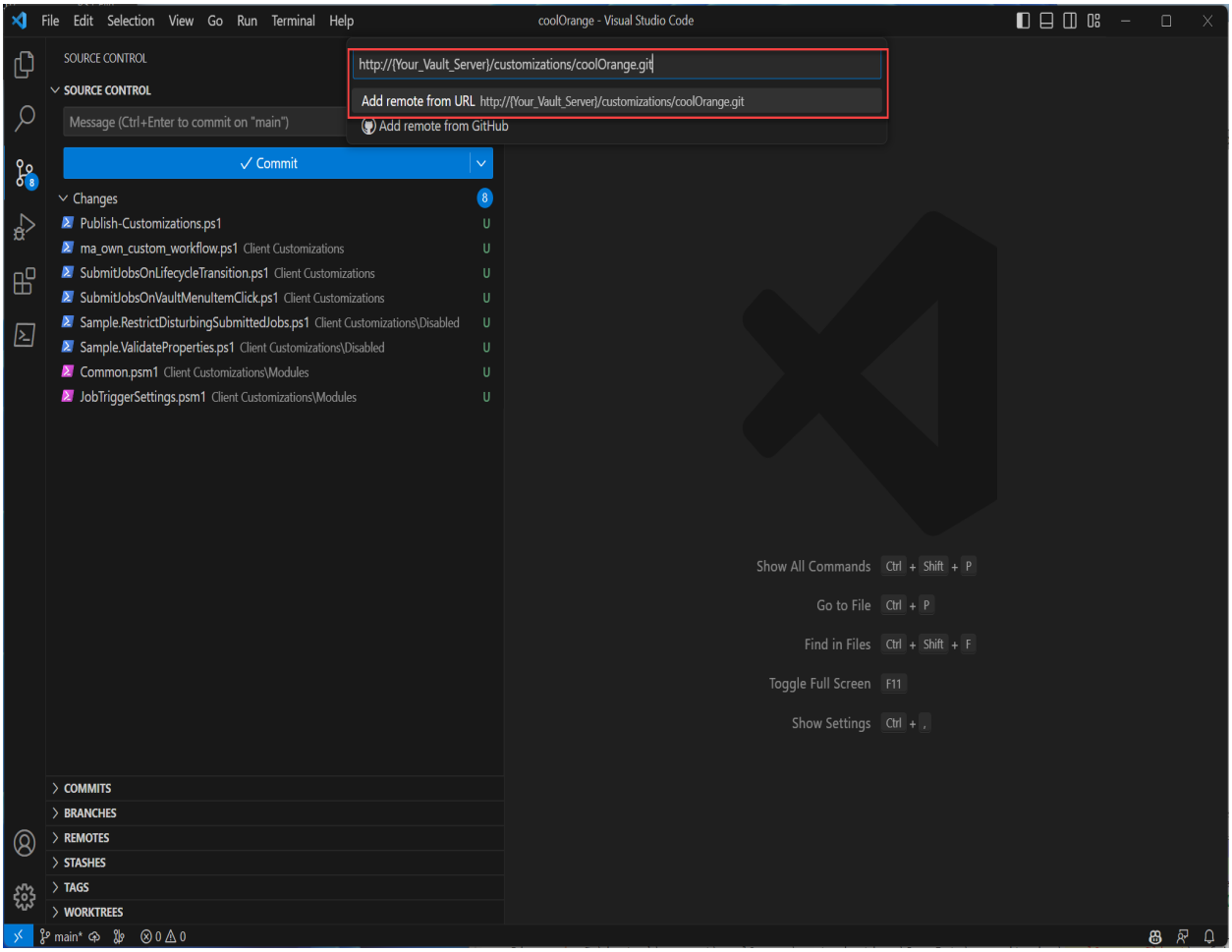
```
cd 'C:\ProgramData\coolOrange\'
git init
git remote add {Your_Vault_Server} http://{Your_Vault_Server}/customizations/coolOrange.
↪git
```

These steps can also be performed in development environments like Visual Studio Code:


coolOrange
New
Sort
View
Prog... › coolOran...
Search coolOrange
boost_interprocess
chocolatey
coolOrange
Client Customizations
dbg
Downloaded Installatic
2 items
Name
Date modified
Type
Client Customizations
20/07/2023 10:19
File folder
Publish-Customizations.ps1
12/07/2023 12:07
PS1 File
File Edit Selection View Go Run Terminal Help
Visual Studio Code
EXPLORER
NO FOLDER OPENED
You have not yet opened a folder.
Open Folder
Show All Commands Ctrl + Shift + P
Open File Ctrl + O
Open Folder Ctrl + K Ctrl + O
Open Recent Ctrl + R
OUTLINE
TIMELINE


File Edit Selection View Go Run Terminal Help
coolOrange - Visual Studio Code
SOURCE CONTROL
The folder currently open doesn't have a git repository. You can initialize a repository which will enable source control features powered by git.
Initialize Repository
To learn more about how to use git and source control in VS Code read our docs.
You can directly publish this folder to a GitHub repository. Once published, you'll have access to source control features powered by git and GitHub.
Publish to GitHub
Show All Commands Ctrl + Shift + P
Go to File Ctrl + P
Find in Files Ctrl + Shift + F
Toggle Full Screen F11
Show Settings Ctrl + ,
0 0


File Edit Selection View Go Run Terminal Help
coolOrange - Visual Studio Code
SOURCE CONTROL
Message (Ctrl+Enter to commit on "...
Commit
Changes
Publish-Customizations.ps1
ma_own_custom_workflow.ps1
SubmitJobsOnLifecycleTransition....
SubmitJobsOnVaultMenuItemClic...
Sample.RestrictDisturbingSubmit...
Sample.ValidateProperties.ps1
Common.psm1 Client Customizati...
JobTriggerSettings.psm1 Client C...
View & Sort
Pull
Push
Clone
Checkout to...
Fetch
Commit
Changes
Pull, Push
Branch
Remote
Stash
Tags
Add Co-authors
Generate Commit Message (Experimental)...
Show Git Output
Add Remote...
Remove Remote
Show All Commands Ctrl + Shift + P
Go to File Ctrl + P
Find in Files Ctrl + Shift + F
Toggle Full Screen F11
Show Settings Ctrl + ,
COMMITS
BRANCHES
REMOTES
STASHES
TAGS
WORKTREES
main*



Please be sure to give your remote the name of your Vault Server!

Then create a local *master* branch for which the previous remote is set up as the primary remote. This enables simple Pull & Push operations to the Vault Server repository.

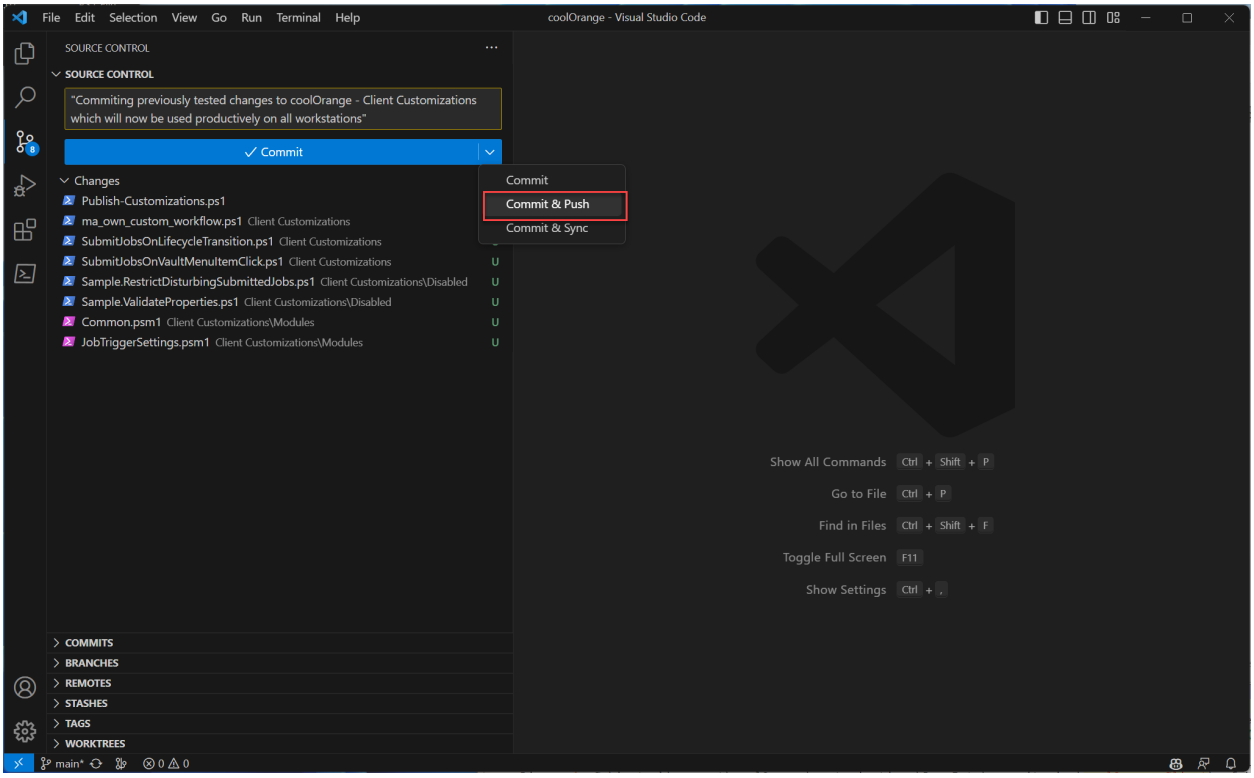
Set your Git username and email address if no global credentials are stored yet:

```
git config user.name "Your Name"
git config user.email "youremail@yourdomain.com"
```

All customization files in the local *master* branch are committed with a meaningful commit message.
Afterwards this commit can directly be pushed to the coolOrange.git repository:

```
git add "Client Customizations"
git commit -m "..."
git push
```

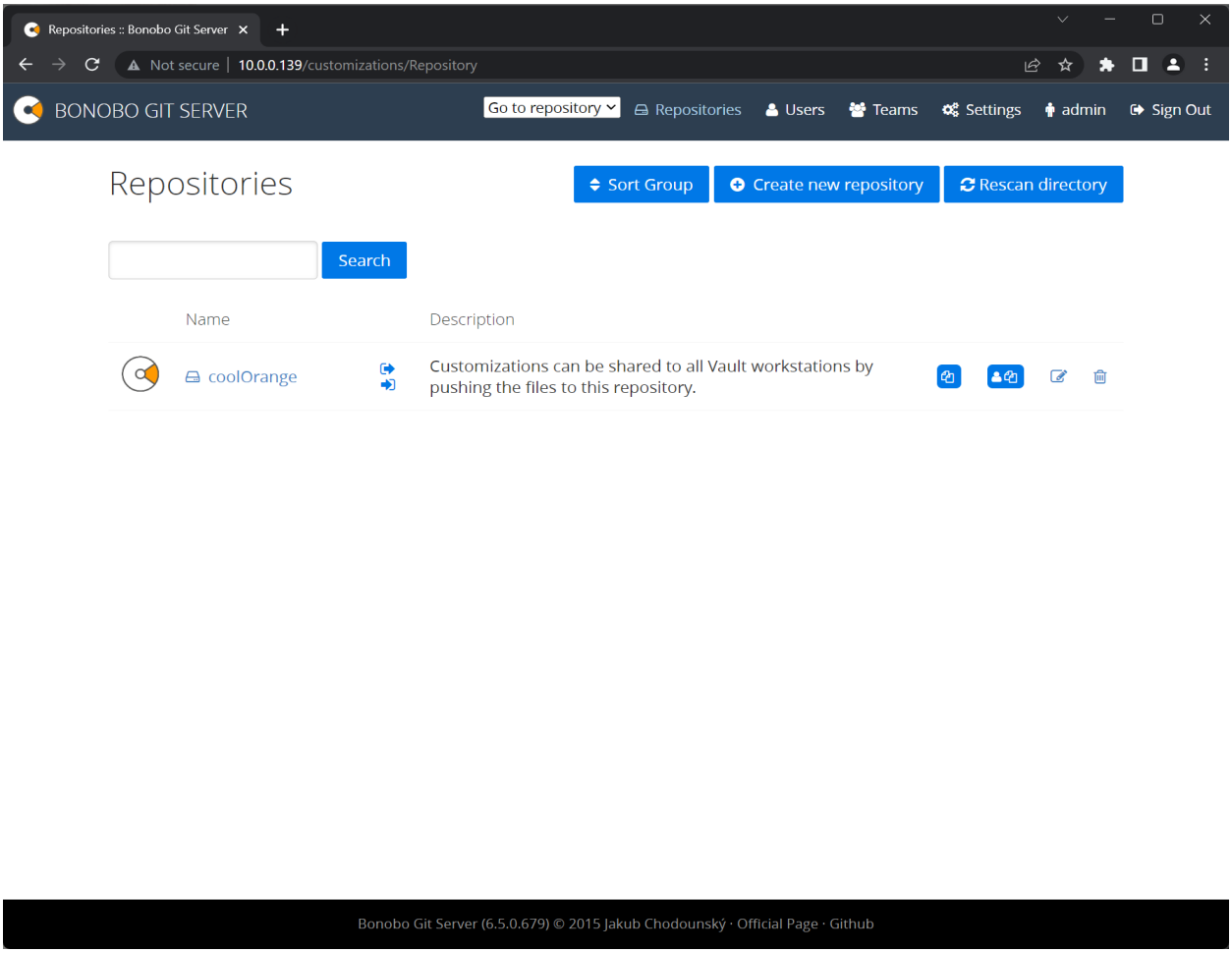
In Visual Studio Code a commit looks like this, followed by a click on the *Commit & Push* button:

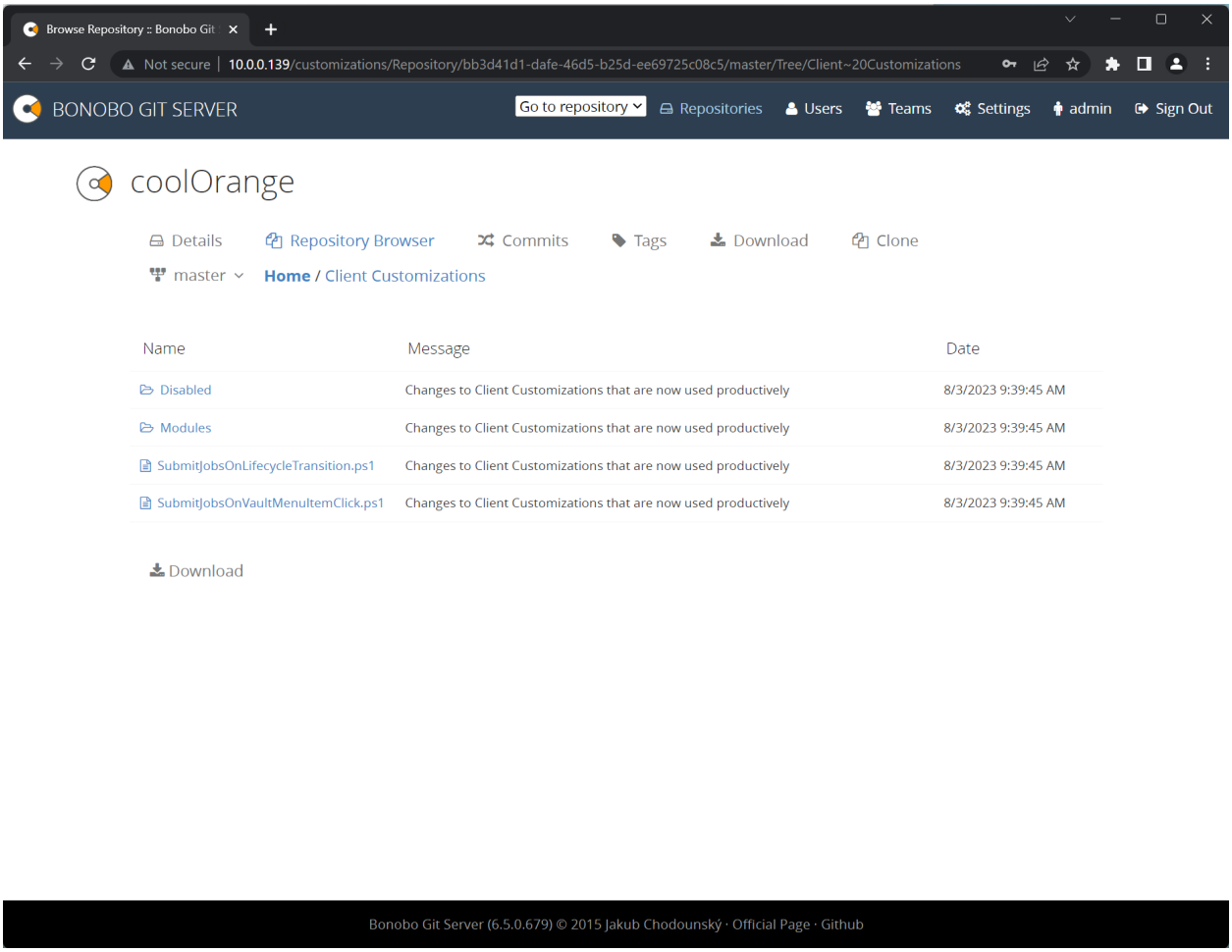


2.3 View customizations using Bonobo Git Server

Launch your browser and go to http://{Your_Vault_Server}/customizations.
Now you can see the initial page of Bonobo Git Server and you can login with the default credentials username: **admin** password: **admin**.

Afterwards you can view the currently published customizations, their publish date, commit messages and more.


Repositories :: Bonobo Git Server
Not secure | 10.0.0.139/customizations/Repository
BONOBO GIT SERVER
Go to repository
Repositories
Users
Teams
Settings
admin
Sign Out
Repositories
Sort Group
Create new repository
Rescan directory
Search
Name
Description
coolOrange
Customizations can be shared to all Vault workstations by pushing the files to this repository.
Bonobo Git Server (6.5.0.679) © 2015 Jakub Chodounský · Official Page · Github


Browse Repository :: Bonobo Git
Not secure | 10.0.0.139/customizations/Repository/bb3d41d1-dafe-46d5-b25d-ee69725c08c5/master/Tree/Client~20Customizations
BONOBO GIT SERVER
Go to repository
Repositories
Users
Teams
Settings
admin
Sign Out
coolOrange
Details
Repository Browser
Commits
Tags
Download
Clone
master
Home / Client Customizations
Name
Message
Date
Disabled
Changes to Client Customizations that are now used productively
8/3/2023 9:39:45 AM
Modules
Changes to Client Customizations that are now used productively
8/3/2023 9:39:45 AM
SubmitJobsOnLifecycleTransition.ps1
Changes to Client Customizations that are now used productively
8/3/2023 9:39:45 AM
SubmitJobsOnVaultMenuItemClick.ps1
Changes to Client Customizations that are now used productively
8/3/2023 9:39:45 AM
Download
Bonobo Git Server (6.5.0.679) © 2015 Jakub Chodounský · Official Page · Github

CHAPTER

THREE

CHANGE LOGS

3.1 Customization Server v2

3.1.1 v2.0.2

`09-08-2023`

General

- Official Release
- Renamed *Customization Share* to *Customization Server*
- An empty `coolOrange` repository is now preinstalled, replacing the original *'powerEvents'* specific repository

3.2 Customization Share v1

3.2.1 v1.0.0

`02-03-2022`

General

- Release of technical preview

Several *coolOrange products* allow scripting guys to provide customizations for automating company wide workflows. After these could be evaluated on test environments in customer-oriented situations, it makes sense to distribute the new files to all workstations.

The Customization Server is a self-hosted Git Server based on a lightweight open-source project, which provides a simple and fast distribution of customizations.