

VERSION CONTROL SYSTEMS

...

VCS

Ein Version Control System (dt. Versionsverwaltung) ist ein System, das zur Erfassung von Änderungen an Dokumenten oder Dateien verwendet wird.

WAS KANN EIN VCS?

- Protokollierungen der Änderungen:
 - Es kann jederzeit nachvollzogen werden, wer wann was geändert hat.
- Wiederherstellung von alten Ständen einzelner Dateien:
 - Somit können versehentliche Änderungen jederzeit wieder rückgängig gemacht werden.
- Archivierung der einzelnen Stände eines Projektes:
 - Dadurch ist es jederzeit möglich, auf alle Versionen zuzugreifen.

WAS KÖNNEN VCS?

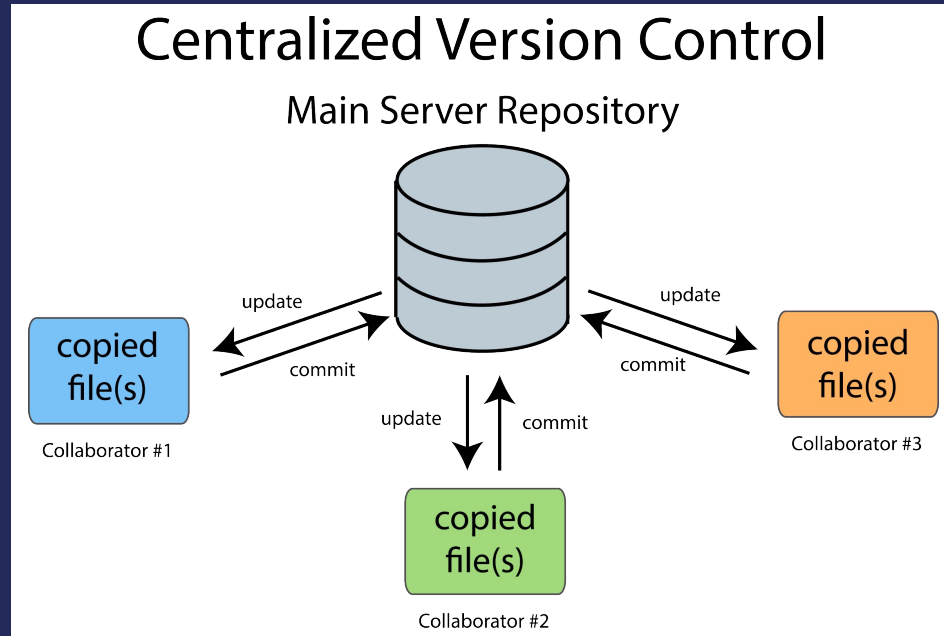
- Koordinierung des gemeinsamen Zugriffs von mehreren Entwicklern auf die Dateien.
- Gleichzeitige Entwicklung mehrerer Entwicklungszweige (engl. *Branch*) eines Projektes.

CVCS & DVCS

- Centralized Version Control Systems
- Distributed Version Control Systems

CVCS

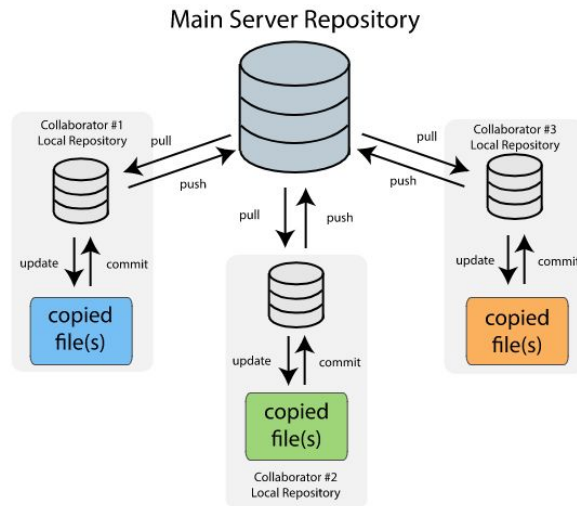
- Centralized Version Control Systems



DVCS

- Distributed Version Control Systems

Distributed Version Control



A Distributed Version Control System. Each collaborator has a local copy of the repository, so no Internet connection is required.

VCS WORKFLOW

- <https://www.git-tower.com/blog/workflow-of-git/>

GIT

...



Git

- Git ist das derzeit meist verwendete VCS und funktioniert als distributed version control system.

Git basics

- Commits
- Branches
- Merge
- Pull
- Push

- Bei jedem Speichervorgang deiner Arbeit erstellt Git einen **commit**.
- Wenn bei einem File keine Änderung vorgenommen wurde übernimmt Git das zuvor unveränderte.
- Erstellte commits sind am **development tree** als weiterführender Link erkennbar.



Commits

- Jeder Developer speichert Änderungen auf seinem lokalen repository, dadurch entstehen unterschiedliche Änderungen basierend auf dem selben Commit.

Aus diesem Grund gibt es tools die es ermöglichen diese auf einen **Branch** zu isolieren und später wieder einzubinden (**merge**).

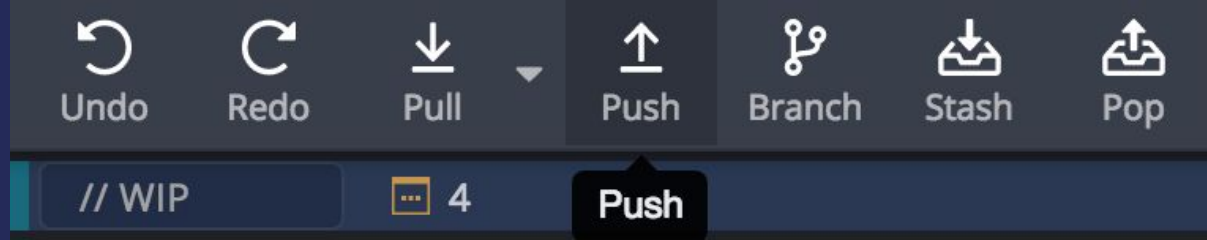


Branches

- Um am aktuellen Stand aller Änderungen Anderer und dem master branch zu bleiben, werden vor einem merge, alle letzten commit mit den Daten der eigenen local repo synchronisiert.

Pull

- Ein Git Push bringt alle Files vom Local ins remote Repository.
- Es versucht alle neuen commits auf den remote branch zu laden und den remote Branch danach mit der local repo zu synchronisieren.



Push

Install git

- Anleitung um git zu installieren:

<https://git-scm.com/book/en/v2/Getting-Started-Installing-Git>

Anleitung um zu Starten

<https://guides.github.com/activities/hello-world/>



Was ist GitHub?

- GitHub ist eine code hosting Plattform für version control und collaboration.
Es ermöglicht mehreren Personen parallel an einem Projekt zu arbeiten.

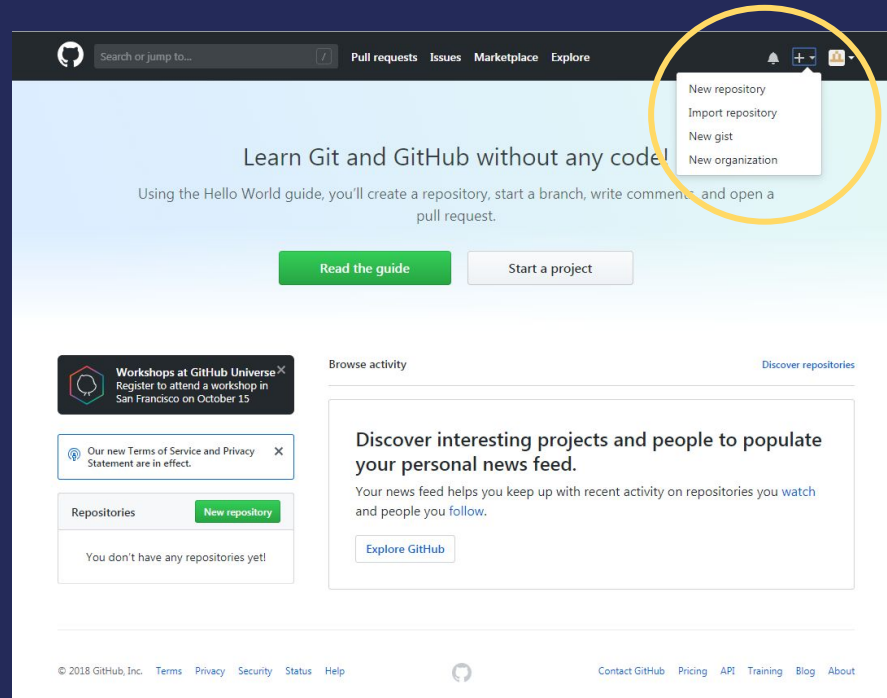
create a GitHub account

- Sign-up on Github:

<https://github.com/join?source=header-home>

Create a repository

- das Repository (auch REPO) wird dazu verwendet dein Projekt zu organisieren



Create a new repository

A repository contains all the files for your project, including the revision history.

Owner



KathrynVil ▾

Repository name

first-repo



Great repository names are short and memorable. Need inspiration? How about [symmetrical-meme](#).

Description (optional)

Das ist mein Beispiel repo



Public

Anyone can see this repository. You choose who can commit.



Private

You choose who can see and commit to this repository.

☒ Initialize this repository with a README

This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.

Add .gitignore: None ▾

Add a license: None ▾



Create repository

adding a collaborator

The screenshot shows the GitHub interface for a repository named 'test-repo' by user 'KathrynVil'. The top navigation bar includes a search bar, links for 'Pull requests', 'Issues', 'Marketplace', and 'Explore', along with notification and user profile icons. Below the repository name, there are buttons for 'Watch', 'Star', and 'Fork', each with a count of 0. A secondary navigation bar contains links for 'Code', 'Issues', 'Pull requests', 'Projects', 'Wiki', 'Insights', and 'Settings' (which is highlighted). On the left, a sidebar menu lists repository settings: 'Options', 'Collaborators' (highlighted), 'Branches', 'Webhooks', 'Integrations & services', 'Deploy keys', 'Moderation', and 'Interaction limits'. The main content area is titled 'Collaborators' with a subtitle 'Push access to the repository'. It contains a message stating that the repository has no collaborators yet and provides a search box to add one. The search instructions specify that users can be found by username, full name, or email address, with a note about email visibility. An 'Add collaborator' button is positioned to the right of the search input.

GitHub interface showing the repository settings for **KathrynVil / test-repo**. The **Collaborators** tab is selected, displaying the option to "Push access to the repository".

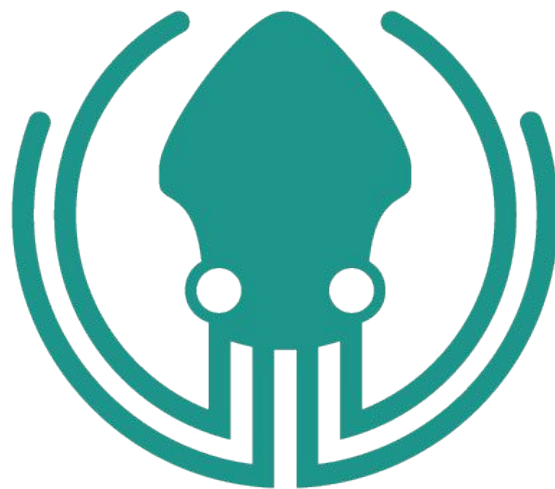
The interface indicates that this repository doesn't have any collaborators yet and provides a form to add a collaborator.

Search by username, full name or email address

You'll only be able to find a GitHub user by their email address if they've chosen to list it publicly. Otherwise, use their username instead.

Add collaborator

Footer: © 2018 GitHub, Inc. | [Terms](#) | [Privacy](#) | [Security](#) | [Status](#) | [Help](#) | [Contact GitHub](#) | [Pricing](#) | [API](#) | [Training](#) | [Blog](#) | [About](#)



axosoft
GitKraken

Was ist GitKraken?

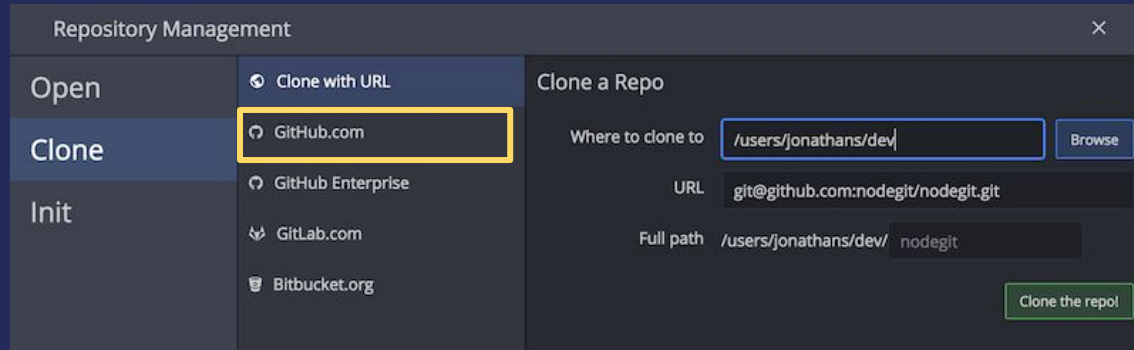
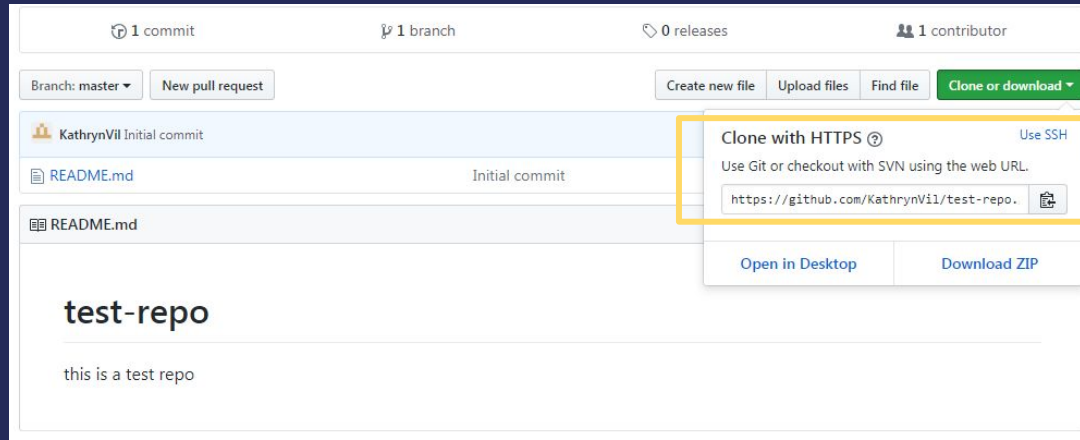
- GitKraken ist ein GIT-GUI, GIT - Graphic User Inteface, diese graphische Benutzeroberfläche ermöglicht einen organisatorischen Überblick eines Projekts

Projektstart

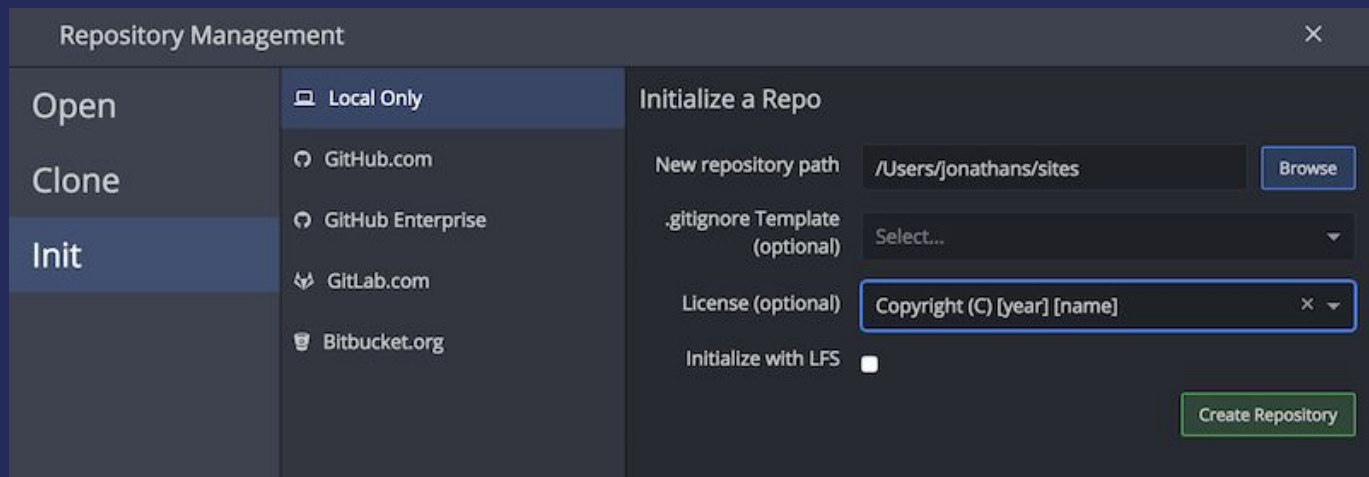
3 Wege eine Git repository für ein Projekt zu starten

1. **Open** - Open a local Git repository already initialized and available locally.
2. **Clone** - Clone a remote Git repository already initialized.
3. **Init** - Create an empty Git repository or reinitialize an existing one.

2. CLONE

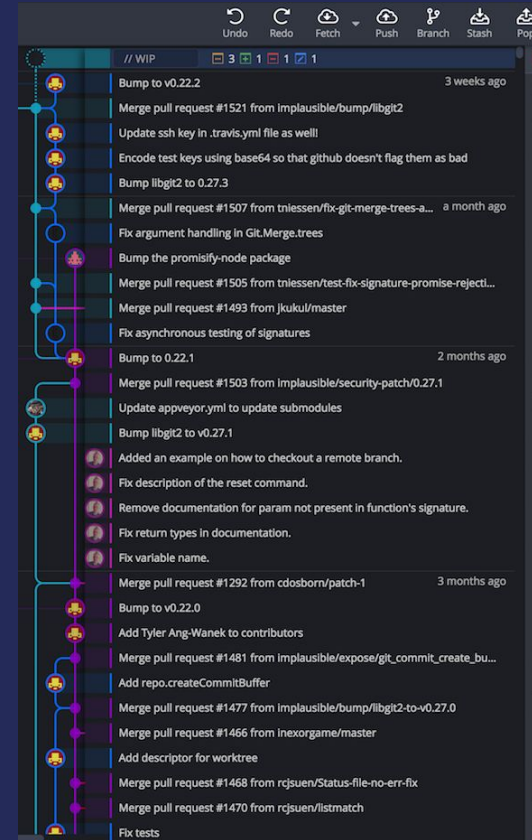


3. INIT

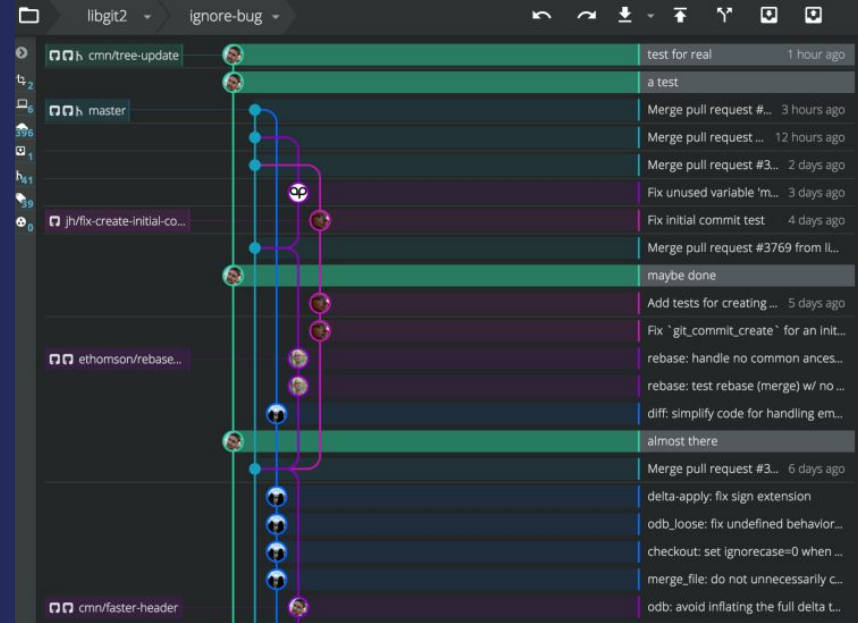
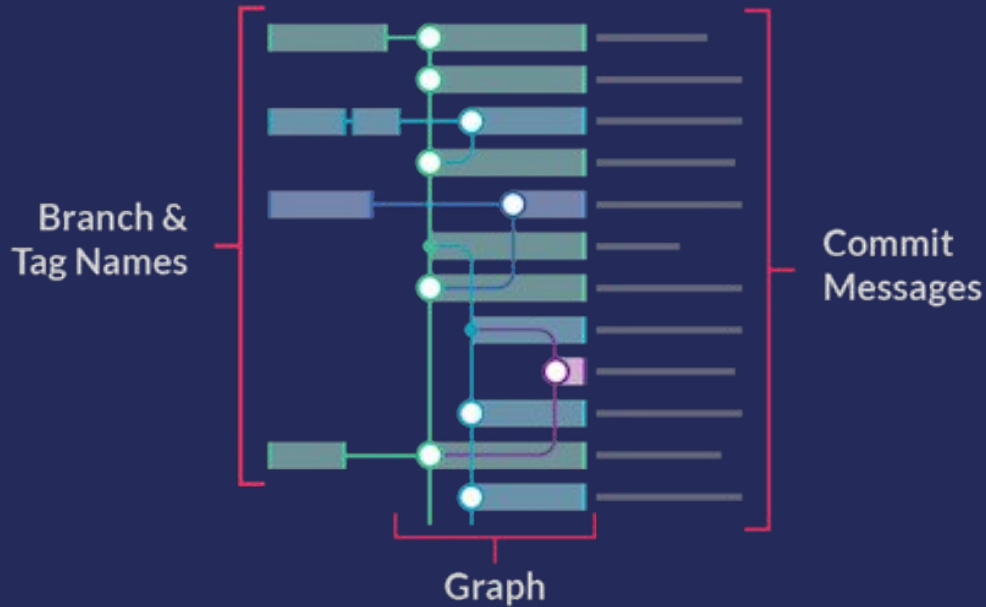


Commit-Panel

Einfacher Überblick aller Commits
der Beteiligten.



Commit-Panel



Commit-Panel

The screenshot displays the Commit-Panel application interface. The top bar includes navigation icons for Undo, Redo, Fetch, Push, Branch, Stash, and Pop, along with a search icon and a menu icon. The main interface is divided into three sections:

- Commit History:** A vertical list of commits on the left, showing commit messages and timestamps. The history includes commits from 3 weeks ago to 3 months ago.
- File Changes:** A central panel showing 6 file changes on the master branch. It lists files: CHANGELOG.md, CONTRIBUTING2.md, new-file.c, package-lock.json, package.json, and TESTING.md. A "Stage all changes" button is visible.
- Commit Message:** A bottom section for entering a commit message, with fields for Summary and Description, and a "Stage files/changes to commit" button.

A legend on the right side of the interface defines the file change status icons: modified (orange square with three dots), added (green square with a plus sign), deleted (red square with a minus sign), and renamed (blue square with a pencil).

Filehistory

The screenshot displays the Filehistory application interface, which visualizes Git commit history and file changes. The interface is divided into several sections:

- Top Bar:** Shows the current repository ('nodegit') and branch ('master'). It includes a search bar for commits and a menu icon.
- Left Panel:** A sidebar showing the local and remote repository structure. It includes a 'LOCAL' section with branches like 'feature-a', 'feature-b', 'master', and 'testing-stash-pop...'. It also shows 'REMOTE' branches, 'PULL REQUESTS', 'TAGS', and 'SUBMODULES'.
- Central Panel:** A large area displaying the commit history as a series of colored circles connected by lines, representing the flow of development. The commits are listed on the right side of this panel, including 'Merge pull request #1079 from novalis/dturner/tree-ent...', 'Fixup errors per @implausible's comments', 'expose rebase options for rebaseBranches', 'expose diff flags option for repo methods', 'fix Tree#entryByName function and add test', '[wlp]', 'expose merge options in rebase options', 'bump vendordep notice', 'Bump to 1.7.0', 'Normalize fn names for overrides', 'pull shallow clone off utils object', 'Stop re-exporting things from lib', 'Sort methods', 'Cap node 6 at 6.2 temporarily', 'Test commit 1', 'Add `maint/0.14` to AppVeyor build branches', 'Increment version in package.json to 0.15.1', 'Bump to 0.14.1', 'Don't fail if postinstall doesn't work', 'Add notes for 0.14.1', 'Bump to 0.15.1', 'Don't fail if postinstall doesn't work', 'Update changelog to use better formatting.', 'Fix windows build', and 'Bump to version 0.15.0'.
- Right Panel:** A detailed view of a specific commit. It shows the commit message, the author's name ('Maximiliano Korp'), the commit hash ('e9f023'), and the parent hash ('7cdf9'). It also lists the files modified in this commit, including 'blob.js', 'clone.js', 'commit.js', 'convenient_hunks.js', 'convenient_patch.js', 'diff.js', 'diff_line.js', 'index.js', 'object.js', 'oid.js', 'reference.js', 'repository.js', and 'revert.js'.

PULL

The screenshot displays the GitKraken application interface. At the top, the repository is identified as 'support.gitkraken.com' and the current branch is 'docs-redesign'. The top toolbar includes icons for Undo, Redo, Pull, Push, Branch, Stash, and Pop. Below the toolbar, the left sidebar shows the local file system structure with branches like 'dev', 'docs-redesign' (checked), 'feature', 'production', 'staging', and 'v3.2.1mg'. The right sidebar shows the commit history, with the most recent commit being 'Update links' 1 hour ago. The central area displays a graphical representation of the commit history, showing a vertical line of commits with a blue line indicating the current branch and a purple line indicating the 'production' branch. A mouse cursor is hovering over the 'dev' branch label, which shows a '+1' change indicator.

GitKraken

- GitKraken kann kostenlos auf der Webseite heruntergeladen werden

<https://www.gitkraken.com/>