

Webcraft

Manual

WebAssembly office and design apps in napp-it CS: menu, installation and update, extensions, acceleration with cs-proxy and integration in cs-team

Version 1.0 · October 29, 2026

Contents

1. What is Webcraft?	3
1.1 The building blocks	3
1.2 Where the files are stored	3
2. The menu Webcraft	4
2.1 Overview	4
3. The base apps	5
4. Installation and update with About > Webcraft Downloads	6
4.1 The columns	6
4.2 Install, update, delete	6
4.3 What napp-it CS takes care of	7
5. Extensions with own .wasm apps	8
5.1 Requirements for an app	8
5.2 Remove, list file	8
6. Acceleration with cs-proxy	9
6.1 Without cs-proxy	9
6.2 Check	9
7. Integration in cs-team	10
7.1 Open the editor	10
7.2 The bar above the app	10
7.3 Exchange of documents	12
7.4 Button for an open app in Text and Calc	12
7.5 Settings and background	13
8. Browser and troubleshooting	14
9. Reference	14

1. What is Webcraft?

Webcraft is the name for a family of **WebAssembly apps** that run completely inside the browser: a spreadsheet (GridCraft), a word processor (WordCraft), a PDF editor (PDFCraft) and many more tools for graphics, audio, video and CAD. napp-it CS does not contain these apps itself. It **installs** them from the GitHub releases of their authors, **serves** them efficiently (cs-proxy) and **links** them (Webcraft menu, cs-team).

- **Runs in the browser, not on the server:** the app is a static web page (index.html, JavaScript, a .wasm file). The server only delivers files. Documents are opened and saved with the file dialog of the browser, they stay on the **local disk** of the user and are not processed by the server.
- **No installation on the clients:** a modern browser is enough (see 8). Every user of the napp-it CS server gets the same apps and the same version.
- **Simple to operate:** an app is a folder below `_my/wwwroot/webcraft/<key>/`. Installing, updating and deleting is done in the GUI (chapter 4), there is no database and no service.
- **Extensible:** any WebAssembly app that is published as a zip in a GitHub release can be added to the list (chapter 5).

1.1 The building blocks

Part	Task	Chapter
Menu Webcraft	Starts the installed apps, each in its own browser window, and shows the state of cs-team.	2
About > Webcraft Downloads	Install, update, delete and open the apps; add your own .wasm apps.	4, 5
cs-proxy (from 0.13.0, storage location from 0.13.1)	Delivers .wasm and the other files compressed, cached and with the correct type (Content-Type application/wasm).	6
cs-team (from 0.65)	Shows WordCraft, GridCraft and PDFCraft directly in its window (menus Text, Calc, Files) and exchanges documents with them.	7

1.2 Where the files are stored

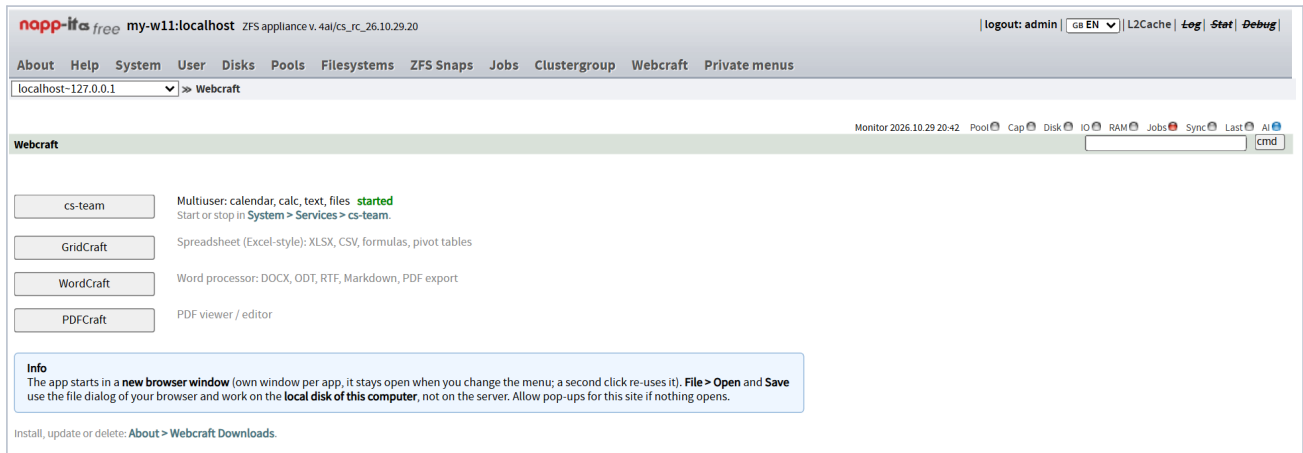
All installed apps are stored below `<csweb-gui>/_my/wwwroot/webcraft/<key>/` (for example `C:\opt\csweb-gui\_my\wwwroot\webcraft\wordcraft`). The folder `_my` is **not part of the napp-it CS distribution**: an update or a new installation of napp-it CS does not touch the apps, and the apps (about 20 to 50 MB each) are not shipped with it. Download, update and delete (About > Webcraft Downloads), the web server (webserver.pl) and cs-proxy all use this one place. Back up `_my` together with `_cfg` if you want to keep the installed apps. An app that was installed by an older version below `data/wwwroot/webcraft/` is still delivered, but is no longer managed by the page: move the folder to `_my/wwwroot/webcraft/`.

Not part of napp-it CS: the apps themselves. WordCraft, GridCraft, PDFCraft and the other ArtCraft apps come from the project **storytold** on GitHub, SolidCraft from **tangency-labs**. napp-it CS only downloads them, on request of an admin, from the public releases. Their licence terms apply, please read them in the respective repository.

2. The menu Webcraft

The main menu **Webcraft** (folder data/menues/60_Webcraft) is the entry point for the users. It lists all installed apps and starts each one with a click.

2.1 Overview



Menu Webcraft: cs-team with state, installed apps, info box and link to the downloads

- **cs-team** is the first entry: a button opens the cs-team web interface in its own window (only if cs-team is started) and shows the state **started** or **stopped**. The check is the same as in System > Services > cs-team; **starting and stopping is done only there**.
- **Installed apps** (here GridCraft, WordCraft and PDFCraft) follow with a short description. The list is built from the folders that really exist below `_my/wwwroot/webcraft/`, a not installed app does not appear.
- Each app starts in its **own browser window** (window name `webcraft_<app>`, no frame). The window stays open when you change the menu; a second click re-uses it. Allow pop-ups for the site if nothing opens.
- **File > Open** and **Save** inside an app use the file dialog of the browser and work on the **local disk of the computer** on which the browser runs, not on the server.
- The link at the bottom leads to About > Webcraft Downloads for install, update and delete (chapter 4).

The address of an app is `https://<host>/webcraft/<app>/`, for example `https://my-server/webcraft/wordcraft/`. It can be bookmarked or given to users directly.

3. The base apps

The list in *About > Webcraft Downloads* contains the **ArtCraft** apps of the project storytold and, as a separate project, SolidCraft. The table shows the key (name of the folder and of the address), the purpose and the source of the download. Which apps are installed is up to the admin; for the office work in cs-team three are enough (marked).

App (key)	Purpose	Source (GitHub)
GridCraft (gridcraft) *	Spreadsheet in the style of Excel: XLSX, CSV, formulas, pivot tables	storytold/gridcraft
WordCraft (wordcraft) *	Word processor: DOCX, ODT, RTF, Markdown, PDF export	storytold/wordcraft
PDFCraft (pdfcraft) *	PDF viewer and editor	storytold/pdfcraft
DeckCraft (deckcraft)	Presentations in the style of PowerPoint	storytold/deckcraft
DesignCraft (designcraft)	Page and layout design (desktop publishing)	storytold/designcraft
CadCraft (cadcraft)	2D CAD and drafting (AutoCAD style)	storytold/cadcraft
VectorCraft (vectorcraft)	Vector graphics	storytold/vectorcraft
FigCraft (figcraft)	Interface design and prototyping	storytold/figcraft
EffectCraft (effectcraft)	Motion graphics and compositing	storytold/effectcraft
SoundCraft (soundcraft)	Audio editing	storytold/soundcraft
PhotoCraft (photocraft)	Photo editor	storytold/photocraft
FilmCraft (filmcraft)	Video editor	storytold/filmcraft
LightCraft (lightcraft)	Non-destructive photo editor	storytold/lightcraft
SolidCraft (solidcraft)	Parametric 3D CAD (separate project, not part of ArtCraft)	tangency-labs/solidcraft

* These three apps are integrated in cs-team (chapter 7): WordCraft for **Text**, GridCraft for **Calc**, PDFCraft for **Files**. Under **Start** all installed apps are listed alphabetically under *Single user (local)*.

Languages: the apps are not translated by napp-it CS. The project storytold keeps shared translations in the repository *craft-i18n* (about 37 languages, among them German, French, Spanish, Italian, Russian and Chinese). An app has them only if its release was built with them: at the time of writing WordCraft, PDFCraft, VectorCraft, PhotoCraft, CadCraft and FilmCraft have catalogs, the other apps (for example GridCraft) show English. The number of languages differs from app to app and grows with new releases.

The list is built into napp-it CS. New releases of the apps appear in the column *Newest* without any change of napp-it CS. Apps that are not in the list can be added as own entries (chapter 5).

4. Installation and update with About > Webcraft Downloads

The page **About > Webcraft Downloads** (only for admins) lists all apps with the installed and the newest version and does everything that is needed: download, update, delete, open.

The screenshot shows the napp-it web interface. At the top, there's a navigation bar with 'About', 'Help', 'System', 'User', 'Disks', 'Pools', 'Filesystems', 'ZFS Snaps', 'Jobs', 'Clustergroup', 'Webcraft', and 'Private menus'. Below this is a header for 'Webcraft Downloads' with a search bar and a 'cmd' button. The main content area shows a table of Webcraft apps. The table has four columns: 'Webcraft app', 'Installed', 'Newest', and 'Action'. The 'Webcraft app' column lists apps like GridCraft, WordCraft, DeckCraft, PDFCraft, CadCraft, VectorCraft, FigCraft, EffectCraft, SoundCraft, PhotoCraft, FilmCraft, LightCraft, and SolidCraft. The 'Installed' column shows the current status (e.g., 'installed, version unknown' or 'not installed'). The 'Newest' column shows the latest version (e.g., 'v0.13.0' or 'v0.26.1'). The 'Action' column contains buttons for 'download', 'update', 'open', and 'delete'. Below the table is a form to 'Add another wasm app' with fields for 'Key (folder, a-z 0-9 _-)', 'GitHub repo', 'Name (optional)', and 'Asset regex (optional)'. At the bottom, there's a note about the latest GitHub release and how to verify it.

Webcraft app	Installed	Newest	Action
GridCraft Spreadsheet (Excel-style): XLSX, CSV, formulas, pivot tables storytold/gridcraft	installed, version unknown	v0.13.0	update open delete
WordCraft Word processor: DOCX, ODT, RTF, Markdown, PDF export storytold/wordcraft	installed, version unknown	v0.12.0	update open delete
DeckCraft Presentations (PowerPoint-style) storytold/deckcraft	not installed	v0.12.0	download
PDFCraft PDF viewer / editor storytold/pdfcraft	installed, version unknown	v0.15.0	update open delete
CadCraft 2D CAD and drafting (AutoCAD-style) storytold/cadcraft	not installed	v0.14.0	download
VectorCraft Vector graphics storytold/vectorcraft	not installed	v0.18.0	download
FigCraft Interface design and prototyping storytold/figcraft	not installed	v0.12.1	download
EffectCraft Motion graphics and compositing storytold/effectcraft	not installed	v0.18.0	download
SoundCraft Audio editing storytold/soundcraft	not installed	v0.14.0	download
PhotoCraft Photo editor storytold/photocraft	not installed	v0.17.0	download
FilmCraft Video editor storytold/filmcraft	not installed	v0.14.0	download
LightCraft Non-destructive photo editor storytold/lightcraft	not installed	v0.15.0	download
SolidCraft Parametric 3D CAD (separate project, not part of ArtCraft) tangency-labs/solidcraft	not installed	v0.26.1	download

Add another wasm app

Key (folder, a-z 0-9 _-):

GitHub repo:

Name (optional):

Asset regex (optional):

[add to list](#)

The latest GitHub release of the repo must contain a zip with a static web build (index.html, .js, .wasm). It is installed to `data/wwwroot/webcraft/<key>/` and opened as `https://<host>/webcraft/<key>/`. If the release publishes a SHA256SUMS file the download is verified. Newest versions are cached for 1 h (`_cfg/webcraft_versions`, GitHub allows 60 anonymous requests per hour). cs-proxy 0.13.0 or newer serves the apps compressed and cached; without it webserver.pl delivers them uncompressed. After an app update press Ctrl+F5 in the browser the old version still shows.

About > Webcraft Downloads: apps with installed and newest version and the actions download, update, open, delete

4.1 The columns

Column	Meaning
Webcraft app	Name, short description and the GitHub repository (owner/repo) of the app.
Installed	<i>not installed</i> or the installed version (tag of the release). <i>installed, version unknown</i> means: the folder exists but was not installed by this page (for example copied by hand).
Newest	Tag of the latest release on GitHub (cached for one hour).
Action	download (not installed), update , open (opens the app in a new window) and delete .

4.2 Install, update, delete

1. Open **About > Webcraft Downloads**. The page asks GitHub for the newest release of every app (anonymous, for one hour at most; **refresh newest versions** forces a new query).
2. Click **download** behind an app. napp-it CS fetches the zip of the latest release, checks it and unpacks it to `_my/wwwroot/webcraft/<key>/`. The installed tag is written to the file `.cs-version` in this folder.
3. A new release shows another tag in the column *Newest*. **update** replaces the app with the new version, **open** starts it.

4. **delete** removes the folder of the app. Documents are not affected, they are on the disks of the users.
5. After an update press **Ctrl+F5** in the browser, otherwise the window may still show the old version.

4.3 What napp-it CS takes care of

- **TLS is always verified:** the download is made with verified TLS. The bundled perl has no CA bundle, therefore the system downloader (curl or PowerShell, with the trust store of the operating system) is used as fall-back. The certificate check is never switched off.
- **Integrity:** if the release contains a file SHA256SUMS (checksums), the zip is checked against it before it is unpacked.
- **Safe unpacking:** only paths inside the app folder are accepted, no `..`, no absolute paths.
- **GitHub limit:** anonymous access allows 60 queries per hour. The newest versions are therefore kept for one hour in `_cfg/webcraft_versions`; *refresh newest versions* reads them again.
- **Which asset:** the release must contain a zip named `<key>-web-...zip` (a static web build). Other assets (installers, source) are ignored.

Without internet on the server: copy the unpacked app by hand into `_my/wwwroot/webcraft/<key>/`. It is then listed as installed (version unknown) and appears in the menu Webcraft; only update and delete via GitHub are not available.

5. Extensions with own .wasm apps

Below the list there is the form **Add another wasm app**. With it any app that is published as a static web build in a GitHub release can be added, without a change of napp-it CS. It then gets download, update, open and delete like the base apps.

Field	Meaning
Key	Name of the folder and of the address, 1 to 40 characters from a-z 0-9 _ -, for example minecraft. The app is then reachable as <code>https://<host>/webcraft/minecraft/</code> . The key must be new.
GitHub repo	owner/repo or the complete address <code>https://github.com/owner/repo</code> , for example tangency-labs/solidcraft.
Name (optional)	Display name in the list and in the menu. Without entry the key is used.
Asset regex (optional)	Regular expression for the name of the zip in the release. Default: <code>^<key>-web-.*\.zip\$</code> .

5.1 Requirements for an app

- A **release on GitHub** with a **zip** of the static web build: `index.html`, JavaScript and the `.wasm` file(s). No server component, no build step on the napp-it CS server.
- The page must load all files through **relative addresses** (as usual for trunk and wasm-bindgen builds), because it is served below `/webcraft/<key>/`.
- The browser must support what the app needs (WebGPU or WebGL, see 8). The apps are drawn with WebGPU, which needs a secure context: `https://` or `localhost`.
- `index.html` must be in the root of the zip or in **one** top folder (for example `<app>-web-<version>/`); the installer takes that folder as the app. Without `index.html` the installation is refused.

5.2 Remove, list file

- Own entries have the button **remove entry** (after a confirmation). The installed folder is deleted separately with *delete*. The built-in entries cannot be removed.
- The list of own entries is kept in `_cfg/webcraft/custom.conf`, one line per app: `key|name|owner/repo|asset_regex`. It can be edited by hand and is part of the normal backup of the `_cfg` folder.
- Entries of the built-in list cannot be overwritten, a key that already exists is refused.

Example: SolidCraft (parametric 3D CAD) is not part of ArtCraft. It is in the list as `tangency-labs/solidcraft` and works exactly like an own entry. The same way a clone of an app, a modified build or a tool of your own can be offered to the users.

6. Acceleration with cs-proxy

A WebAssembly app is big: the .wasm file of GridCraft has about **21 MB**. Without help every user would load it uncompressed with every start. **cs-proxy from version 0.13.0** (file webcraft.go) therefore has a special path for everything below /webcraft/. It looks for the files first in _my/wwwroot/webcraft/ (storage location, chapter 1.2) and, from version 0.13.1, falls back to data/wwwroot/webcraft/:

Function	Effect
Correct type	.wasm is delivered as application/wasm. Browsers need this for fast streaming compilation. Every other file type (js, mjs, json, css, svg, ...) gets its proper Content-Type.
gzip, compressed once	wasm, js, html, css and json are compressed once per file version and kept in memory. The 21.4 MB of GridCraft become 8.2 MB ; the first request takes about 0.8 s, all others are served from the cache. Condition: proxy_compress contains gzip and the browser sends Accept-Encoding: gzip.
Own small cache	Not counted in proxy_cache_max_mb, only for /webcraft/. A changed file (size or time) is compressed again automatically, so an update needs no restart.
ETag, 304, Range	Unchanged files are not sent again (If-None-Match gives 304), parts of a file can be requested (Range gives 206).
Immutable caching	Files and folders with a 16 digit hash in the name (name-32ffbcd5056039fd.js, trunk and wasm-bindgen style) get <i>immutable</i> : the browser never asks again. All other files are revalidated with the ETag (<i>no-cache</i>).
Folder addresses	/webcraft/app is redirected to /webcraft/app/, there index.html is delivered. The apps load everything through relative addresses and need this slash.
Safe	Only GET and HEAD, .. is refused (403), symbolic links must stay inside the document root, a missing file gives 404.

6.1 Without cs-proxy

If the web server is used directly (proxy=off), webserver.pl delivers the apps as well (since version 26.10.29.45): MIME types for wasm, mjs, map and webmanifest, redirect of folder addresses, index.html. It works, but **without gzip and without immutable caching**, so the first start is slower. For more than a few users cs-proxy is recommended.

6.2 Check

```
curl -skI -H "Accept-Encoding: gzip" https://my-server/webcraft/gridcraft/
```

```
HTTP/1.1 200 OK          index.html is delivered (not a redirect, not 404)
```

```
curl -skI -H "Accept-Encoding: gzip" https://my-server/webcraft/gridcraft/<name>.wasm
```

```
Content-Type: application/wasm
```

```
Content-Encoding: gzip    with cs-proxy; a repeated request is answered from the cache
```

After a new installation of cs-proxy the running process must be restarted once (System > Services), otherwise the old program is used.

7. Integration in cs-team

cs-team (calendar, calc, text, files, chat) shows the three apps **inside its own window** and exchanges documents with them. The apps stay what they are, single-user editors with the file formats of Office; cs-team stays the multi-user tool for the shared documents. Both complement each other:

cs-team menu	Editor	Typical use
Text	WordCraft	Letters and reports with Word formats (DOCX, ODT, RTF, Markdown), PDF export.
Calc	GridCraft	Spreadsheets with XLSX and CSV, formulas and pivot tables.
Files	PDFCraft	View and edit PDF files.

Requirement: cs-team **0.65 or newer** (test builds from 0.64.1-test3) and the apps installed as in chapter 4. cs-team and the apps run on the same host.

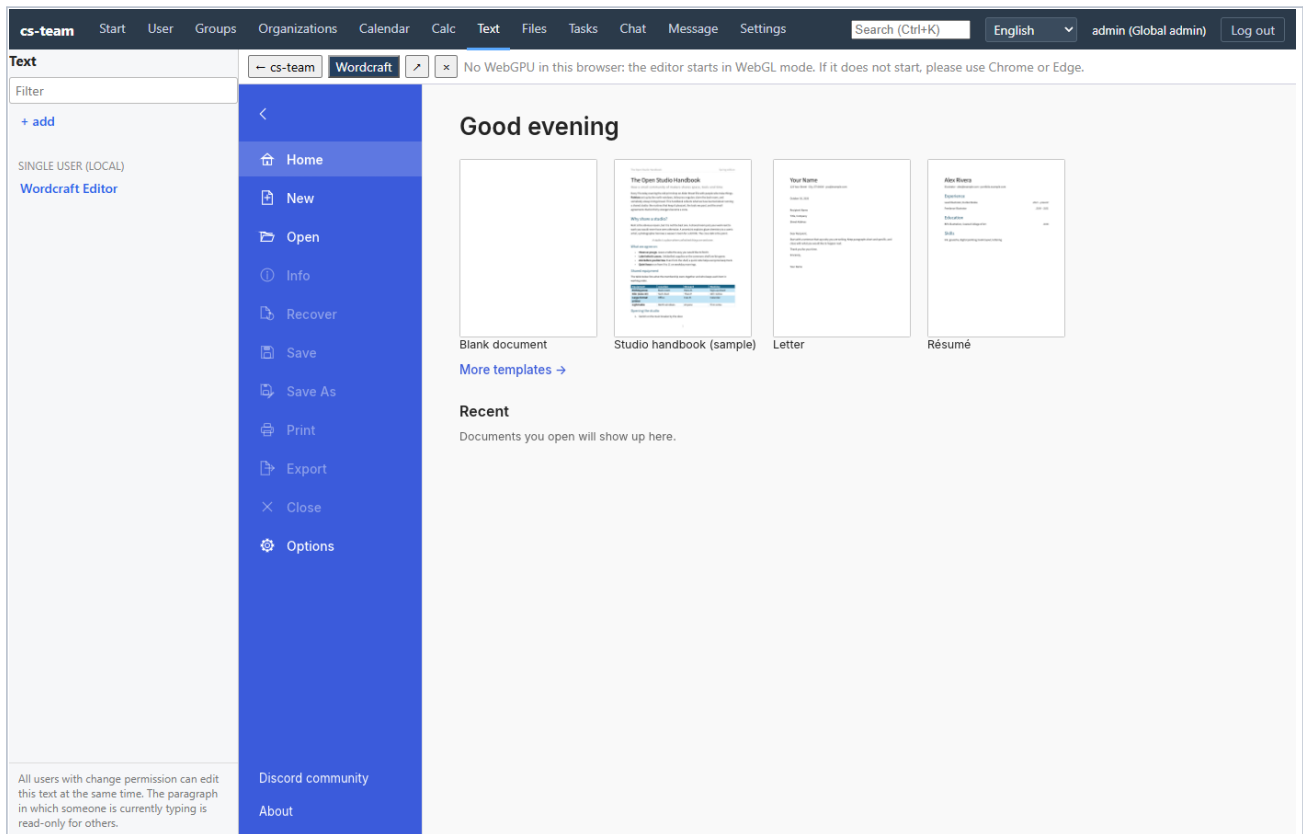
7.1 Open the editor

In the menu **Start** the left column shows the section **Singleuser (local)** with **all installed apps** in alphabetical order. In the menus Text and Calc it shows, below **+ add**, the entry of the matching editor. A click opens the app **in the right part of the cs-team window**.

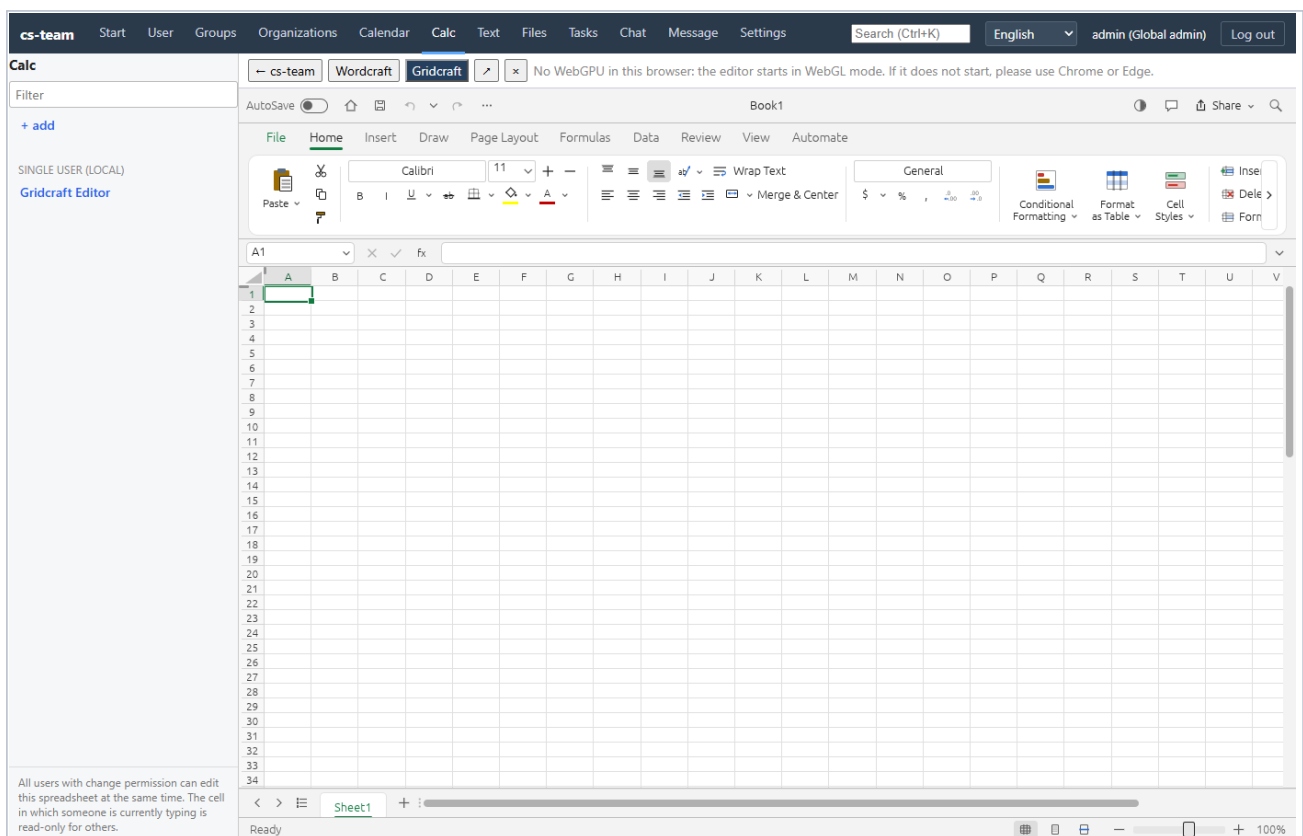
Menu Text: below + add the section Singleuser (local) with Wordcraft Editor

7.2 The bar above the app

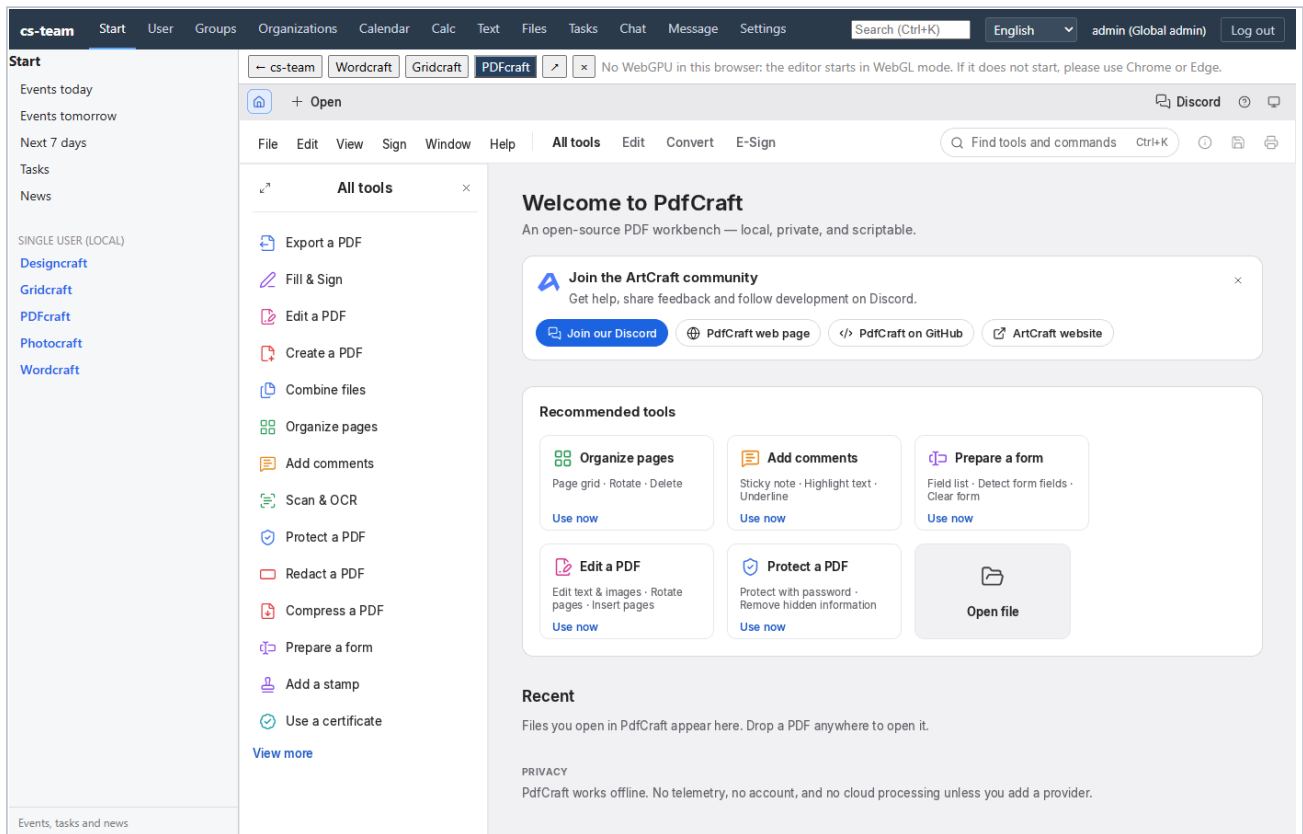
- ← **cs-team** goes back to the cs-team menu. The apps stay open in the background with all their content.
- One button per **opened app** switches between WordCraft, GridCraft and PDFCraft; the active one is highlighted.
- ↗ opens the active app in its own browser window (for a big screen or a second monitor).
- ✕ closes the app. Changes that were not saved with *File > Save* are lost.



WordCraft in the cs-team window: bar with back button, app buttons, new window and close



GridCraft in the menu Calc



PDFCraft opened from the menu Start (Singleuser (local))

7.3 Exchange of documents

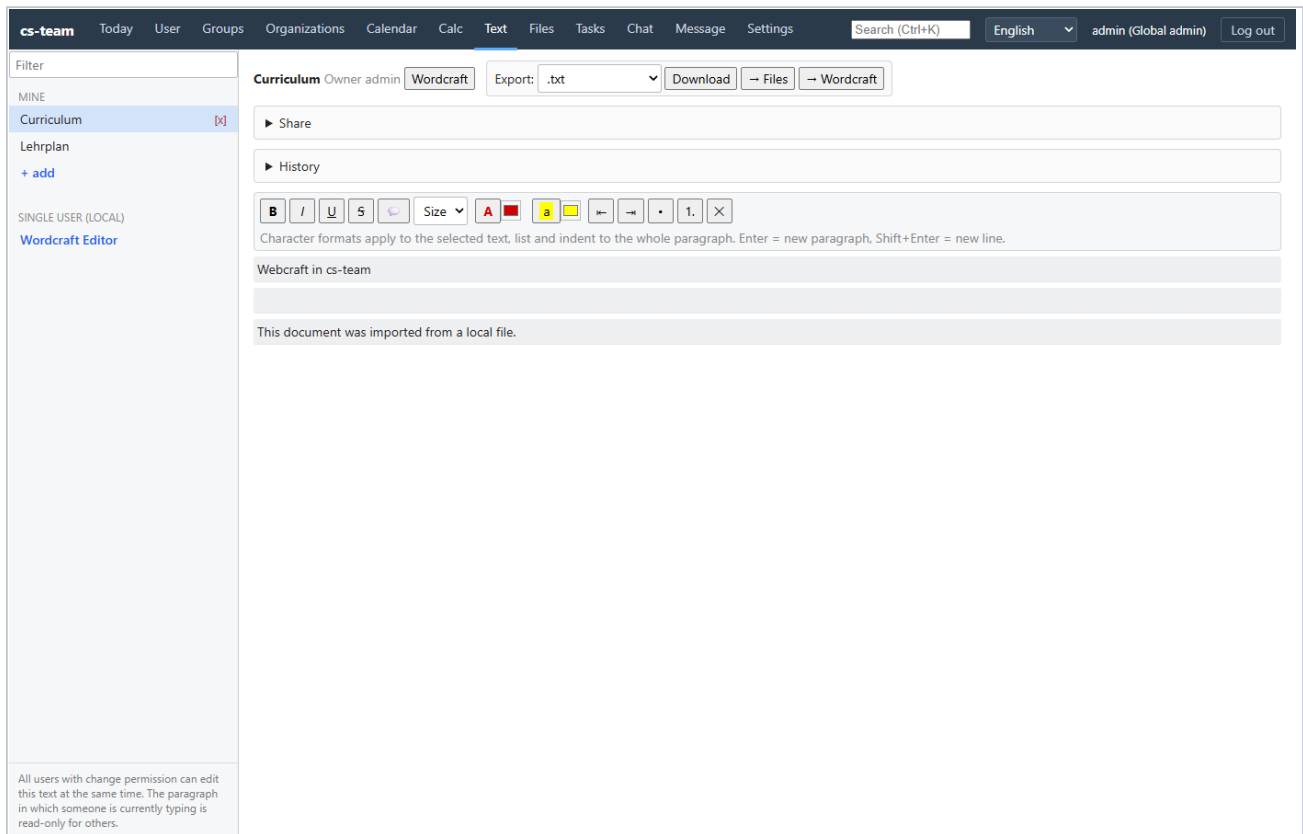
There are three ways, from simple to comfortable:

1. **Clipboard:** in an open Text or Calc document the button → **Wordcraft** or → **Gridcraft** (next to Download) copies the content (Text as plain text, Calc as cell values separated by tabs, without formulas and formats) to the clipboard and switches to the app. There insert with **Ctrl+V**.
2. **File, from the app to cs-team:** in the app *File* > *Save* (DOCX, TXT, XLSX or CSV) to the local disk, then in cs-team + **add** > **Import from local file:** select the file, the name is taken from the file name and can be changed, share with the group as with every new document. Formats: Text: txt, docx, ctext; Calc: csv, xlsx, cscalc.
3. **File, from cs-team to the app:** download the document in cs-team (Export), then in the app *File* > *Open*.

cs-team can **not** steer the apps: they offer no interface to the host (only the parameters ?webgl and ?sample). The exchange therefore always goes through the clipboard or a file on the local disk.

7.4 Button for an open app in Text and Calc

If WordCraft (or GridCraft) is already open and you work in a Text (or Calc) document of cs-team, the head line of the document shows a **button with the name of the app** next to the name of the document (above the text or the table). One click switches back to the open editor, **without copying**. The button appears only as long as the app is open; → **Wordcraft** (copy and switch) remains available.



Open document in Text: the button Wordcraft in the head line switches to the open editor

7.5 Settings and background

- **Address:** cs-team loads the app from `https://<host of the page>/webcraft/<app>/`. If napp-it CS answers on another address (other port), enter it once in the console of the browser: `localStorage.setItem("cs.webcraft.base", "https://host:8443/webcraft/")`.
- **Security:** the Content Security Policy of cs-team allows frames only from the page itself and from the same host (https, any port). Other addresses are not framed.
- **Import from local file** in the REST interface: POST `/api/docs/importfile?file=<name.ext>&name=<optional>`, the body is the content of the file; same checks and formats as the import from Files.
- **Start of cs-team:** from the menu Webcraft (state) and System > Services > cs-team (start, stop).

8. Browser and troubleshooting

The apps draw their surface with **WebGPU**. Chrome and Edge (current versions) have it; other browsers may not (older Firefox versions, depending on system and graphics driver). The page must be opened with `https://` or from `localhost` (secure context).

Symptom	Cause and remedy
The app stays at <i>Loading ...</i>	The browser has no WebGPU. In cs-team the app is started automatically in the WebGL mode (parameter <code>?webgl</code>) and the bar shows a hint. In a window of its own add <code>?webgl</code> to the address or use Chrome or Edge.
Nothing opens on click	The browser blocks pop-ups of the site. Allow them (menu Webcraft opens a window per app).
404 or blank page	The app is not installed (About > Webcraft Downloads) or the address has no slash at the end. Without cs-proxy 0.13.0 the web server must be a version with wasm support (26.10.29.45).
Very slow first start	Without cs-proxy the 20 MB are sent uncompressed. Activate cs-proxy and <code>proxy_compress = gzip</code> (chapter 6).
After an update the old app shows	Press Ctrl+F5 (the browser cache).
Download fails with GitHub error	Anonymous limit of 60 queries per hour or no internet. Wait, or install the app by hand (chapter 4.3).
Frame in cs-team stays empty	Address differs (other port): set <code>cs.webcraft.base</code> (chapter 7.5). cs-team and napp-it CS must be reachable on the same host name.
My documents are gone after closing	Documents are not stored on the server. Save with <i>File > Save to the local disk</i> , then import into cs-team if they should be shared.

9. Reference

Item	Location / value
Installed apps (storage location)	<code>_my/wwwroot/webcraft/<key>/</code> (not in the distribution), version in <code>.cs-version</code>
Address of an app	<code>https://<host>/webcraft/<key>/</code>
Own entries	<code>_cfg/webcraft/custom.conf</code> (key name owner repo asset_regex)
Cache of the versions	<code>_cfg/webcraft_versions</code> (1 hour)
Menu Webcraft	<code>data/menues/60_Webcraft/action.pl</code>
Download page	<code>data/menues/_lib/tools/Webcraft_Downloads/action.pl</code> , logic in <code>data/menues/_lib/windows/webcraftlib.pl</code>
Delivery	cs-proxy 0.13.0 <code>webcraft.go</code> ; fall-back <code>webserver.pl</code> (26.10.29.45)
cs-team	0.65 or newer: <code>web/app.js</code> (editors in Text, Calc, Files), route <code>/api/docs/importfile</code>
More documentation	napp-it CS first steps, cs-proxy manual, cs-team manual (napp-it.org > Downloads)