

ComfyUI Followup

Von phänomenalen Bildern, Videos und Musik bis DeepFakes



- Digitale Sprechstunde am 19.1.2026 -

Klaus Knopper <klaus.knopper@hs-kl.de>

Vizepräsident für Digitalisierung an der Hochschule Kaiserslautern

Zum Kennenlernen: Umfrage

Haben Sie ComfyUI schon installiert (oder in der letzten DigiSprech gesehen)?

Ja/Nein

Wdh: Ein paar notwendige Infos vorab (Deepfakes etc.)...

Nicht alles, was man mit KI-Werkzeugen tun **kann**, **darf** man auch:

- Verletzung der Privatsphäre → [§ 201a StGB](#)
 - Verbreitung (und Besitz) pornografischer Inhalte → [§ 184 StGB](#) u. → § 184 [a,b,c](#)
 - Ehrverletzungen → [§ 185 StGB](#)
 - falsche Tatsachenbehauptungen über eine Person → [§ 186 StGB](#)
 - Jugendschutz → [§ 18 Abs. 1 JuSchG](#)
 - Täuschung mit Vermögensschaden → [§ 263 StGB](#)
 - Urheberrecht des Originalmaterials → [§ 106](#) und → [§ 108 UrhG](#)
 - Kennzeichnungspflicht für KI-generierte Inhalte → [EU KI-VO Art. 50](#)
- **Anbieter** von **Online KI-Diensten** müssen entsprechende **Filter** und **Altersverifikation** implementieren (s.a. → [Heise über X/Grok](#))
- **Nutzer** (online wie offline) müssen sich (natürlich) auch **an die Gesetze halten**.

ComfyUI Webseite

The logo for ComfyUI, featuring the word "Comfy" in a large, bold, blue, italicized sans-serif font, followed by "ui" in a smaller, blue, italicized sans-serif font.

→ <https://comfy.org>

Installation und Start „from Source“ unter Debian (weil alles andere zu einfach ist. ; -)]

Nur einmal (Installation)

```
sudo apt update
sudo apt install -y git python3 python3-venv python3-pip wget ffmpeg
git clone https://github.com/comfyanonymous/ComfyUI.git
cd ComfyUI
python3 -m venv venv
source venv/bin/activate
python -m pip install --upgrade pip
python -m pip install -r requirements.txt
Fertig (ggf pytorch noch mal reinstallieren, s.u.)
Als nächstes: python main.py (s. nächste Folie)
```

Immer (zum Starten)

Achtung: Bei der Installation von pytorch per `requirements.txt` auf eventuelle Hinweise achten (you need pytorch xxxxx for your graphics card, install with `pip install ...`). In diesem Fall, nach Durchlauf mit `python -m pip uninstall pytorch` wieder deinstallieren, und dann das vorgeschlagene Kommando ausführen (z.B. `python -m pip install torch torchvision torchaudio --extra-index-url https://download.pytorch.org/whl/cu118`)

ComfyUI lokal starten (s.o.)

```
cd Installationsverzeichnis
```

```
source venv/bin/activate
```

```
python main.py
```

→ GUI läuft unter <http://127.0.0.1:8188/>

Stoppen: mit Strg-C

Wdh: Wichtige Begriffe für Komponenten in ComfyUI (models/*)

- **checkpoints:** Snapshot von Diffusions-Modellen, die alle für die Bildgenerierung notwendigen Schritte enthalten
- **diffusion_models:** Sprachmodelle spezifisch für Video- und Bildgenerierung (für den Generationsprozess)
- **loras:** „Low-Rank Adaption“, Fine-Tuning für generative Modelle, ohne das Modell an sich neu trainieren zu müssen.
- **text_encoders:** Erzeugen aus natürlicher Sprache die numerischen Anweisungen für die bildgenerierenden LLMs
- **vae:** „variational autoencoder“, transformiert Bilder zwischen Pixel- und interner Repräsentation (Arbeitsbereich) und zurück.

Sobald ein Workflow aus den „Templates“ ausgewählt wird, erscheint ggf. ein Dialog, der anbietet, fehlende der o.g. **Komponenten herunterzuladen**. Zielverzeichnis unter models/... beachten!

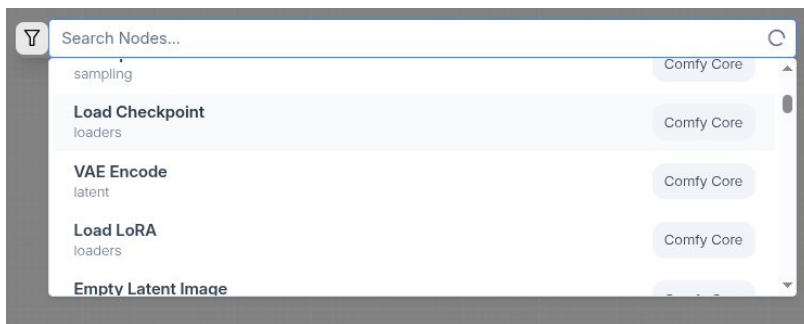
Ein Workflow „from scratch“ - Beispiel (1)

Voraussetzung: Es sollte schon mindestens ein „checkpoint“-Modell heruntergeladen worden sein (z.B. aus einem Template). Im Beispiel:
[checkpoints/sd3.5_large_fp8.safetensors](https://civitai.com/models/100000/sd3.5_large_fp8_scaled.safetensors).

Wir starten mit einem LEEREN Workflow (+).

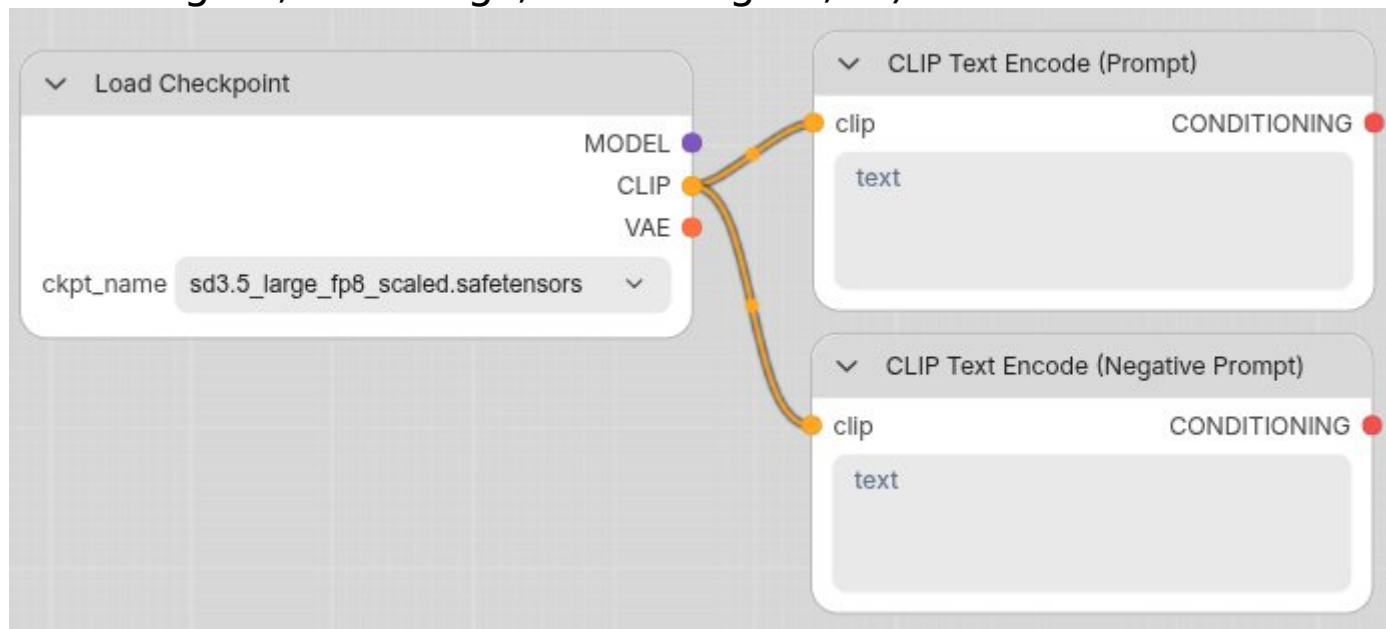
1. Doppelklick ins leere Fenster, Node „Load Checkpoint Model“ auswählen und platzieren.

Von hier aus arbeiten wir uns weiter, indem wir die Output-Punkte nach außen ziehen, neue Nodes einfügen oder output-Punkte mit Input-Punkten verbinden, s. nächste Schritte.



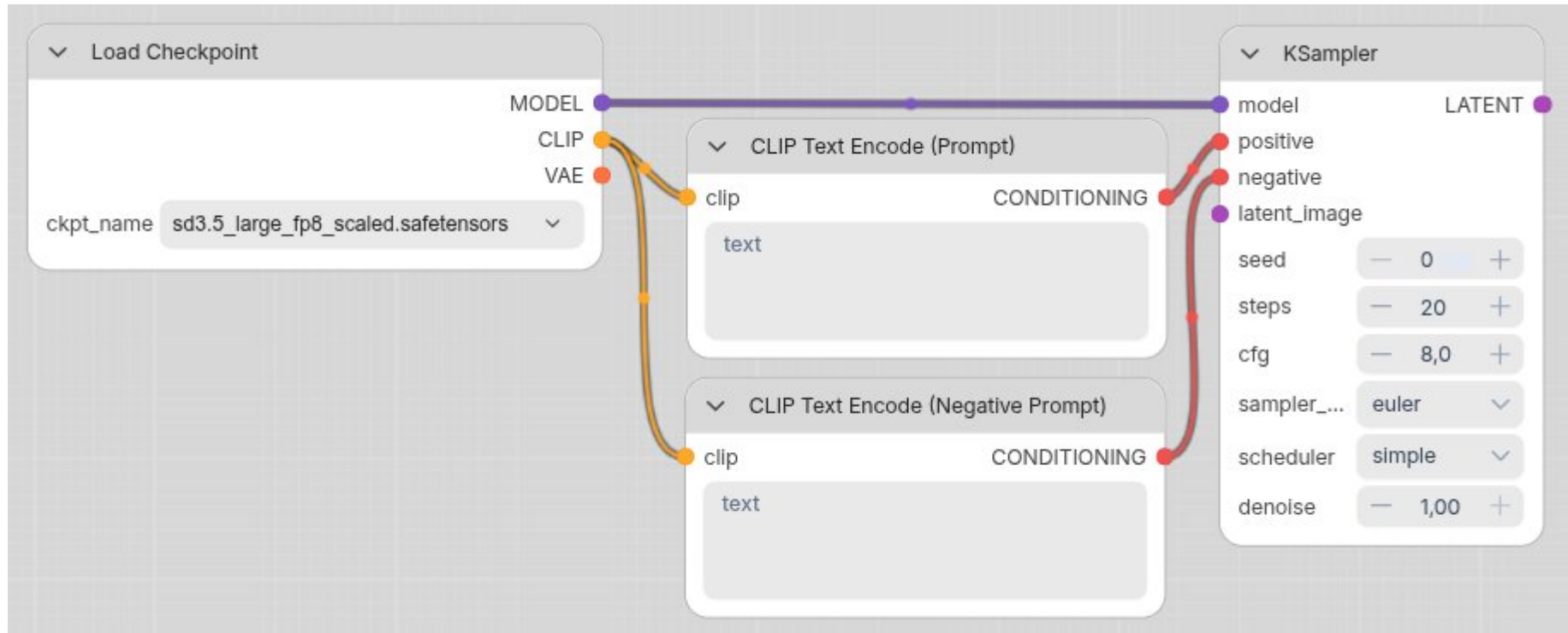
Ein Workflow „from scratch“ - Beispiel (2)

2. Klick und Herausziehen nach rechts auf „Clip“ im gerade erzeugten Node.
3. Im Auswahlfenster „Clip Text Encode (Prompt)“ auswählen. Das ist der Eingabeprompt für das zu erzeugende Bild (positiv).
4. Das gleiche noch mal ab Schritt 2, ebenfalls mit „Clip Text Encode (Prompt)“. Dies soll der Negativ-Prompt werden, also Dinge, die im Bild vermieden werden sollen (NSFW, blurring, overexposed, extra fingers, extra legs, fused fingers, ...).



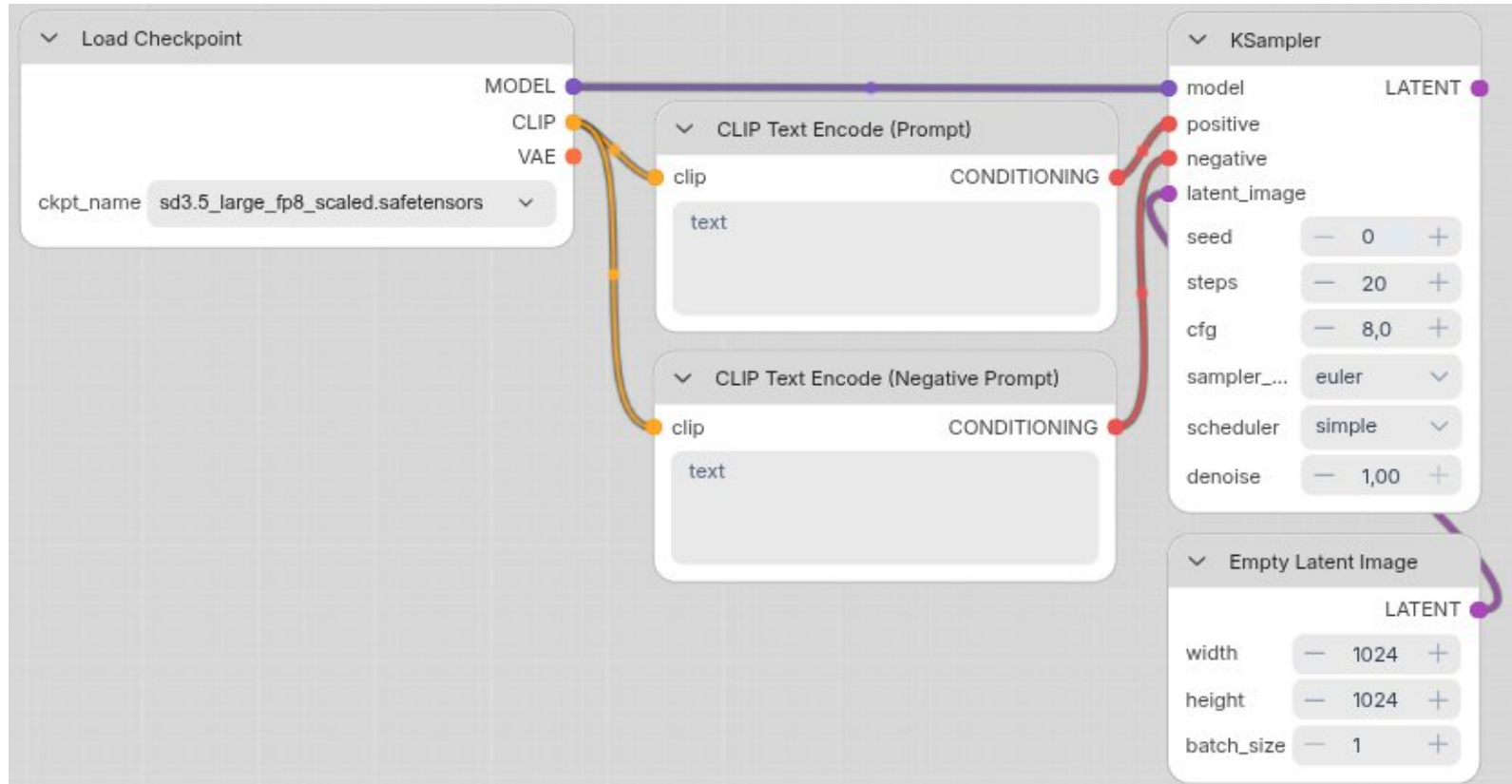
Ein Workflow „from scratch“ - Beispiel (3)

5. Von ersten Clip (Prompt) eine Linie nach rechts ziehen und „K-Sampler“ auswählen.
6. Den zweiten Clip (Negativ-Prompt) auch mit K-Sampler verbinden, bei „negative“.
7. „Load Checkpoint“, Punkt „Model“ mit dem Input „Model“ des „K-Sampler“ verbinden.



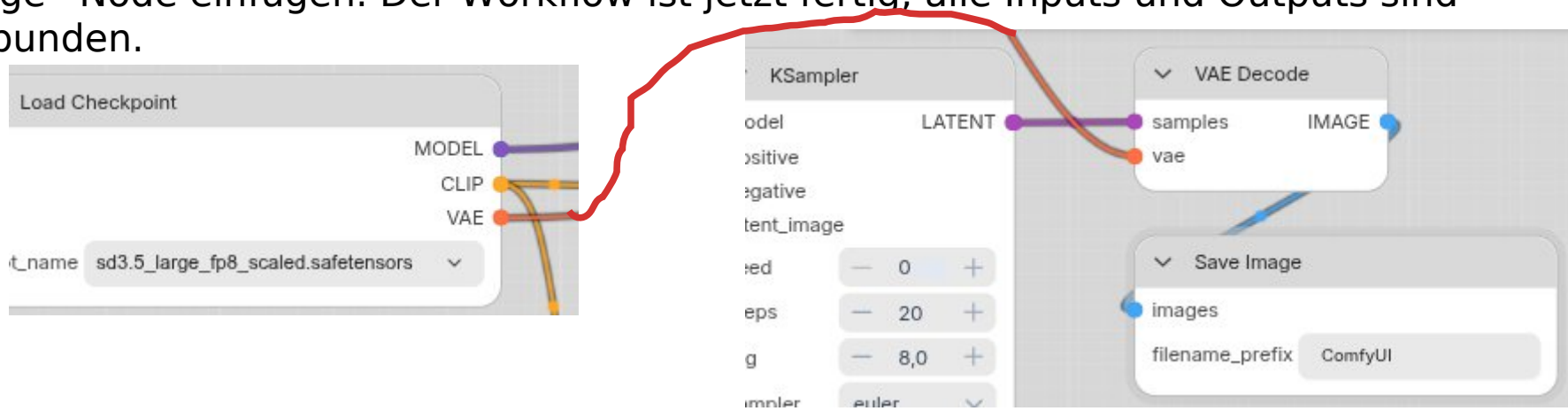
Ein Workflow „from scratch“ - Beispiel (4)

8. Von Input-Punkt „Latent_Image“ des „K-Sampler“ nach links ziehen und ein „Empty Latent Image“ einfügen. Hier kann die Auflösung des Bildes festgelegt werden.



Ein Workflow „from scratch“ - Beispiel (5)

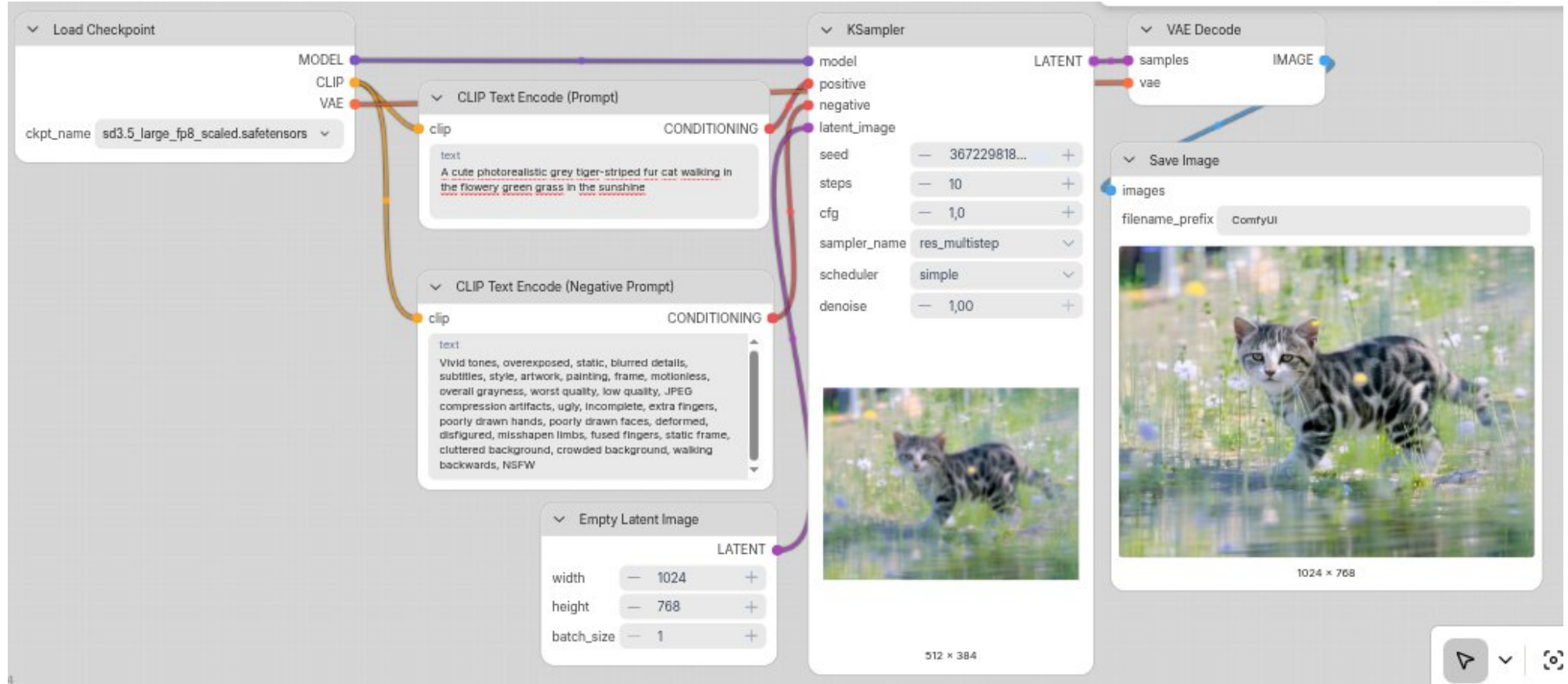
9. Wir sind fast fertig! Vom Output-Punkt „LATENT“ des „K-Sampler“ nach rechts ziehen und „VAE Decode“ auswählen. Dies ist der Transformer vom „Arbeitsbereich“ des Samplers zum „Bildgenerierungs“-Bereich.
10. Den Input-Punkt des „VAE Decode“ noch mit dem VAE-Punkt des „Load-Checkpoint“ Node verbinden.
11. Schließlich von Ausgabe-Punkt „IMAGE“ heraus einen „Preview Image“ oder „Save Image“-Node einfügen. Der Workflow ist jetzt fertig, alle Inputs und Outputs sind verbunden.



Nach Eingabe eines Prompt und (optional) eines Negativ-Prompts kann der Workflow nun gestartet und ein Bild erzeugt werden.

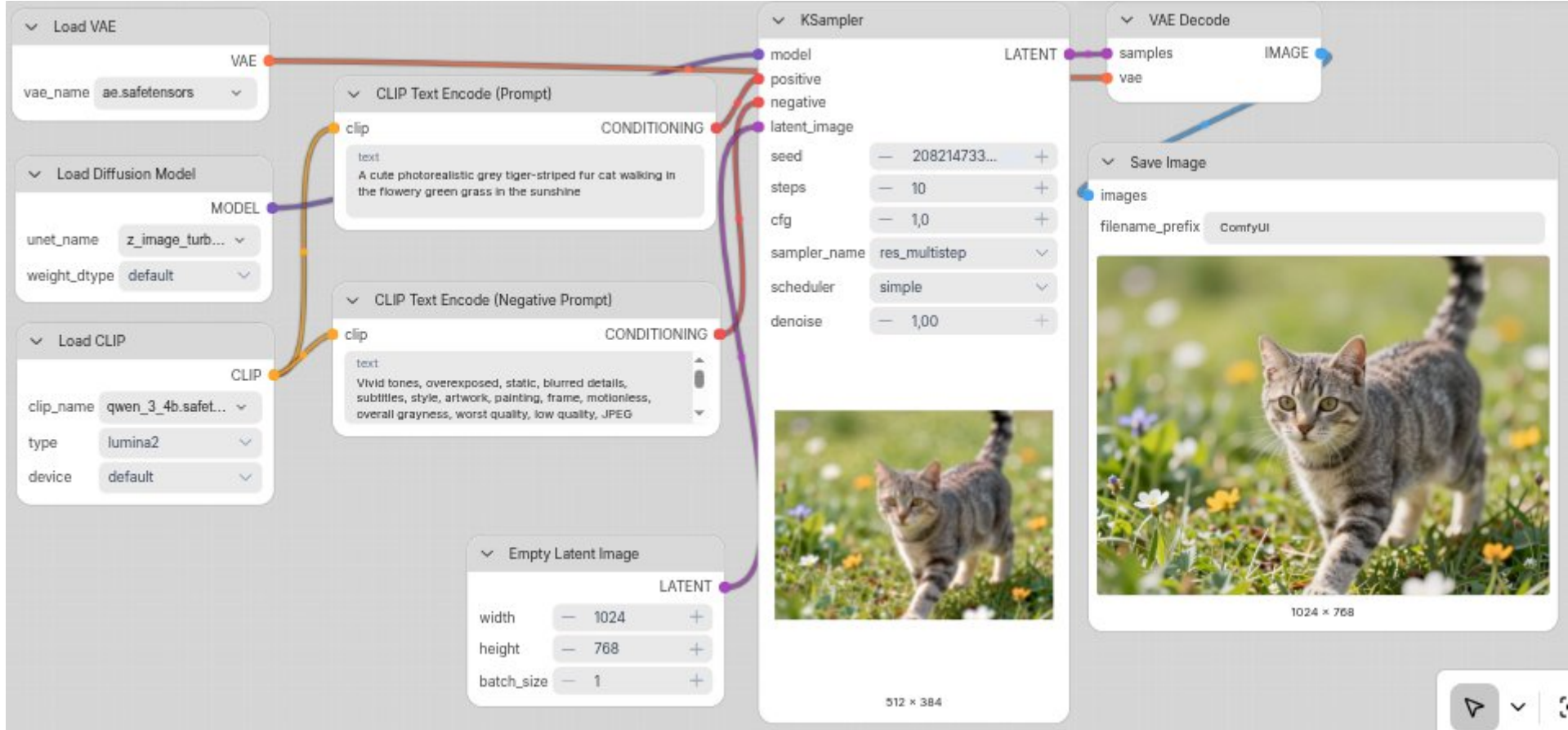
Ein Workflow „from scratch“ - Beispiel (6)

Der fertige Workflow mit angepassten Parametern.



Ein Workflow „from scratch“ - Beispiel (6b)

Hier eine modifizierte Version mit unterschiedlichen Modellen für VAE, Diffusion und Text Encoding, bessere Qualität!.



Wdh: Was braucht man für ComfyUI? Systemvoraussetzungen

Cloud Variante:

- Browser + ein ComfyUI-Konto

Lokale Installation:

- GPU (Nvidia Grafikkarte ≤ 10 Jahre alt, aktuelle AMD GK oder einige INTEL Grafikchipsätze) mit Minimum 4GB VRAM (Grafikspeicher), s.a
→ <https://github.com/comfyanonymous/ComfyUI/wiki/Which-GPU-should-I-buy-for-ComfyUI>
- 16GB RAM oder mehr (für „offloading“ von LLMs und Temporärspeicher)
- Je nach Installationsmethode, funktionierende Installation von Python
- Genug Platz für einige große LLMs (bewegen sich zwischen 400MB und 64GB pro LLM)

Mit der Option `python main.py --cpu` ist auch ein Start mit reiner CPU+RAM Leistung möglich (aber sehr langsam, und wird ggf. nicht von allen Nodes unterstützt).

ENDE

Dieser Vortrag steht unter einer Creative Commons Lizenz



<http://creativecommons.org/licenses/by-nd/4.0/>

Klaus.Knopper@hs-kl.de