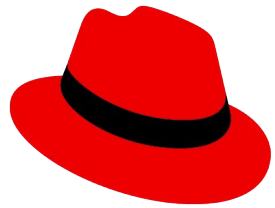


Red Hat  
**Summit**

## **Connect**

Cloud Development Environment  
with Red Hat OpenShift Dev Spaces  
and Personal AI Assistants



**Red Hat**

**Ilya Buziuk**

Principal Software Engineer

**Dr.-Ing. Manuel Hahn**

Associate Principal Specialist Solution Architect

# Cloud Development Environment (CDE)



# Why CDEs are gaining popularity?



Cloud Spreading



Low Latency Networking



Large Codebases / Monorepos



Laptop / Desktop Compute  
Power Plateauing

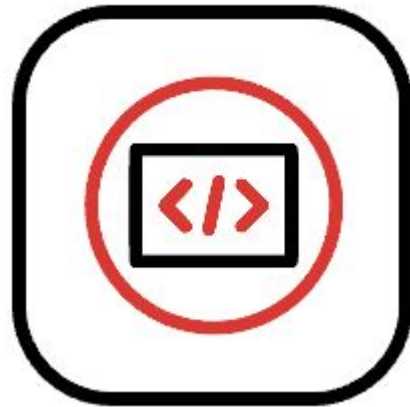


Developer Productivity



Code Leaks

# Red Hat OpenShift Dev Spaces



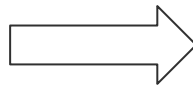
Kubernetes-based Cloud Development  
Environments for Enterprise Teams

# Devfile

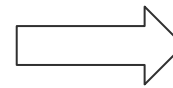
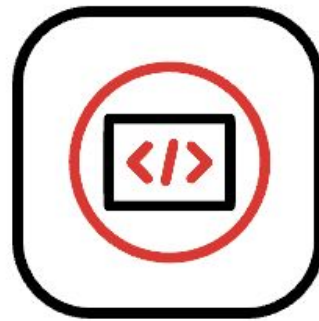


An open standard defining containerized development environments.

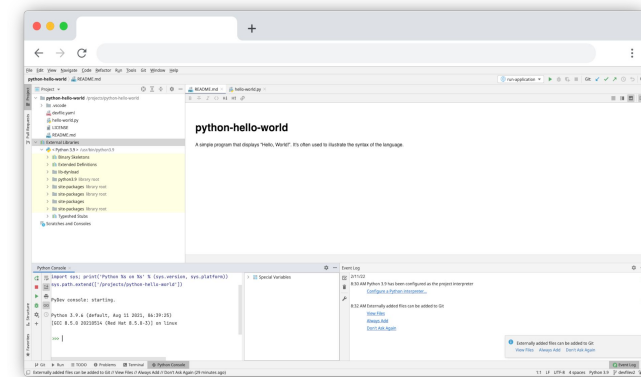
# Devfile



# Dev Spaces



# CDE



```
schemaVersion: 2.3.0
metadata:
  name: sample-java-devfile
components:
  - name: tools
    container:
      image:
        registry.redhat.io/devspaces/udi-rhel8:3.2-26
      memoryLimit: 3Gi
      cpuLimit: 2
      endpoints:
        - exposure: public
          name: my-spring-boot-api
          protocol: https
          targetPort: 8080
      volumeMounts:
        - name: m2
          path: /home/user/.m2
  - name: m2
    volume:
      size: 5G
commands:
  - id: 1-build
    exec:
      component: tools
      workingDir: ${PROJECT_SOURCE}
      commandLine: mvn -DskipTests clean package
```



```
schemaVersion: 2.3.0
metadata:
  name: sample-java-devfile
components:
```

```
- name: tools
```

```
  container:
```

```
    image:
```

```
      registry.redhat.io/devspaces/udi-rhel8:3.2-26
```

```
    memoryLimit: 3Gi
```

```
    cpuLimit: 2
```

```
    endpoints:
```

```
      - exposure: public
```

```
        name: my-spring-boot-api
```

```
        protocol: https
```

```
        targetPort: 8080
```

```
    volumeMounts:
```

```
      - name: m2
```

```
        path: /home/user/.m2
```

```
- name: m2
```

```
  volume:
```

```
    size: 5G
```

```
commands:
```

```
- id: 1-build
```

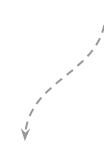
```
  exec:
```

```
    component: tools
```

```
    workingDir: ${PROJECT_SOURCE}
```

```
    commandLine: mvn -DskipTests clean package
```

## Development Image with all the dependencies



## Full control over required resources

```
schemaVersion: 2.3.0
metadata:
  name: sample-java-devfile
components:
- name: tools
  container:
    image:
      registry.redhat.io/devspaces/udi-rhel8:3.2-26
    memoryLimit: 3Gi
    cpuLimit: 2
    endpoints:
      - exposure: public
        name: my-spring-boot-api
        protocol: https
        targetPort: 8080
    volumeMounts:
      - name: m2
        path: /home/user/.m2
- name: m2
  volume:
    size: 5G
commands:
- id: 1-build
  exec:
    component: tools
    workingDir: ${PROJECT_SOURCE}
    commandLine: mvn -DskipTests clean package
```

```
schemaVersion: 2.3.0
metadata:
  name: sample-java-devfile
components:
- name: tools
  container:
    image:
      registry.redhat.io/devspaces/udi-rhel8:3.2-26
    memoryLimit: 3Gi
    cpuLimit: 2
    endpoints:
      - exposure: public
        name: my-spring-boot-api
        protocol: https
        targetPort: 8080
    volumeMounts:
      - name: m2
        path: /home/user/.m2
- name: m2
  volume:
    size: 5G
commands:
- id: 1-build
  exec:
    component: tools
    workingDir: ${PROJECT_SOURCE}
    commandLine: mvn -DskipTests clean package
```

**Expose application for testing  
and collaboration purposes**

## Persist maven/npm/python repositories across workspace restarts

```
schemaVersion: 2.3.0
metadata:
  name: sample-java-devfile
components:
- name: tools
  container:
    image:
      registry.redhat.io/devspaces/udi-rhel8:3.2-26
    memoryLimit: 3Gi
    cpuLimit: 2
    endpoints:
      - exposure: public
        name: my-spring-boot-api
        protocol: https
        targetPort: 8080
      - name: m2
        path: /home/user/.m2
    volumeMounts:
      - name: m2
        volume:
          size: 5G
  commands:
    - id: 1-build
      exec:
        component: tools
        workingDir: ${PROJECT_SOURCE}
        commandLine: mvn -DskipTests clean package
```

```
schemaVersion: 2.3.0
metadata:
  name: sample-java-devfile
components:
- name: tools
  container:
    image:
      registry.redhat.io/devspaces/udi-rhel8:3.2-26
    memoryLimit: 3Gi
    cpuLimit: 2
    endpoints:
      - exposure: public
        name: my-spring-boot-api
        protocol: https
        targetPort: 8080
    volumeMounts:
      - name: m2
        path: /home/user/.m2
- name: m2
  volume:
    size: 5G
commands:
- id: 1-build
  exec:
    component: tools
    workingDir: ${PROJECT_SOURCE}
    commandLine: mvn -DskipTests clean package
```

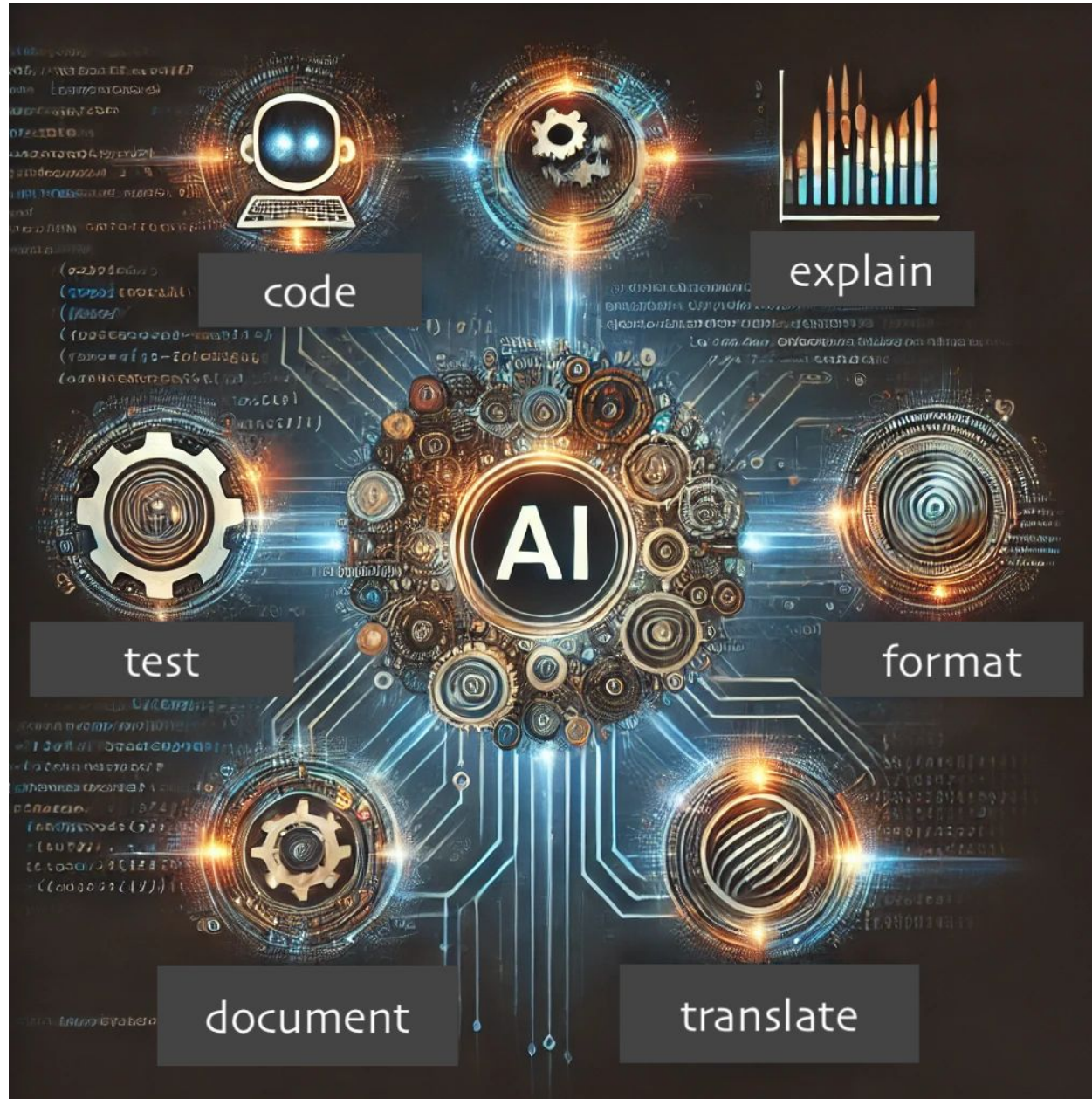
## Add shortcuts for favourites commands



Demo time...

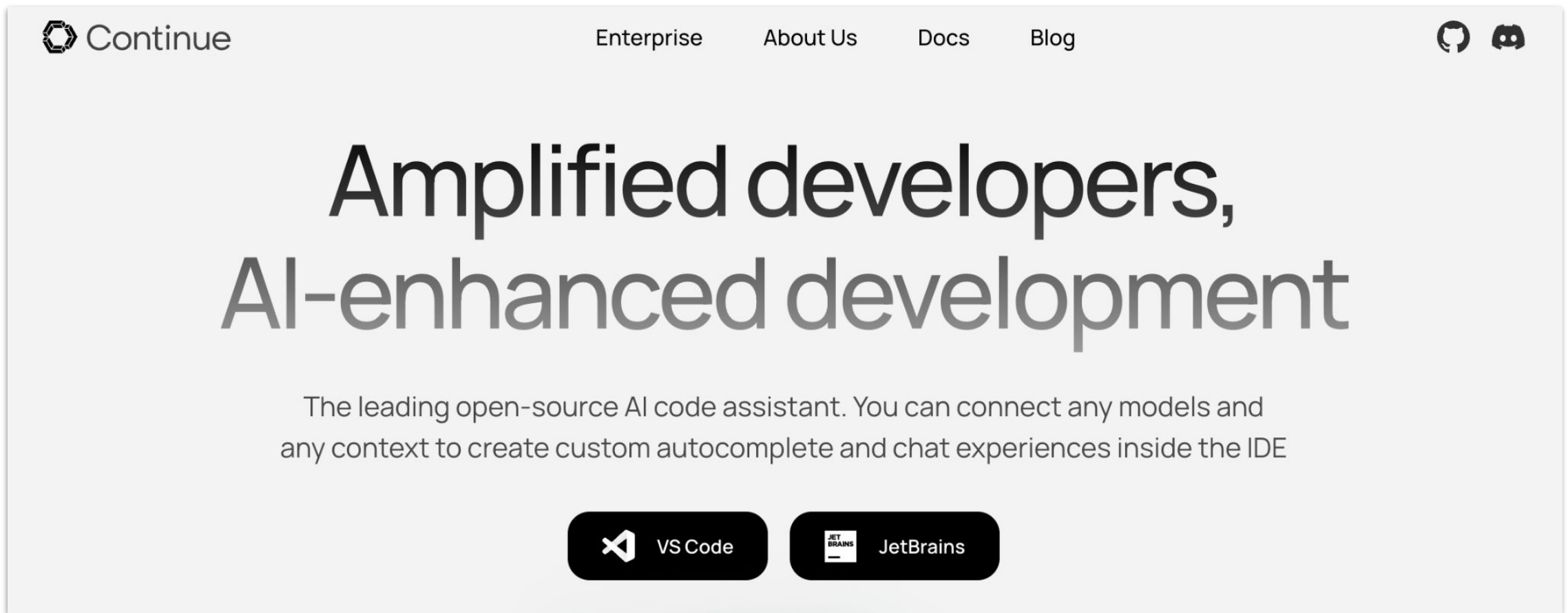
# Personal AI Assistant

# Why Personal AI Assistants?





# The Open-Source Personal AI Assistant: Continue



The screenshot shows the homepage of the Continue project. At the top left is the Continue logo, a hexagon with a stylized 'C' inside. To its right are navigation links: 'Enterprise', 'About Us', 'Docs', and 'Blog'. In the top right corner are icons for GitHub and Discord. The main heading is 'Amplified developers, AI-enhanced development' in a large, bold, sans-serif font. Below this is a sub-heading: 'The leading open-source AI code assistant. You can connect any models and any context to create custom autocomplete and chat experiences inside the IDE'. At the bottom of the main content area are two dark buttons with white text and icons: 'VS Code' with the VS Code logo and 'JetBrains' with the JetBrains logo.

<https://www.continue.dev> | <https://github.com/continuedev/continue>

Demo time...

# Large Language Model (LLM)

# Where to locate LLMs in the AI jungle?

## Predictive AI

- Predicts or forecasts outcomes based on historical data
- Training a model **from scratch**
- Needs **moderate** amounts of data



## Generative AI

- Generates new, original content, based on trained data
- Using a **foundational model**
- Needs **massive** amounts of data
- **LLMs** are foundation models for natural language processing

Hi there, XYZ Insurance Company. I hope this email is okay and finds you okay. I had an accident, and I'm not exactly sure how to go about this, but I think it's something to do with a car accident claim, and my policy number is 12345. I'd like to know what to do next.

Okay, so I'm not sure what to do next. I'd like to know what to do next.

Accident Details: I had an accident involving my car on Maple Avenue, near Smith Street, on approximately 2:30 PM. I sustained minor injuries, but the damage to my car was significant. I have the contact information for the other party's insurance company, and I'd like to know what to do next.

Date and Time: The accident occurred on approximately 2:30 PM on approximately 2:30 PM.

Location: The accident occurred on Maple Avenue, near Smith Street.

The Accident: I was driving my car when I was involved in an accident with another car. The other car was driving through a red light.

Weather Conditions: The weather was clear and a bit warm.

Traffic Conditions: Traffic was heavy at the time of the accident.

Car Details: My car is a Ford Escape.

What Happened: I was driving my car when I was involved in an accident with another car. The other car was driving through a red light.

Injuries: I sustained minor injuries, but the damage to my car was significant.

Witnesses: I have the contact information for the other party's insurance company.

Witness Statement: I was driving my car when I was involved in an accident with another car. The other car was driving through a red light.

**Summary:** The text is an email from John Smith to XYZ Insurance Company regarding an accident involving his car. The accident occurred approximately 2:30 PM at the intersection of Maple Avenue, near Smith Street. John sustained minor injuries, but the damage to his car was significant. He has the contact information for the other party's insurance company and is seeking assistance in processing the claim.

**human-readable summary**

**Sentiment:** The sentiment of the person writing this text appears to be calm, assertive, and cooperative.

**Sentiment:** The sentiment expressed in this text seems to be assertive and frustrated.

# How to choose the right LLM?

## What to consider?

- Dataset (e.g. quality?)
- Licensing (e.g. permissive, IP indemnification?)
- Deployment options (on-prem or online-only?)
- Resource needs (e.g. RAM, storage, GPU needed?)
- Customization options (e.g. fine-tuning, RAG, LAB?)
- ...

## The Stack

6 TB of permissive code data

@BigCodeProject  
<https://www.bigcode-project.org/>  
contact@bigcode-project.org

### Dataset Collection

GH Archive → query → 220 M repo names → git clone → Raw dataset (137 M repos, 52 B files, 102 TB of data) → selecting file extensions → 69 TB of data → license filtering → 6.4 TB of data → near-deduplication → 2.9 TB of data

Find the filtered and deduplicated datasets at: [www hf.co/bigcode](http://www hf.co/bigcode)

### Licensing + Governance

Raw dataset: No license, MIT, Apache 2.0, Others

Permissive: MIT, Apache 2.0, BSD-3-Clause, Others

Opt-out: If users would like to exclude their code from the corpus we have an opt-out mechanism. Visit: <https://www.bigcode-project.org/docs/about/the-stack/>

Permissive license distribution of licenses used to filter the dataset:  
MIT (67.7%) | Apache-2.0 (19.1%) | BSD-3-Clause (3.9%) | Unlicense (2.0%) | CC0-1.0 (1.5%) | BSD-2-Clause (1.2%) | CC-BY-4.0 (1.1%) | CC-BY-3.0 (0.7%) | 0BSD (0.4%) | RSA-MD (0.3%) | WTFPL (0.2%) | MIT-0 (0.2%) | Others (166) (2.2%)

### Programming Languages

Language	File Size (approx)	Number of Files (approx)
python	1 TB	100 M
java	100 GB	10 M
javascript	100 GB	10 M
cpp	100 GB	10 M
rust	100 GB	10 M
typescript	100 GB	10 M
scala	100 GB	10 M
go	100 GB	10 M
ruby	100 GB	10 M
php	100 GB	10 M
yaml	100 GB	10 M
xml	100 GB	10 M
json	100 GB	10 M
markdown	100 GB	10 M
css	100 GB	10 M
text	100 GB	10 M
yaml	100 GB	10 M
xml	100 GB	10 M
json	100 GB	10 M
markdown	100 GB	10 M
css	100 GB	10 M
text	100 GB	10 M
others	100 GB	10 M

### Evaluation

We trained several GPT-2 models (350M parameters) on different parts of the dataset both with and without near-deduplication. The models trained on the Python subset of The Stack performed on par with CodeX and CodeGen of similar size when using near-deduplication.

Dataset	Filtering	pass@1	pass@10	pass@100
CodeX (300M)	unknown	13.17	20.17	36.27
CodeGen (350M)	unknown	12.76	23.11	35.19
Python all-license	None	13.11	21.77	36.67
	Near-dedup.	17.34	27.64	45.52
Python permissive-license	None	10.99	15.94	27.21
	Near-dedup.	12.89	22.26	36.01

\*results obtained with The Stack v1.0

### Usage

```
!pip install datasets
from datasets import load_dataset

# full dataset (6TB of data)
ds = load_dataset('bigcode/the-stack', split='train')

# near-deduplicated dataset (1.5TB of data)
ds = load_dataset('bigcode/the-stack-deDup', split='train')

# specific language (e.g. Dockerfiles)
ds = load_dataset('bigcode/the-stack', data_dir='data/dockerfile', split='train')

# dataset streaming (will only download the data as needed)
ds = load_dataset('bigcode/the-stack', streaming=True, split='train')
for sample in ds: print(sample['content'])
```

For more info visit about the Datasets library visit: [hf.co/docs/datasets](https://hf.co/docs/datasets)

### Dataset Trivia

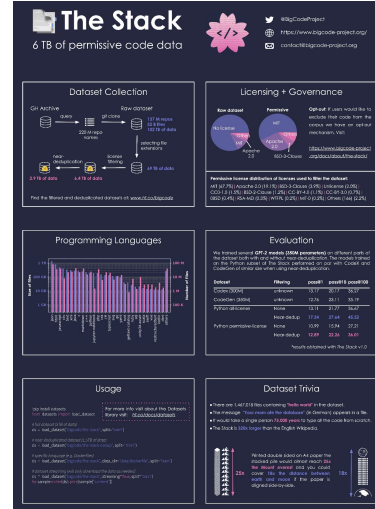
- There are 1,467,018 files containing "hello world" in the dataset.
- The message "Your mom ate the database" (in German) appears in a file.
- It would take a single person 75,000 years to type all the code from scratch.
- The Stack is 320x larger than the English Wikipedia.

Printed double sided on A4 paper the stacked pile would almost reach 25x the Mount everest and you could cover 18x the distance between earth and moon if the paper is aligned side-by-side.

# How to choose the right LLM?

## What to consider?

- Dataset (e.g. quality?)
- Licensing (e.g. permissive, IP indemnification?)
- Deployment options (on-prem or online-only?)
- Resource needs (e.g. RAM, storage, GPU needed?)
- Customization options (e.g. fine-tuning, RAG, LAB?)
- ...



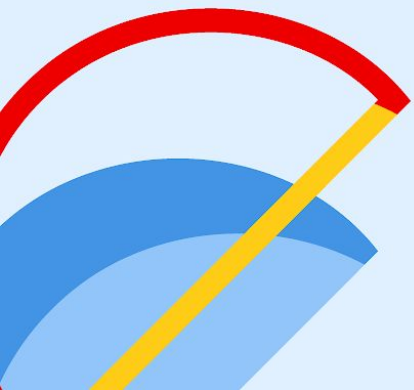
## Why open source models?

- Transparency  
No black box, training data, community-driven development, ...
- Control  
Stability & consistency through deliberate LLM changes, ...
- Flexibility  
On-prem or cloud, customization like fine-tuning, not relying on a single provider, ...
- ...

Demo time...

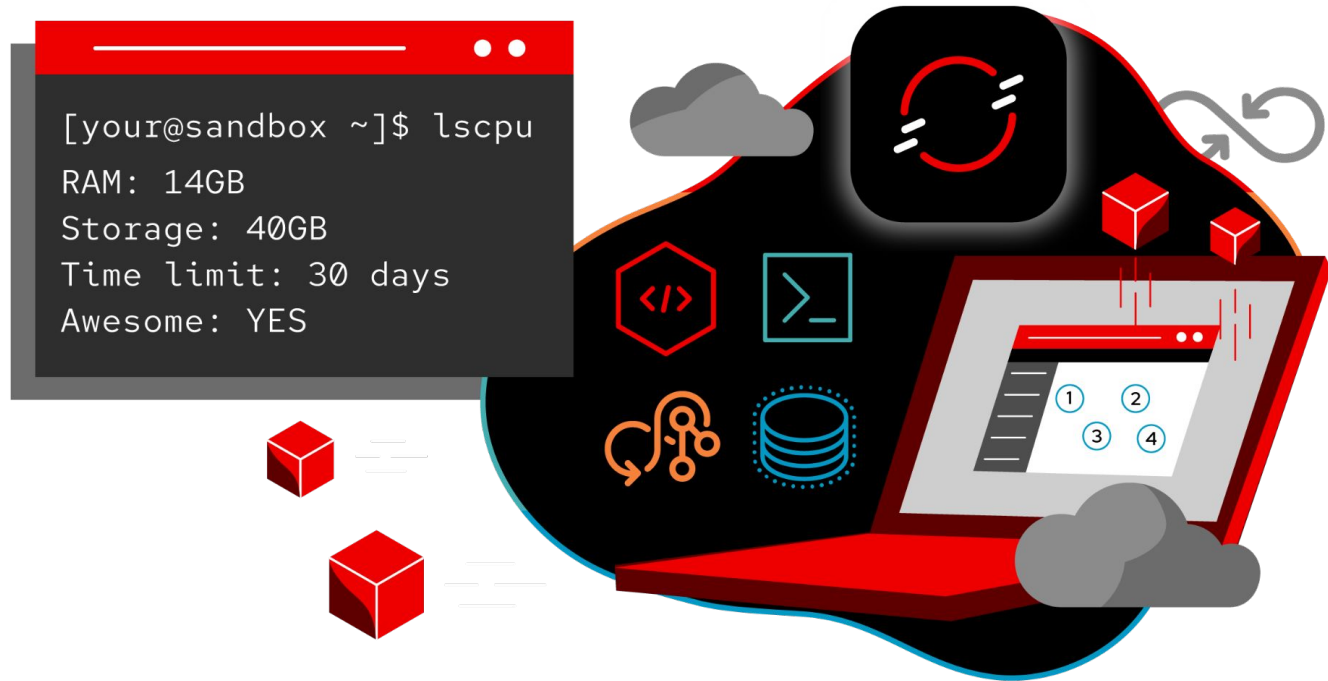


# How to get started?





# Developer Sandbox



# Integrate a private AI coding assistant into your CDE using Ollama, Continue, and OpenShift Dev Spaces

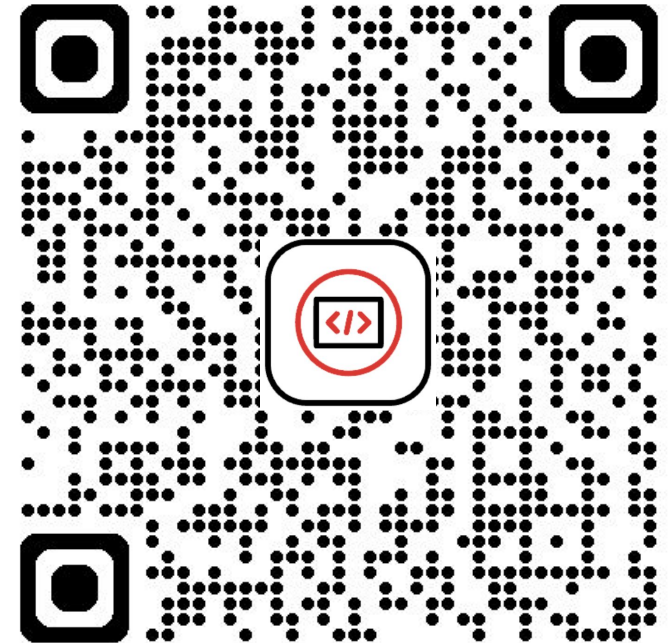
Level up your cloud development environment

August 12, 2024 | [Ilya Buziuk](#), [Manuel Hahn](#)

Related topics: [AI/ML](#), [Containers](#), [Developer Productivity](#), [Developer Tools](#)

Related products: [Developer Sandbox](#), [Developer Tools](#), [Red Hat OpenShift Dev Spaces](#)

Share: [Twitter](#) [Facebook](#) [LinkedIn](#) [Email](#)



Questions?

Session: 14:40 - 15:10



# Jetzt Session bewerten!

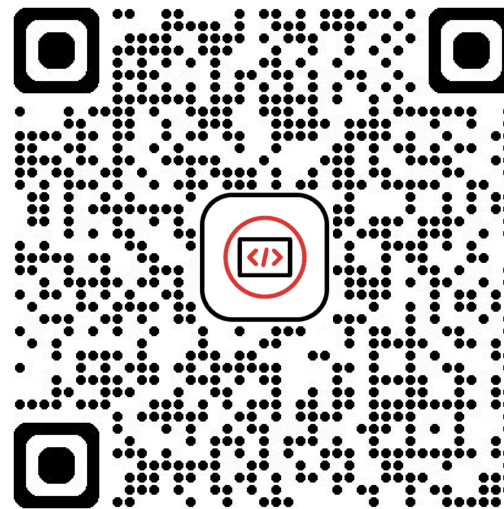
Einfach QR-Code  
scannen, Session  
wählen und bewerten.  
**Vielen Dank!**

[red.ht/rhsc24-de-s4](https://red.ht/rhsc24-de-s4)

Red Hat  
**Summit**

**Connect**

**Thank you**



[linkedin.com/company/red-hat](https://www.linkedin.com/company/red-hat)



[facebook.com/redhatinc](https://www.facebook.com/redhatinc)



[youtube.com/user/RedHatVideos](https://www.youtube.com/user/RedHatVideos)



[twitter.com/RedHat](https://twitter.com/RedHat)