Local LLM Hosting at TU Delft

TULIP, Use Cases, and Next Steps

First alignment workshop

June 22, 2026

Why are we here?

Concrete questions we keep hearing on campus

> "Can we have an endpoint that students can use in my lecture of 400 students for learning to code with LLMs?"

> "Can we use LLM infra for annotating data in my research, because OpenAI is too expensive?"

> "I don't want to send resumes of PhD candidates to a commercial LLM — data privacy."

The pattern

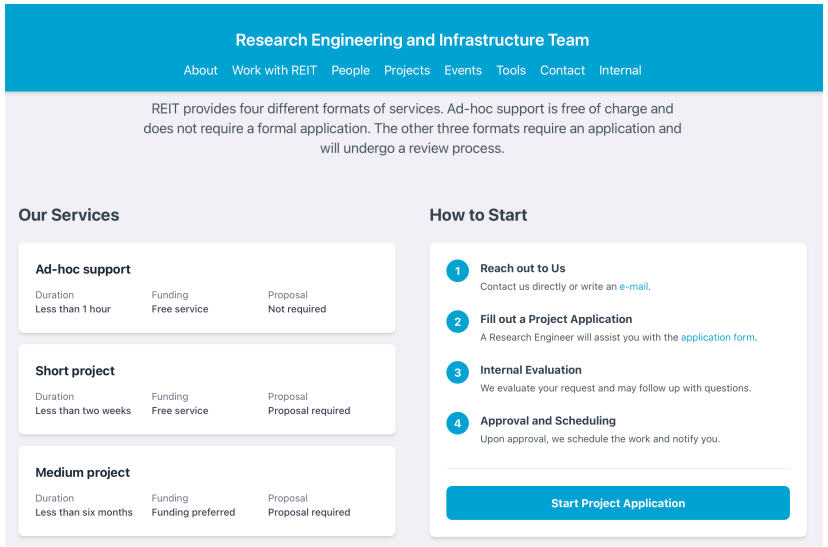
These aren't asks for a chatbot. They are asks for shared, governed, affordable inference that an institution can stand behind.

Who are we?

REIT — Research Engineering & IT

A small team in **REIT** prototyping **TULIP** as institutional LLM serving for TU Delft.

Scan or visit → reit.tudelft.nl



TU Delft LLM Infrastructure Platform

Program

Time	Session
11:00–11:05	Opening
11:05–11:20	Vision & what we've been building
11:20–12:00	**Breakout 1:** research and education use cases
12:00–12:25	**Breakout 2:** current tools, dependencies, and unmet needs
12:25–12:35	*Grab food*
12:35–13:00	Sharing, reflections & wrap-up

What we want from today

Your concrete needs → our next-phase priorities.

LLMs are becoming infrastructure

Researchers and educators are already using LLMs through APIs

Common patterns today:

- chat interfaces
- coding assistants (Copilot, Cursor, Claude Code)
- RAG over internal documents
- agentic workflows
- research prototypes and educational tools

Most of these applications do not run models directly. They consume **model endpoints** provided by external services.

LLM serving ≠ LLM experimentation

Two different questions, two different infrastructures

LLM experimentation	LLM serving
Fine-tuning models	Reliable inference
Benchmarking	Stable endpoints
Trying new architectures	Multi-user access
One researcher	Hundreds of users
GPU allocation	API management
Accuracy focus	Reliability focus

The framing

- **Research asks:** can we build a *better* model?
- **Serving asks:** can hundreds of users *reliably* use a model?

Why universities need LLM serving — Accessibility

Shared access to AI capabilities

Where shared access matters

- students without paid subscriptions
- teaching at scale
- consistent model access across cohorts
- shared infrastructure for groups
- reproducible research environments

Concrete examples

- course assistants
- coding support in classrooms
- shared research prototypes
- student projects on stable endpoints

The accessibility argument

Today the *quality* of a student's or researcher's AI access often depends on whether they can pay for a personal subscription. A serving layer flattens this.

Why universities need LLM serving — Governance

Institutional control over AI usage

Where governance matters

- data remains within TU Delft
- GDPR compliance
- controlled access and authentication
- auditable usage
- reduced dependence on external providers

Concrete examples

- student assessments and feedback
- interviews and human-subject data
- internal and unpublished documents
- sensitive research data

The governance argument

Without institutional serving, sensitive workflows fall back to shadow IT (personal accounts on external APIs) or no AI at all.

Why universities need LLM serving — Institutional capability

Endpoints that outlive grants, subscriptions, and projects

Where capability matters

- universities increasingly *build* AI-powered systems
- RAG platforms over institutional content
- research software and pipelines
- educational tools
- agentic workflows

Concrete examples

- shared research tools
- durable educational infrastructure
- prototypes that survive personnel changes
- platforms used across departments

The strategic point

An LLM serving layer isn't just a convenience for individual researchers — it is a piece of institutional infrastructure that the university needs to own, the same way it owns DAIC, NetID, or DelftBlue.

Our vision

A large-scale academic LLM serving platform for TU Delft

A platform that can support

- **Research** — annotation pipelines, experimentation, reproducibility
- **Education** — course-scale endpoints, classroom tools, student access at scale
- **Data governance** — sensitive data, GDPR, institutional control

Shaped *by* the institution, *for* the institution.

What we want from you today

Concrete use cases, dependencies, and unmet needs → we turn them into the next phase of the platform.

TU Delft LLM Infrastructure Platform

Our vision

A large-scale academic LLM serving platform for TU Delft

A platform that can support

- **Research** — annotation pipelines, experimentation, reproducibility
- **Education** — course-scale endpoints, classroom tools, student access at scale
- **Data governance** — sensitive data, GDPR, institutional control

Shaped *by* the institution, *for* the institution.

But today — we need your input

This workshop is to gauge your short-term needs so that we build from your requirements, not from our assumptions.

The deal:

- you tell us what you need;
- we build with those priorities;
- you have skin in the decisions that affect you.

Two **breakout sessions** are the mechanism.

What we want from you today

Concrete use cases, dependencies, and unmet needs → we turn them into the next phase of the platform.

TU Delft LLM Infrastructure Platform

What have we been cooking?

From conversations to a working prototype

March 2026 — workshop

Towards democratised academic LLM hosting

A cross-faculty conversation on what local LLM hosting should look like at TU Delft.

Since then

- gathered concrete use cases
- agreed on shared constraints: governance, access, integration
- aligned across REIT and ICT

What have we been cooking?

From conversations to a working prototype

March 2026 — workshop

Towards democratised academic LLM hosting

A cross-faculty conversation on what local LLM hosting should look like at TU Delft.

Since then

- gathered concrete use cases
- agreed on shared constraints: governance, access, integration
- aligned across REIT and ICT

Now — an experimental serving platform

We are building **TULIP** — TU Delft's experimental LLM inference platform — as a working prototype of that idea.

The rest of this section is what TULIP is and where it stands today.

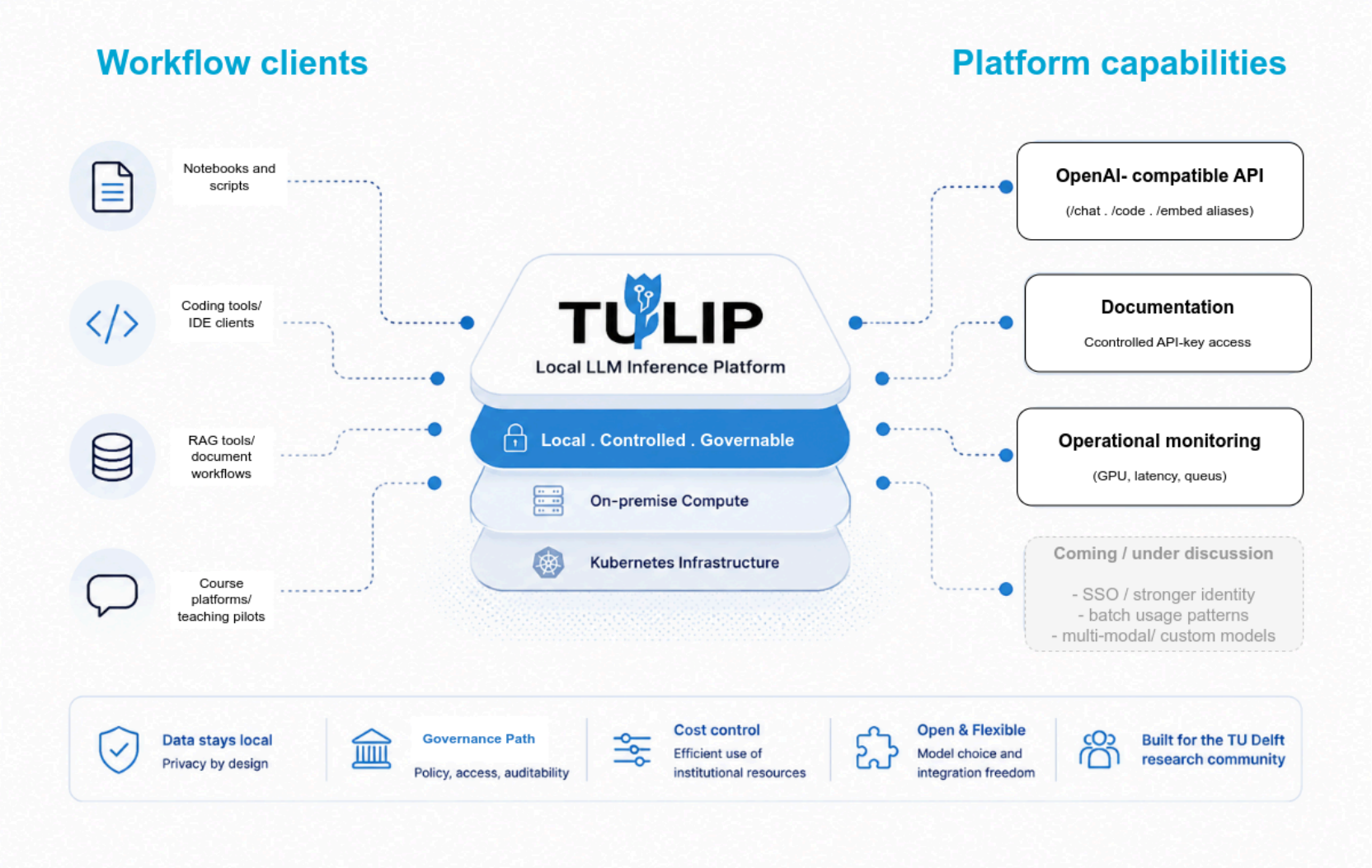
TU Delft LLM Infrastructure Platform

TULIP: *current state and future directions*

~11:15–11:20

What is TULIP?

TU Delft LLM Inference Platform



TU Delft LLM Infrastructure Platform

What is available now?

Capability	Current status
Models	Chat, code, embeddings
Access	Controlled pilot users
Interface	*OpenAI-compatible* API endpoints
Integration	Scripts, notebooks, IDE tools, compatible clients, and *agentic workflows/tools*
Operations	GPU, latency, queue, and service monitoring
Scale	Small platform: 1 server, 3× L40S GPUs

Current status

TULIP is a controlled pilot-scale platform, not yet a full campus-wide production service.

With thanks to TU Delft ICT

The server node and L40S GPUs that TULIP runs on are generously hosted by TU Delft ICT. The platform

TU Delft LLM Infrastructure Platform

Example use patterns

Use case	Example workflow	Implied requirements

Example use patterns

Use case	Example workflow	Implied requirements
Sensitive text analysis	Interviews, internal documents, unpublished material, assessments & feedback	Data locality and clearer governance

Example use patterns

Use case	Example workflow	Implied requirements
Sensitive text analysis	Interviews, internal documents, unpublished material, assessments & feedback	Data locality and clearer governance
Education	Stable endpoints for courses and student projects	Predictability, fairness, no individual SaaS accounts

Example use patterns

Use case	Example workflow	Implied requirements
Sensitive text analysis	Interviews, internal documents, unpublished material, assessments & feedback	Data locality and clearer governance
Education	Stable endpoints for courses and student projects	Predictability, fairness, no individual SaaS accounts
Coding workflows	IDE clients and coding-agent experiments	Institutional API, observability, cost visibility

Example use patterns

Use case	Example workflow	Implied requirements
Sensitive text analysis	Interviews, internal documents, unpublished material, assessments & feedback	Data locality and clearer governance
Education	Stable endpoints for courses and student projects	Predictability, fairness, no individual SaaS accounts
Coding workflows	IDE clients and coding-agent experiments	Institutional API, observability, cost visibility
RAG / retrieval	Embeddings over internal document collections	Reproducibility and local control

Example use patterns

Use case	Example workflow	Implied requirements
Sensitive text analysis	Interviews, internal documents, unpublished material, assessments & feedback	Data locality and clearer governance
Education	Stable endpoints for courses and student projects	Predictability, fairness, no individual SaaS accounts
Coding workflows	IDE clients and coding-agent experiments	Institutional API, observability, cost visibility
RAG / retrieval	Embeddings over internal document collections	Reproducibility and local control
Research-support tools	Integration into platforms, demos, and prototypes	Shared infrastructure instead of fragmented deployments

How we progress and what we aim for today

We aim to capture:

- your needs, expectations, and constraints;
- examples that are as specific and concrete as possible;
- where you see room for commitment, contribution, or collaboration.

Progression

We will move from **use cases → current practice → priorities and collaboration**.

Guided discussion: Research and Education Use Cases

11:20–12:00

Research and education use cases

- What LLM-based workflows are you using or planning?
- How could AI support coding, debugging, software design, feedback, scientific writing (grants, papers)?
- How could AI support tutoring, assessments, exercises, course design, feedback … etc?

Think of

- who the users/developers are;
- what model or capability is needed;
- what the soft and hard needs/requirements are, e.g. endpoint, tool, or support;
- whether the need is exploratory, near-term, or urgent.

Guided discussion: Current tools, dependencies, and unmet needs

12:00–12:25

Current tools, dependencies, and gaps

- What are you using? Copilot, Claude Code, Cursor, OpenRouter, Hugging Face Spaces, self-hosted models / collaborators' APIs, something else?
- Are there paid external services or recurring API costs?
- What makes current workflows hard, risky, expensive, unsuitable, or difficult to support?

Think of

- which tools or services you actually rely on (paid or free);
- what data you send to them, and what data you would never send to them;
- which workflows are blocked, fragile, expensive, or hard to support;
- what would change in your work if those dependencies disappeared tomorrow.

Sharing and Reflections:
Priorities and next steps

12:35–13:00

Group report-back

Briefly share:

- 1–2 concrete use cases;
- current tools or dependencies;
- main unmet needs or blockers;
- possible priorities, commitments, or collaborations.

After today

We will consolidate the notes into use cases, dependencies, unmet needs, and next steps for TULIP.

Next steps

What happens after today

1. Report & summarise

We consolidate what we heard today into:

- a use-case inventory
- common dependencies
- unmet needs and gaps

The throughline

Today: your needs → soon: a wider scan → next: a shared plan.

Next steps

What happens after today

1. Report & summarise

We consolidate what we heard today into:

- a use-case inventory
- common dependencies
- unmet needs and gaps

2. Widen the requirements net

Based on these findings, we:

- approach more groups across faculties
- collect broader requirements
- map them against TULIP's roadmap

The throughline

Today: your needs → soon: a wider scan → next: a shared plan.

Next steps

What happens after today

1. Report & summarise

We consolidate what we heard today into:

- a use-case inventory
- common dependencies
- unmet needs and gaps

2. Widen the requirements net

Based on these findings, we:

- approach more groups across faculties
- collect broader requirements
- map them against TULIP's roadmap

3. Follow-up workshop

When the demand picture is clearer, we hold a second workshop:

"What we want from you" — concrete asks back to you, scaled to the demand and needs we've heard.

The throughline

Today: your needs → soon: a wider scan → next: a shared plan.

Thank you!

TU Delft LLM Infrastructure Platform