

Jitsi Meet

Ein Erfahrungsbericht

Aleksander Paravac

[<aleksander.paravac@uni-wuerzburg.de>](mailto:aleksander.paravac@uni-wuerzburg.de)

Übersicht

- Was ist Jitsi?
- Einsatz an der Uni Würzburg

Was ist Jitsi?

- Ursprünglich ein auf Java basierender SIP/Instant-Messaging-Client
- Basierend auf XMPP
- Nun: “Jitsi Meet” Videokonferenzplattform
 - Basierend auf WebRTC
 - Sammlung diverser Softwarekomponenten
 - Freie Software (Apache Lizenz)
 - Desktop Clients (Electron basierend)
 - Webbasierter Client
 - Smartphone Clients

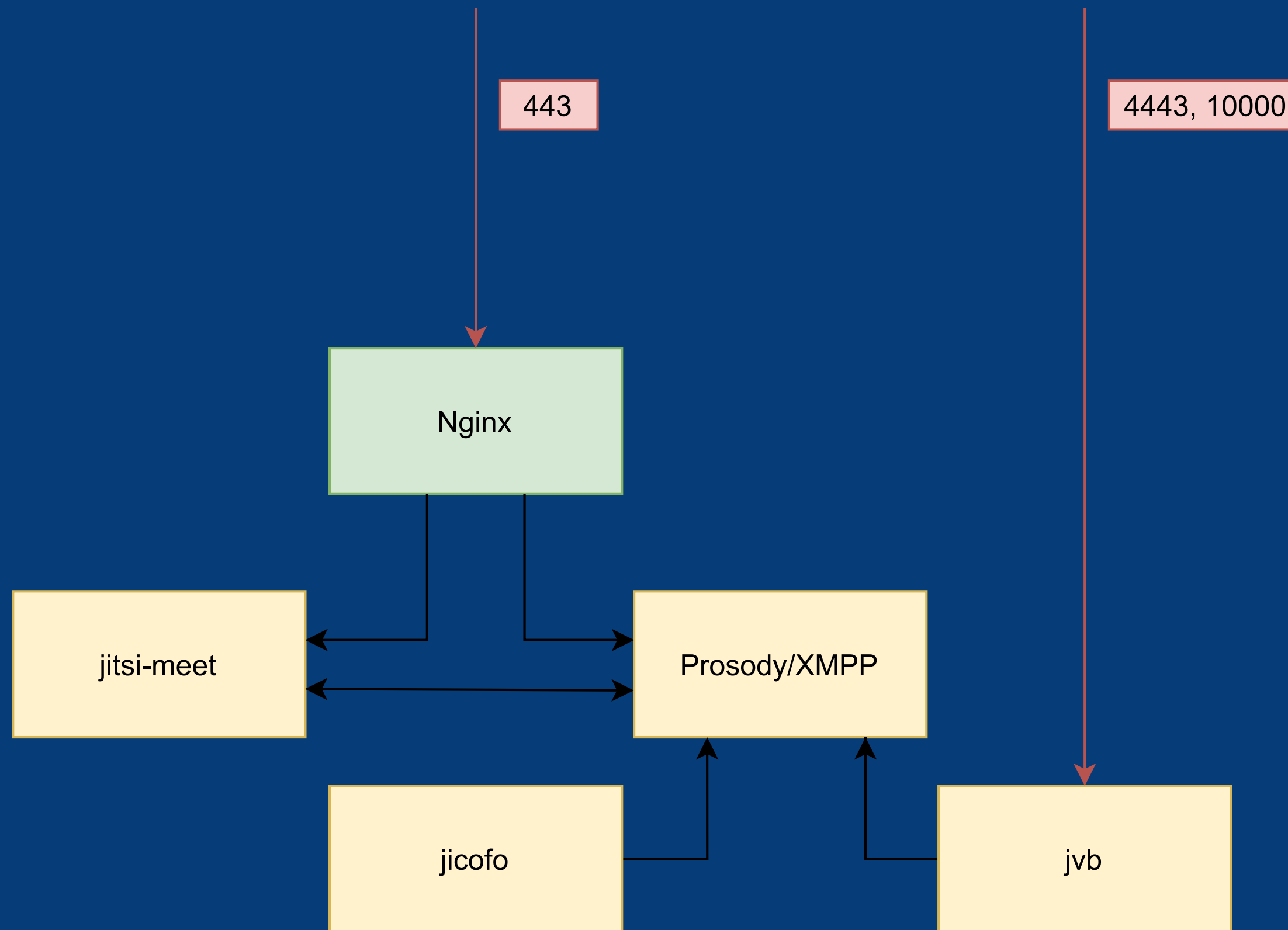
Architektur I

Jitsi comprises a collection of projects:

- **Jitsi Meet** - WebRTC compatible JavaScript application that uses Jitsi Videobridge to provide high quality, scalable video conferences
- **Jitsi Videobridge (jvb)** - WebRTC compatible server designed to route video streams amongst participants in a conference
- **Jitsi Conference Focus (jicofo)** - server-side focus component used in Jitsi Meet conferences that manages media sessions between each of the participants and the videobridge
- **Jitsi Gateway to SIP (jigasi)** - server-side application that allows regular SIP clients to join Jitsi Meet conferences
- **Jibri** - set of tools for recording and/or streaming a Jitsi Meet conference that works by launching a Chrome instance rendered in a virtual framebuffer and capturing and encoding the output with ffmpeg

Übernommen von: <https://jitsi.github.io/handbook/docs/architecture>

Architektur II



Features

- Live Stream in Richtung Youtube
- Recordings
- Youtube Video Sharing
- Desktop Sharing
- Hand heben
- Backgroundblur
- Raum-Passwort
- Chat
- Etherpad-Integration
- neu: Ende-zu-Ende-Verschlüsselung

Integration

- Moodle
- RocketChat
- Nextcloud
- Slack
- [Workadventure](#)
- ...

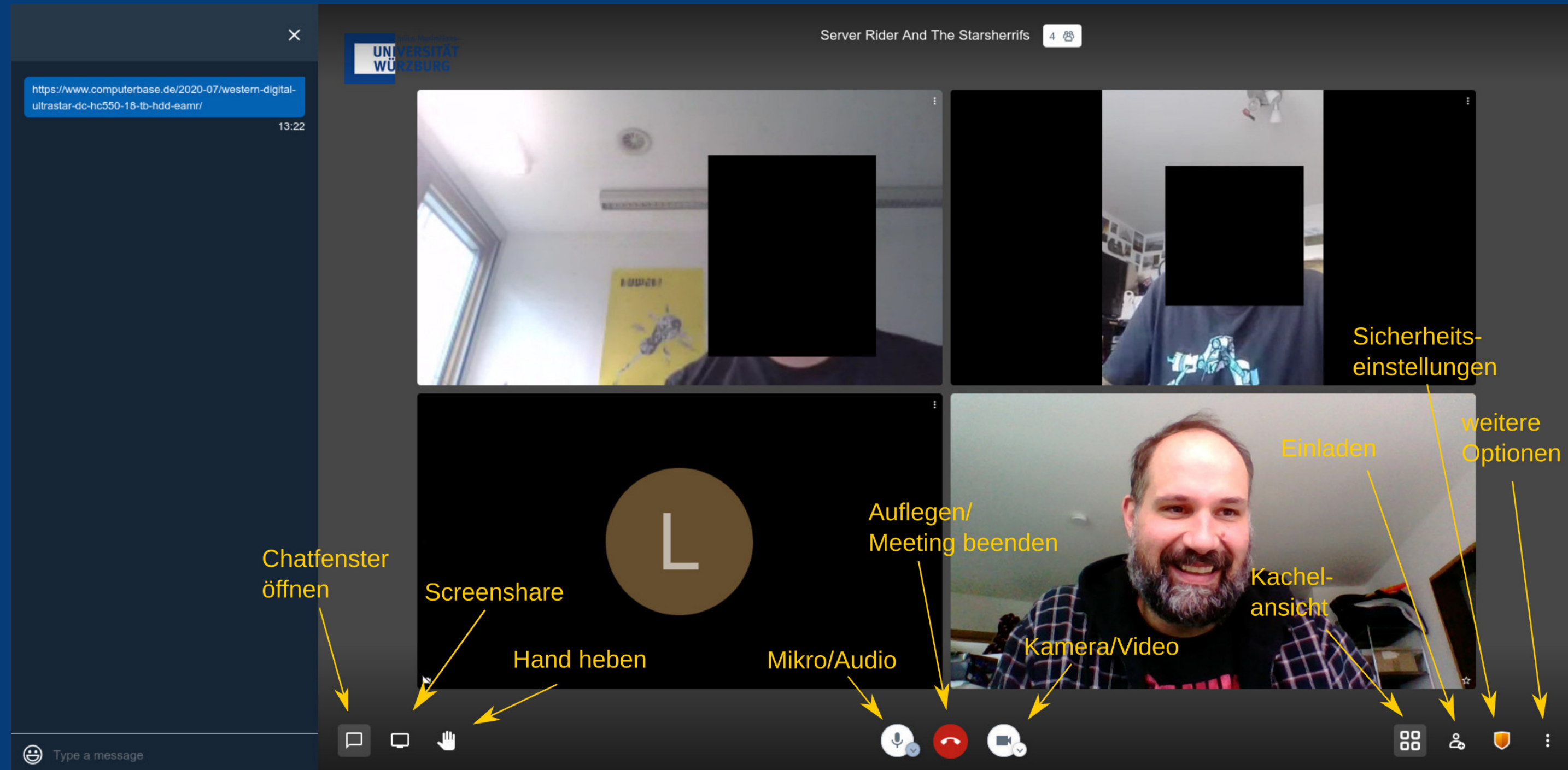
Performance

- Benötigt kaum CPU/Speicher
- Videobridge verteilt nur Videostreams
- Auf verfügbare Bandbreite achten!
- [Jitsi Videobridge Performance](#)

Einsatz an der Uni Würzburg

- Einführung März/April 2020
- “Testbetrieb”
- Empfehlung
 - bei Datenschutzbedenken
 - für kleine Gruppen/Team-Meetings
 - für Ad-hoc Meetings
- Universitäts-Account zum Starten eines Meetings erforderlich
- Docker-Setup basierend auf dem “Stable Release“-Zweig

Wie sieht das aus?



Feature-Einschränkungen

- Deaktiviertes Live-Streaming
- Kein Youtube-Sharing
- Keine Recordings
- Kein Backgroundblur
- Keine Etherpad-Integration
- Keine SIP-Einwahl
- Noch keine Ende-zu-Ende-Verschlüsselung

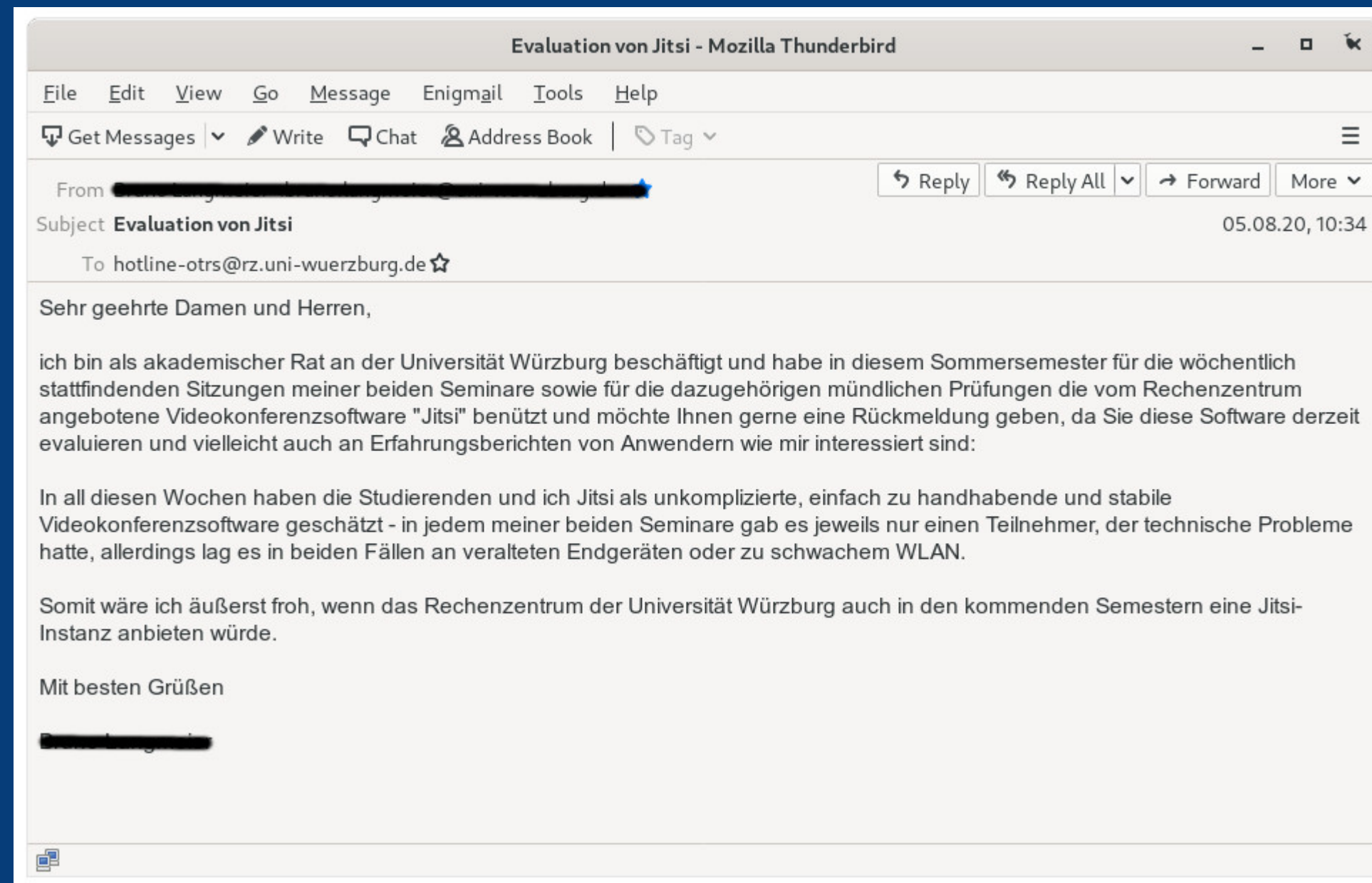
Erfahrungen I

- Schnelle und einfache Bedienung
- Funktioniert am zuverlässigsten mit aktuellen Browsern
 - Probleme mit FF ESR (68er-Zweig) und Chrome
- Android/iOS-App funktioniert
 - Haben u.U. noch Tracker vgl. [Kuketz-Blog](#)
- Desktop-Electron-Apps etwas umständlich
- Bandbreitenproblematik
 - Deaktivierter Videostream (`StartAudioOnly-Option`)

Erfahrungen II

- Browser ist nicht gleich Browser
 - Unterschiedliche Betriebssysteme
 - Bei Linux-Distributionen, je nach Build, unterschiedliche Binary-Pakete mit entsprechenden Features
- Firefox ESR, auf 68er Serie basierend, unterstützt nicht modernes WebRTC
- Firefox ESR, auf 78er Serie basierend, tut in der Regel
- Aktuelle, auf Chrome basierende Browser sollten funktionieren

Feedback



Datenschutz

- Gravatar deaktivieren
- P2P deaktivieren: Somit wird kein externer STUN-Server benötigt
- oder: eigenen STUN-Server betreiben
- Optionale Analytics-Dinge einfach nicht verwenden
- Logging reduzieren
- Datenschutzerklärung anheften

Danke

Nun! Testen:

<https://meet.rz.uni-wuerzburg.de/dachadobeconnect/>

Oder immer aktuell:

<https://meet.jit.si/>