Technische Hochschule
Brandenburg
University of
Applied Sciences

A Lightweight Presentation Server with Generic Web Front End for Developer-Centric Hypermedia Publishing

Tobias Kiertscher, M. Sc.
Prof. Dr. Robert U. Franz
Daniel Kiertscher, M. Sc. (independent researcher)

Overview

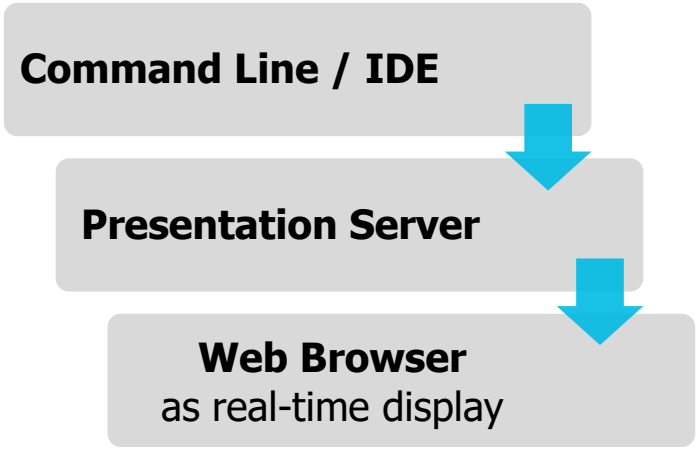
- Idea
- Requirements
- Related Systems
- Proposed System
  - Media Type Support
  - Front End
- Implementation
  - Server
  - Media Processing Pipeline
  - Clients
- Questions, Demo?

Still from „Iron Man 2", directed by Jon Favreau, 2010

Idea

HTTP API for displaying media items
in the web browser

- Presenting media with a simple API
- Displaying a variety of media types: hypertext, source code, PDF pages, data tables, plots, diagrams, images, videos
- Controlling from within an IDE or via command line / script
- Having immediate feedback



Requirements

1. HTTP API backend with minimal protocol overhead
2. Support for arbitrary media content
3. Generic web front end for visual organization of media items
4. Real-time display of published content in the web front end
5. Navigable history for changed media items
6. Local execution on a desktop computer
   as well as deployment on a server

Related Systems

- Headless Presentation Server (e. g. Strapi)
- Hyper Media Display Server (e. g. OpenMHEG)
- Content Management System (e. g. WordPress, Adobe Experience Manager)
- Programming Notebook (e. g. Jupyter Notebook, Wolfram Mathematica)
- Data Visualization View in an IDE (e. g. DataGraph, ILNum. Array Vis.)
- Monitoring and Data Visualization Dashboard (e. g. Grafana, MS Power BI)
- Cloud-Based Presentation App (e. g. Prezi, Google Slides)

Best fitting are Headless Presentation Server and Content Management System

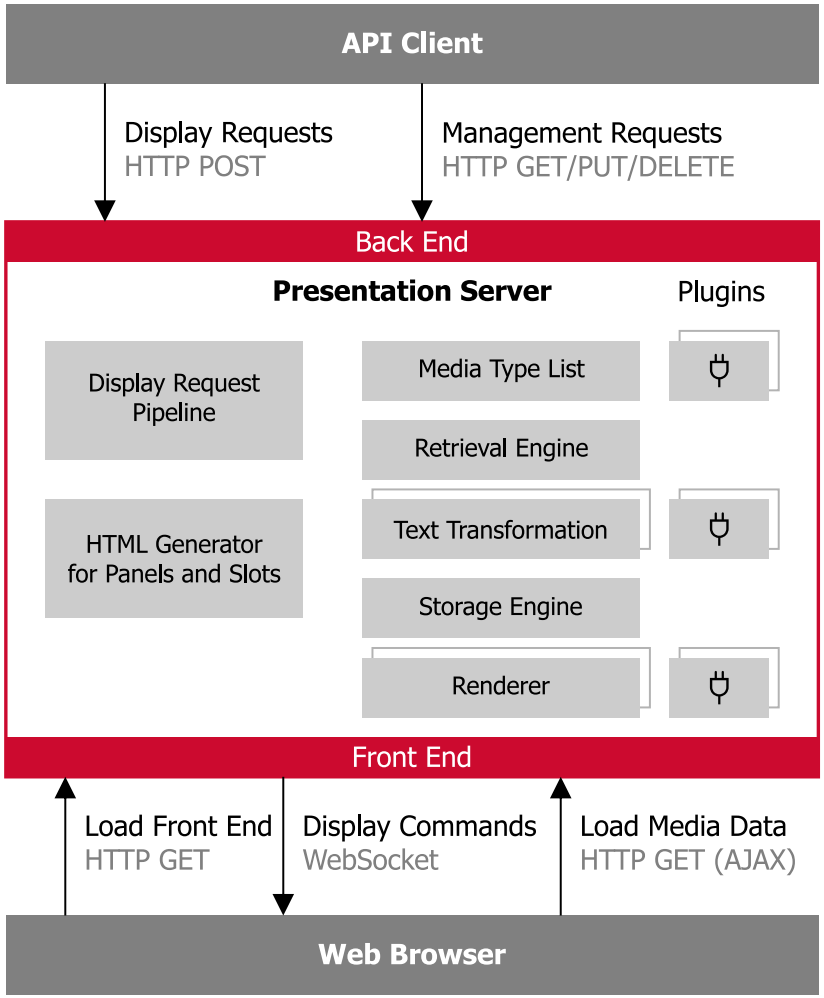
But both lack essential features for our field of application

(arbitrary media types, generic front end, real-time display, navigable history)

Proposed System

A Lightweight Presentation Server with two key innovations:

- Display Requests with arbitrary media types
- Generic Front End with two layout systems



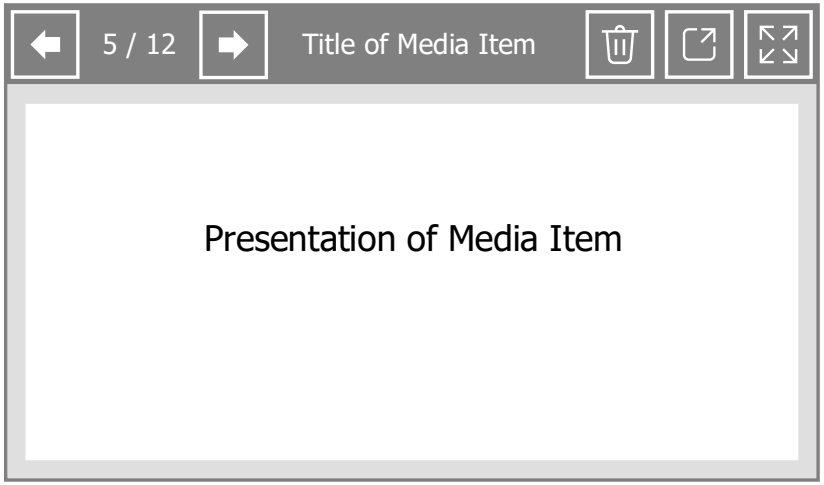
Proposed System – Arbitrary Media Types

- A display request (HTTP POST) contains:
  - Content type (standard, or custom, e. g. `text/x-my-custom-type`)
  - Display options (opt. param., e. g. syntax language, alignment, …)
- Text or binary
  - either the content is text and can be **transformed** into HTML
  - or the content can be **rendered** with HTML + JS + resource URL
- Selection of transformation and renderer is based on given content type
- Text transformations and renderers can be added as plugins
- Media content is stored in different ways:
  In memory, as file on disk, or as external URL

Proposed System – Generic Front End

Structure of a Slot

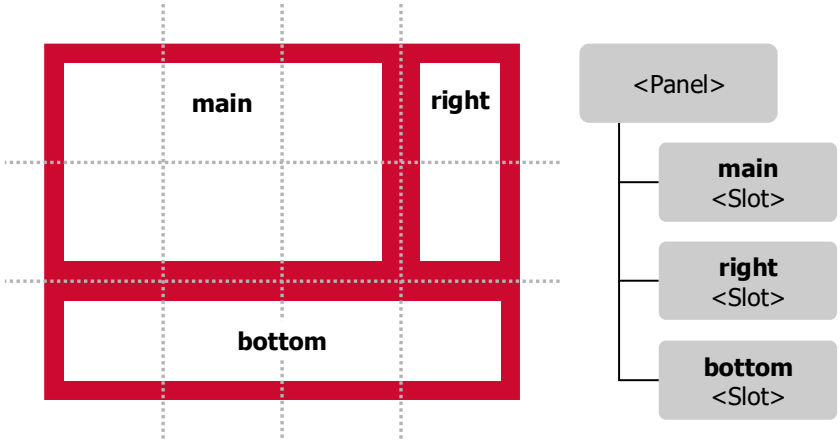
- Optional title bar
- History navigation
- Slot controls for
  - Clear content
  - Open in separate tab
  - Maximize in grid layout



Proposed System – Generic Front End

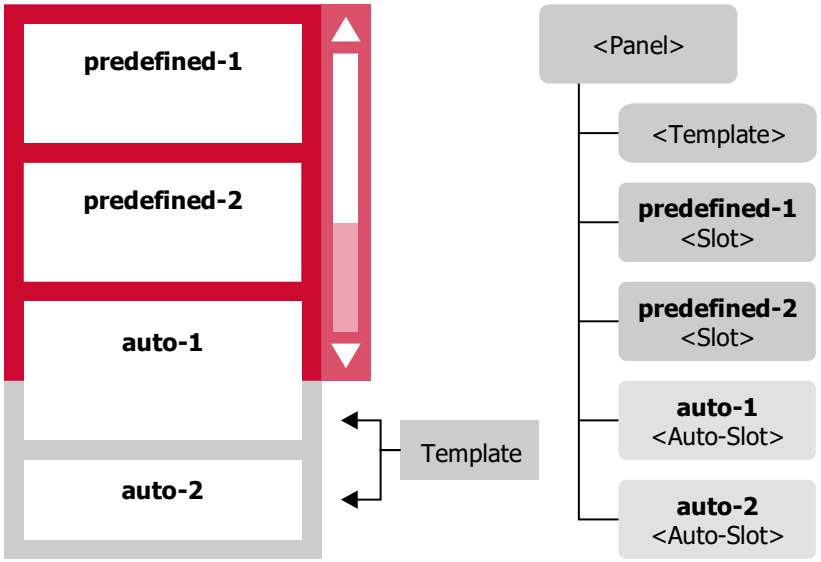
Grid Layout

only predefined slots
with fixed size



Document Layout

predefined and dynamically added slots
with vertically growing size

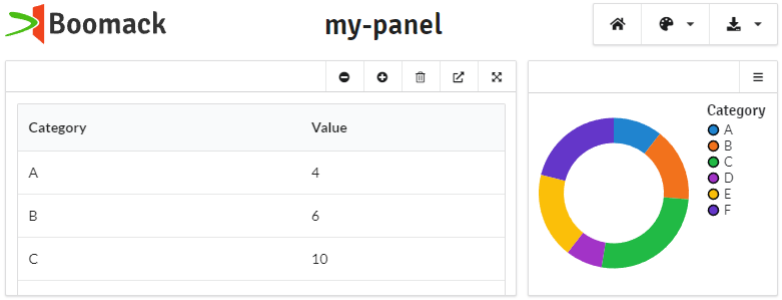


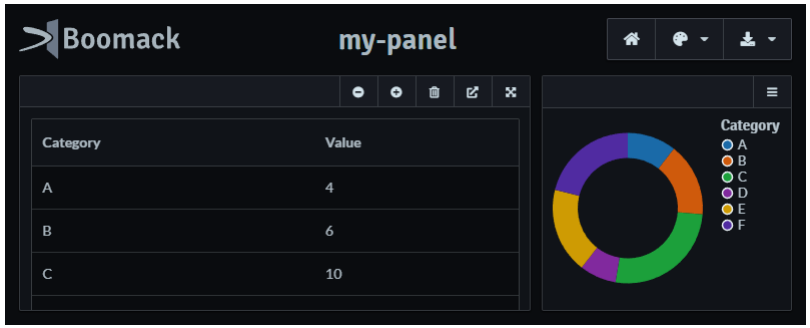
Implementation – Server

Boomack

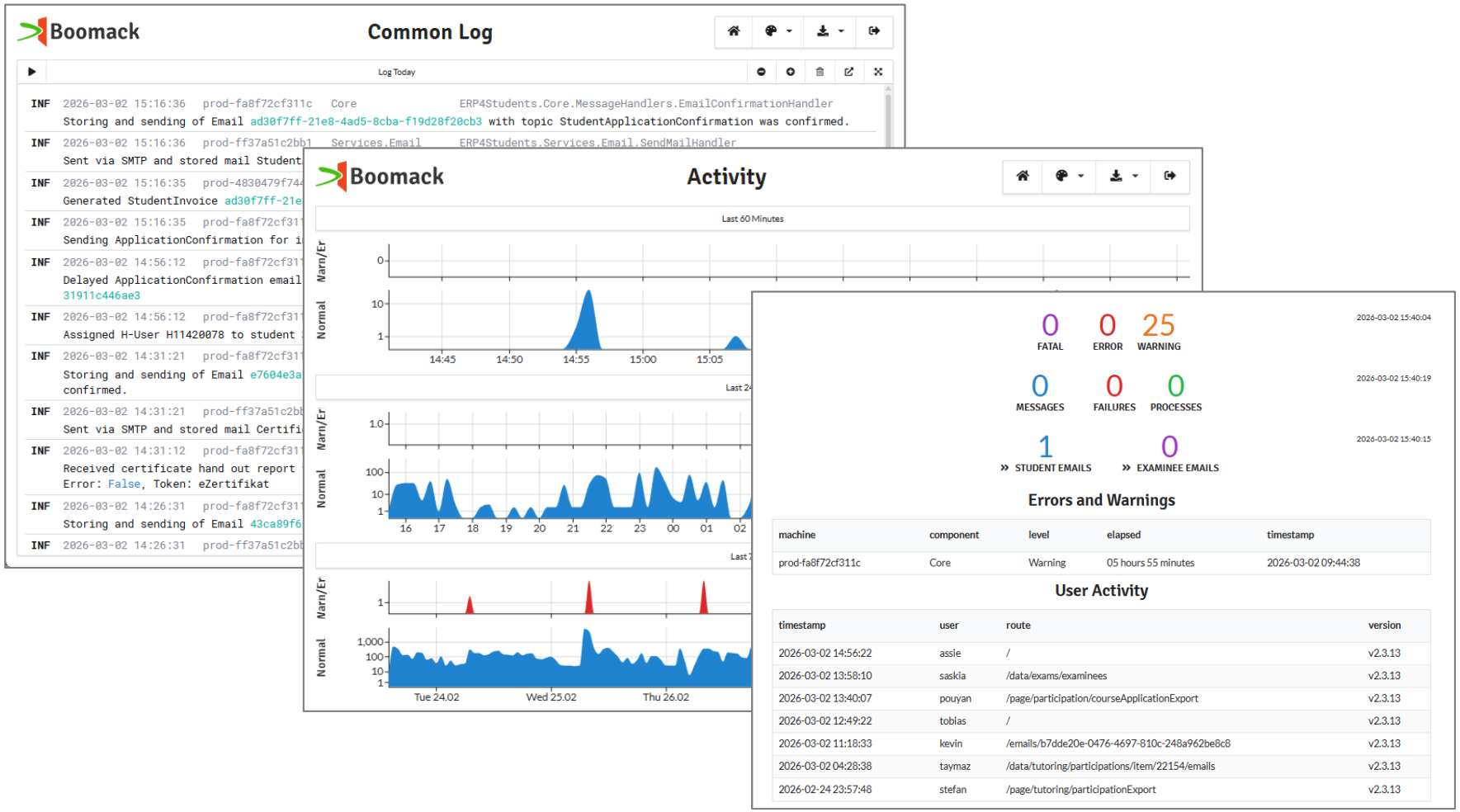
https://boomack.com

- NodeJS application
- Front end uses a fat CSS framework
- Themes (light, dark, scientific, iron, …)
- Media types
  - HTML, plain text, Markdown
  - Source code with syntax highlighting
  - CSV/TSV as table
  - URL in an IFrame
  - Image, video, audio, PDF page
  - Vega / Vega Lite plots
  - Mermaid diagrams
  - OpenStreetMap with markers

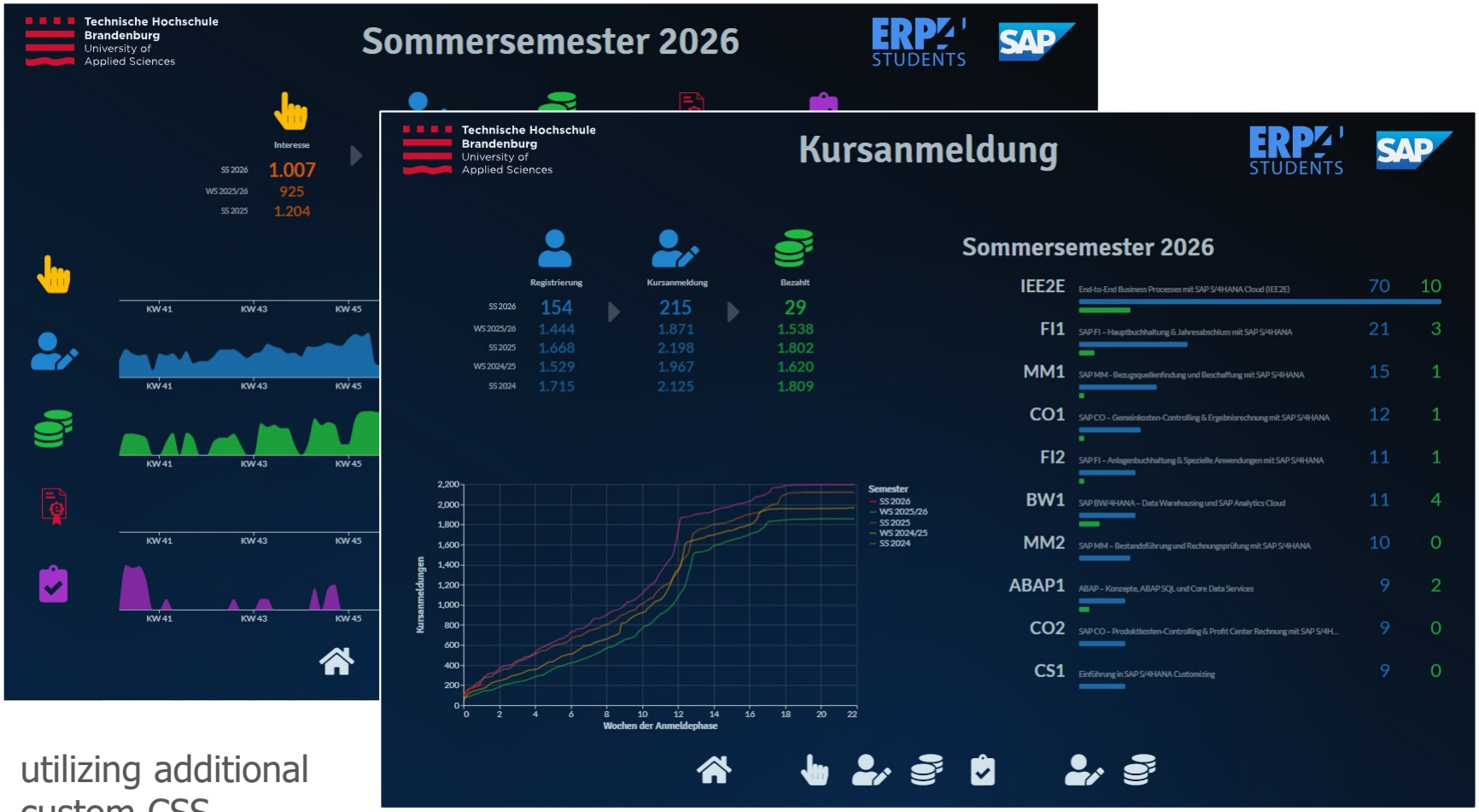




Screenshots – Monitoring

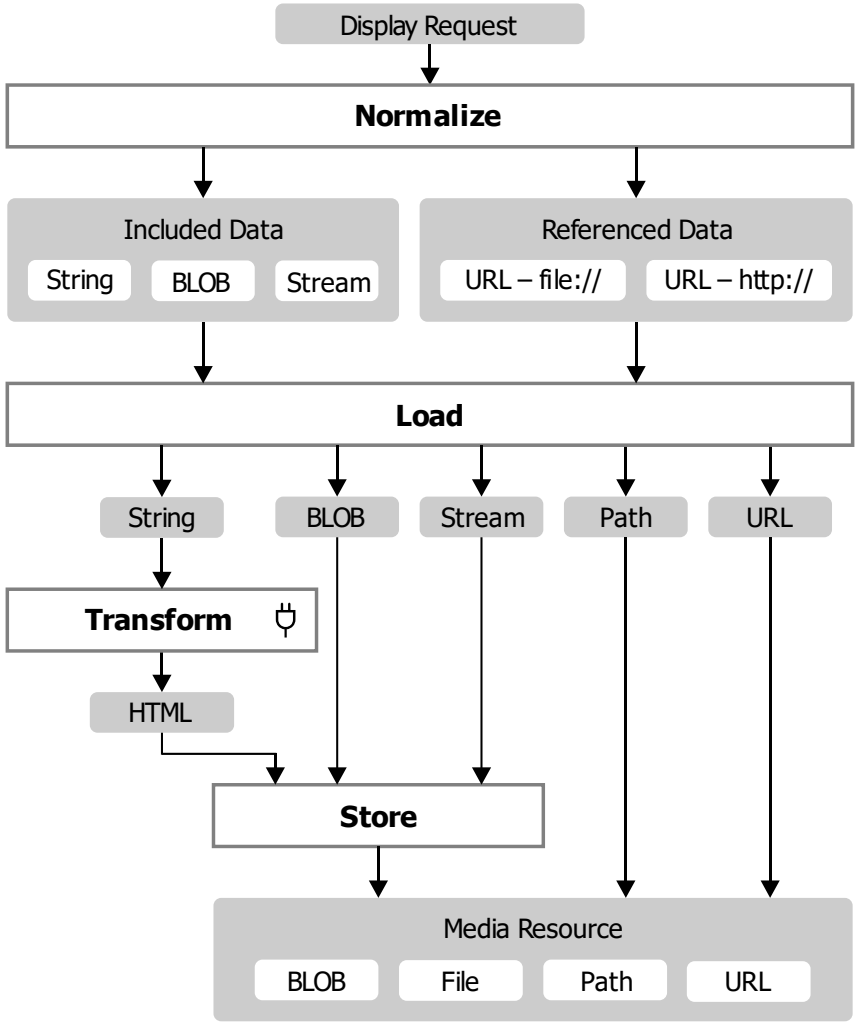


Screenshots – Live Business Dashboard

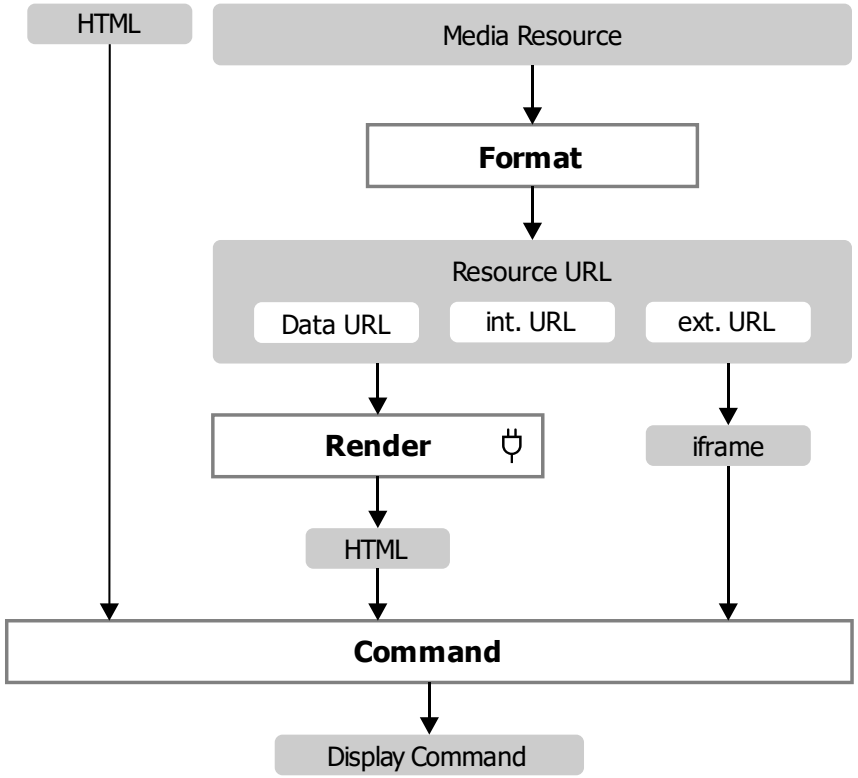


utilizing additional custom CSS

Implementation – Media Processing Pipeline



Implementation – Media Processing Pipeline



Implementation – Clients

- Programming Libraries
  - JavaScript/NodeJS: BoomackJS
  - C#/.NET: NBoomackClient
  - Rust: Boomack Library for Rust
  - …or any HTTP client library
- Command Line Clients
  - `boom` – Boomack Client CLI (NodeJS, most advanced)
  - `bo` – Boomack Client CLI (Rust, fast)
  - `curl` – cURL (for systems where `boom` or `bo` is not installed)
- IDE Integration
  - Boomack VS Code Extension

Technische Hochschule
Brandenburg
University of
Applied Sciences

Thank you for your attention.

Questions? Demo?

tobias.kiertscher@th-brandenburg.de

https://boomack.com