PAUL SCHERRER INSTITUT

PSI

Simon Rees :: Software Engineer :: Large Facility Controls Section

WICA: an *extra* tool in our Control System Toolkit

PSI Controls Section Lightning Talks November 2022

Introduction

Several years ago a friend of mine working in industry told me:
“the days of the desktop client application are over”
“nobody makes applications that way any more, everything is in the browser”

That got me thinking:

- Are we somehow *missing a trick* at PSI, should we be making controls applications using the web ?
- Is our scientific environment so *special* that we can ignore what’s going on in software engineering elsewhere ?
- What would be the *benefits* and *challenges* of bringing our control system to the web?

This talk is far too short to present that topic in any detail. Instead I will present briefly my very early experiences looking into this.

Then, to make things more concrete I will present a new tool called WICA.

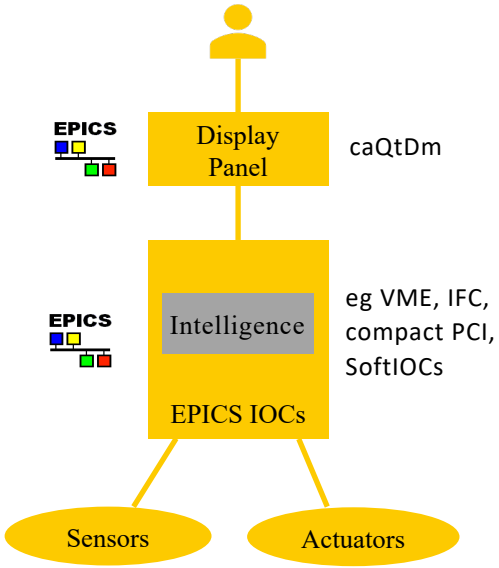
PAUL SCHERRER INSTITUT

Sunset on Desktop Client Applications

...we loved you, but is it time to move on ?

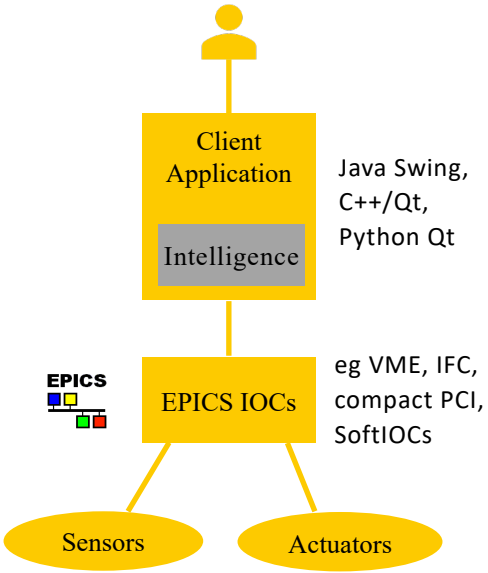
Desktop Client Applications – wtf ?

SLS and SwissFEL were designed from the start to use EPICS



Fat IOC / Thin GUI Client

HIPA and PROSCAN started with a different control system architecture (ACS) and were migrated to EPICS later



Thin IOC / Fat GUI Client

=> today's GUI architectures have been determined by design decisions > 20 years ago !

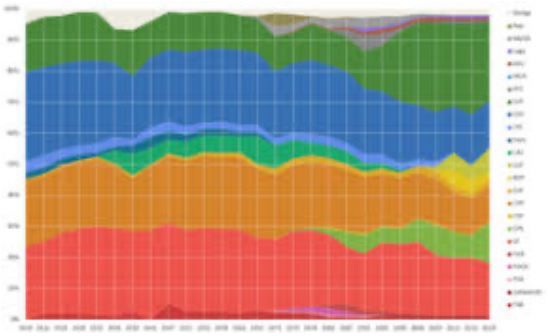
Control System ➔ Web: an *impossible* task ?
Control System ➔ Web: a *pointless* task ?
Control System ➔ Web: an [*insert adjective*] task ?

…The use of web technologies at PSI has invoked strong feelings on both sides

The web can bring benefits, when used *appropriately*, but doesn't come for free

…during the last couple of years I have been looking at some of the trade-offs

A short interrupt…

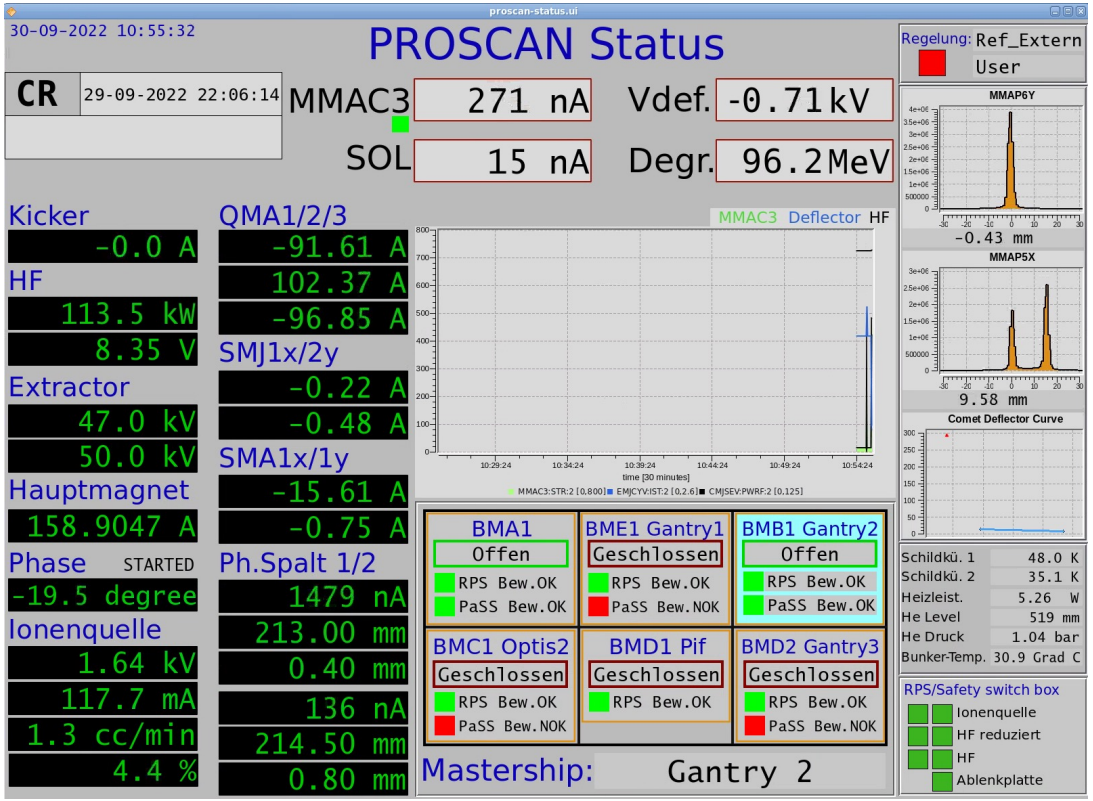


…a Party Political Broadcast on behalf of the web community !

Web Project - Benefits

- Web technologies offer the leading paradigm for today's software development. If we ignore them our controls solutions will be perceived as increasingly outdated and stuck in a timewarp ! Or worse, our users may simply choose to go elsewhere.
- Richer state-of-the-art UX possibilities are now achievable… animations, sound, speech-synthesis, messaging… can be integrated quickly with powerful libraries.
- Web technologies help our on-site staff respond quickly to issues in our facilities.
- Web technologies fit the demands of modern society: remote working flexible/hybrid work models; access anywhere on multiple platforms, onsite, whilst travelling, or at home.
- Web technologies help our *Pikett* support staff diagnose faults more quickly and efficiently, reducing down time.
- The pool of trained web developers is much larger than the pool of, say, EPICS or TANGO developers. Using web technologies should make our recruitment easier.

A Pilot Web Project - PROSCAN Status Page

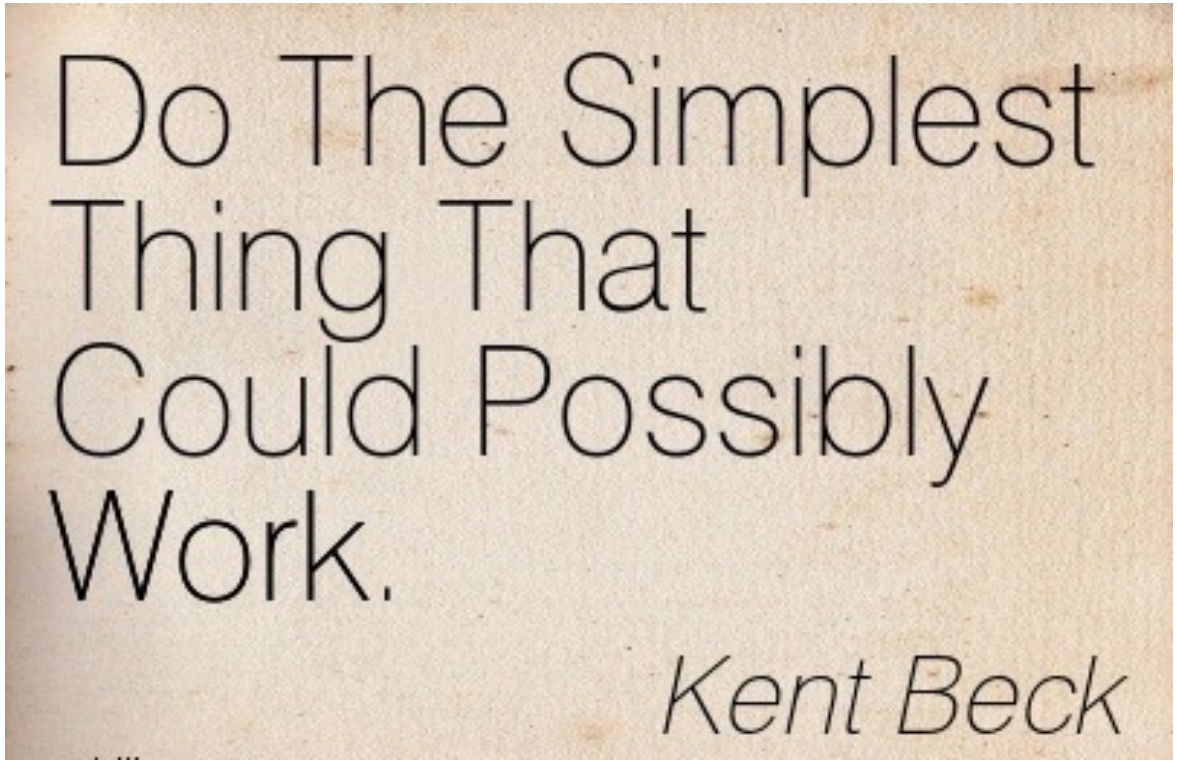


The original display page made with caQtDm

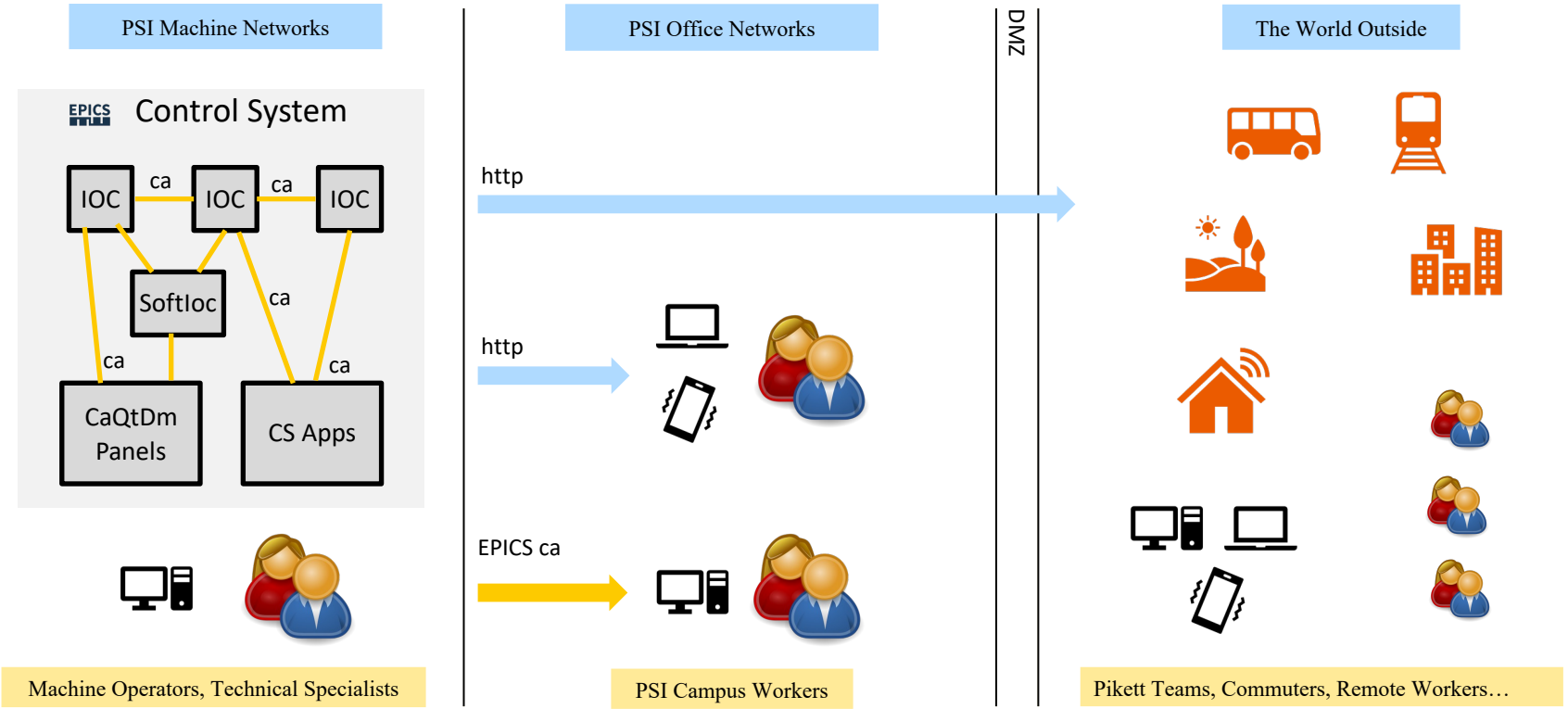
Ugly, but used effectively by many people !

Goal: make something just as ugly, but available on the web

How to start ?



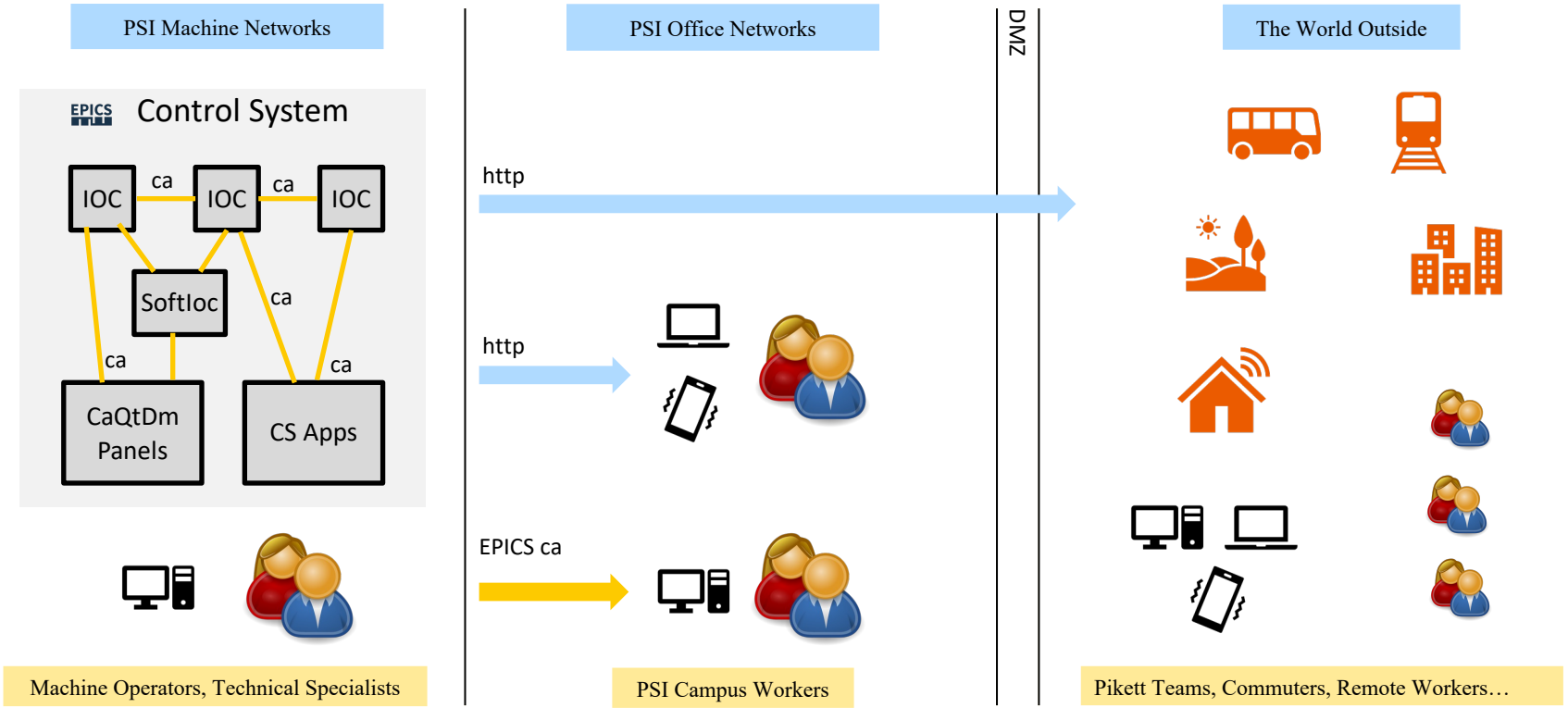
Where to Start 1 – Network Protocols



…if you want to speak to a wide audience it helps if you speak in the user's language !

=> we need an **EPICS->HTTP Gateway**

HTTP: the lingua-franca of our connected world



...if you want to speak to a wide audience it helps if you speak in the user's language !

=> we need an **EPICS->HTTP Gateway**

Where to Start 2 – PSI GUI Review

- Status: Text Only
- Status: Plots (time series, X-Y, barchart, piechart etc…)
- Status: Graphical
- Status: Images (camera)
- Control and Status (eg vacuum displays)
- Applications (eg PROSCAN Ball, HIPA Interlock Display)

…our applications are diverse, the web now has solutions for everything

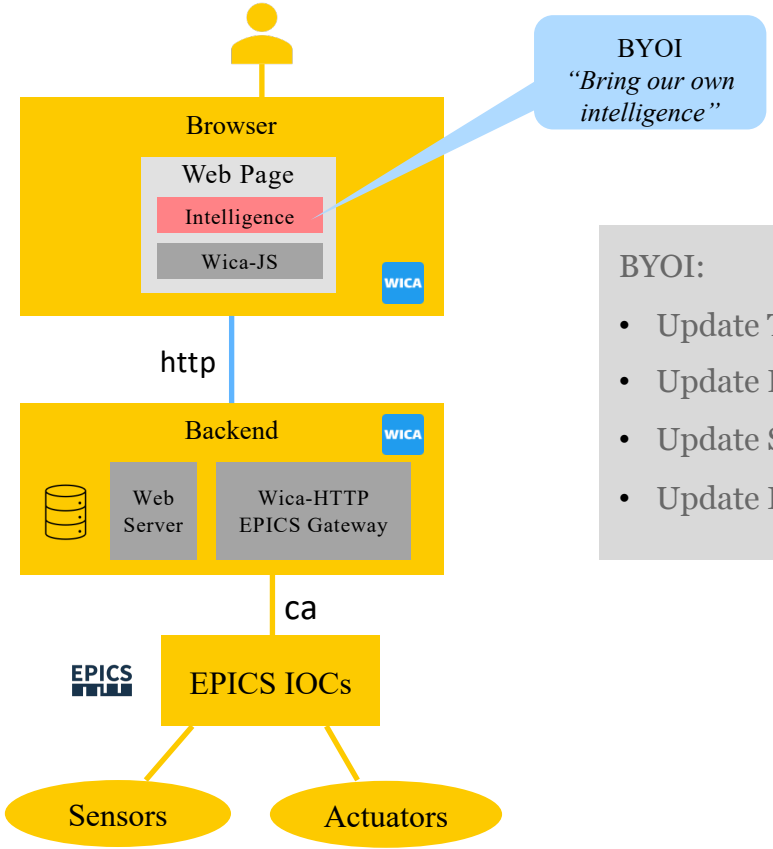
⇒ don't try to do too much !

⇒ bring the **live control system data to the user's web page**

⇒ use **standard techniques** to create the user interface (html, CSS, Javascript)

⇒ WICA is **not** a caQtDm replacement !

WICA – Less is more



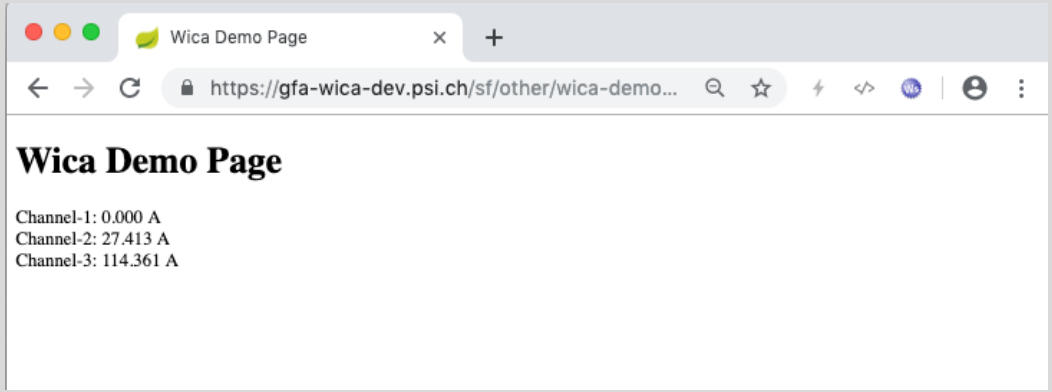
- WICA brings control system data directly to the attributes on the user's web page.
- What you do with it is up to you !

BYOI:

- Update Text Widgets
- Update Plots
- Update SVG Graphics
- Update Images

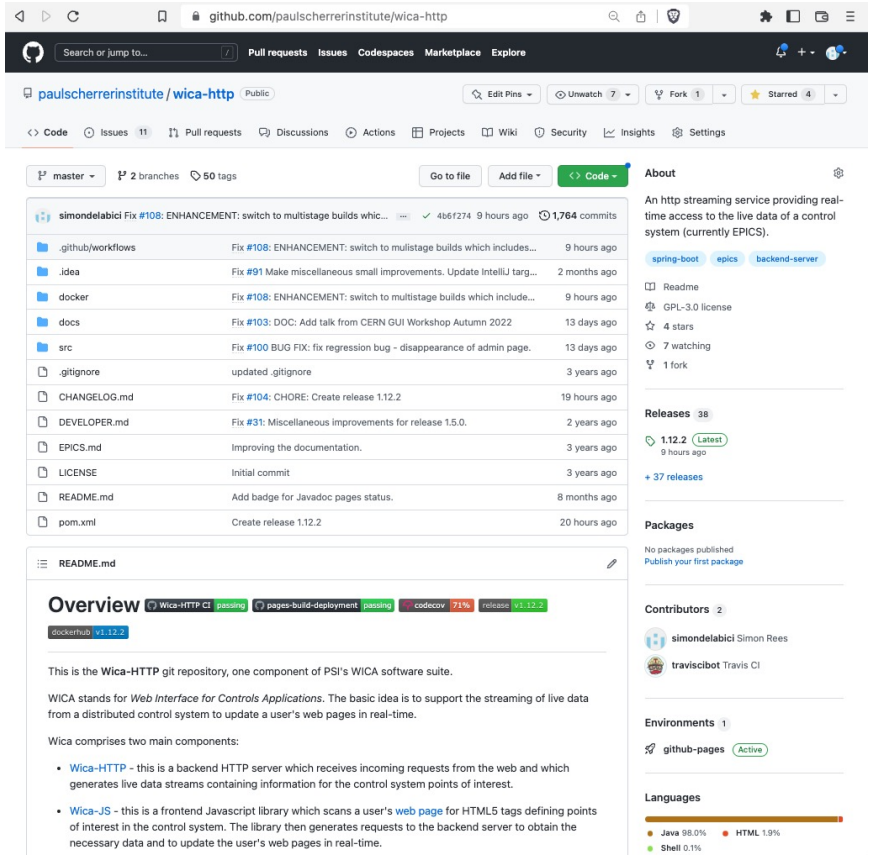
WICA – the simplest web page

```
<!doctype html>
<html lang="en">
<head>
    <meta charset="utf-8"/>
    <title>Wica Demo Page</title>
    <script src="/wica/wica.js" type="module"></script>
</head>
<body>
    <h1>Wica Demo Page</h1>
    <label>Channel-1:</label> <span data-wica-channel-name="SINEG01-MBND300:I-READ"></span> <br>
    <label>Channel-2:</label> <span data-wica-channel-name="SINLH02-MBND100:I-READ"></span> <br>
    <label>Channel-3:</label> <span data-wica-channel-name="SINBC02-MBND100:I-READ"></span> <br>
</body>
</html>
```



Attempting the risky – a live demo of WICA ?

WICA Resources



GitHub Projects
https://github.com/paulscherrerinstitute/wica-http
https://github.com/paulscherrerinstitute/wica-js

PSI External Server
https://wica.psi.ch

PSI Internal Server
https://gfa-wica.psi.ch

Summary

WICA provides a mechanism for accessing live control system data both in and outside PSI

WICA works on phones, tablets, laptops, desktops without the need to install anything

Creating text displays can be done very easily and quickly

Richer user experiences are possible but require extra programmatic effort

Given our resource situation we need to think carefully about where they makes sense

If you want to find out more please get in touch

Thank you for your attention :-)

A particular thank you to Simon Ebner for his encouragement on my learning journey

Also to Daniel Lauk, ex office buddy and my web guru !

Bonus Slides

Web Projects - Challenges

- Richer, state-of-the art user experiences don't come without effort. Is our lab willing to pay that price, or do we prefer to invest the effort elsewhere ?
- Rich web applications tend to be created programmatically by software developers rather than by domain experts. Software engineers are not an easily scalable resource !
- Currently at PSI, we have no Web GUI builder tools to offer to our users. Such a tool would be necessarily complex (if we hope to deliver richer UX) or rely on simplifying abstractions (where we throw away some of the advantages of the web).
- Users may be reluctant to give up their old GUIs, even bad ones: "better the devil you know" ! The users sometimes resist our upgrade attempts unless there is a tangible payoff for them.
- We have no automatic conversion tool for our existing 5000+ caQtDm panels. Should we even create a tool which would only clone the user interface experiences of the past with their more limited richness ?
- The web is still in its infancy: as we try out and discard new approaches the rate of change is far higher than with our classical software development paradigms. There is an "impedance mismatch" between the lifecycle of web technologies and the lifecycle we would wish for the components used on our machines.

Leveraging WICA – supported workflows

Text-only Display Page

Write some html to define your control points

WICA will render the text content of your elements directly - no JS required.

Display Page with Text and Plots

Write some html to define your control points

Choose a JS plot library eg *Plotly* or *Highcharts*

Write some JS to update the plot using the received data

Display Page with Text and SVG

Write some html to define your control points

Use an SVG or text editor to create your graphic.

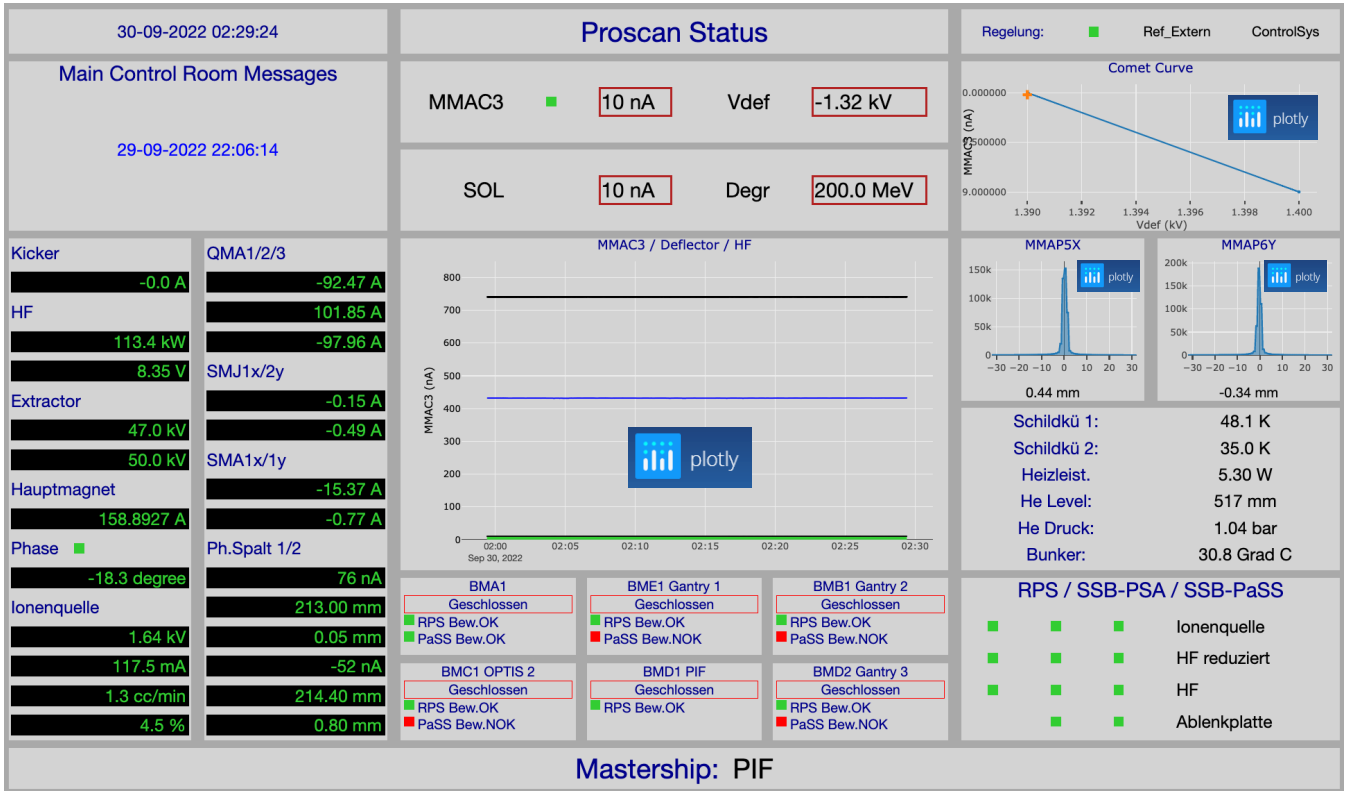
Write some JS to update the graphic using the received data

Display Page with Camera Images

Obtain a Web Component that supports rendering camera image data. At PSI we use *Lit*

Write some html to define the control points associated with each web component

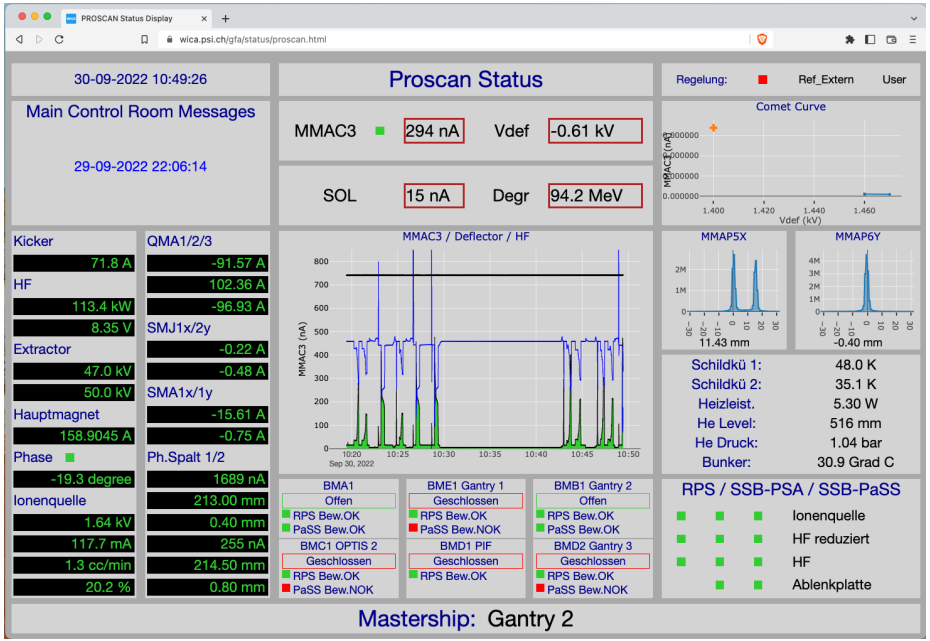
PROSCAN Status Page – Mixed Text & Plot Example



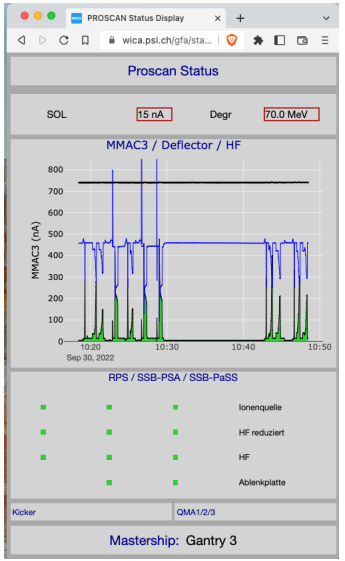
CSS3 Grid

CSS3 Flexbox

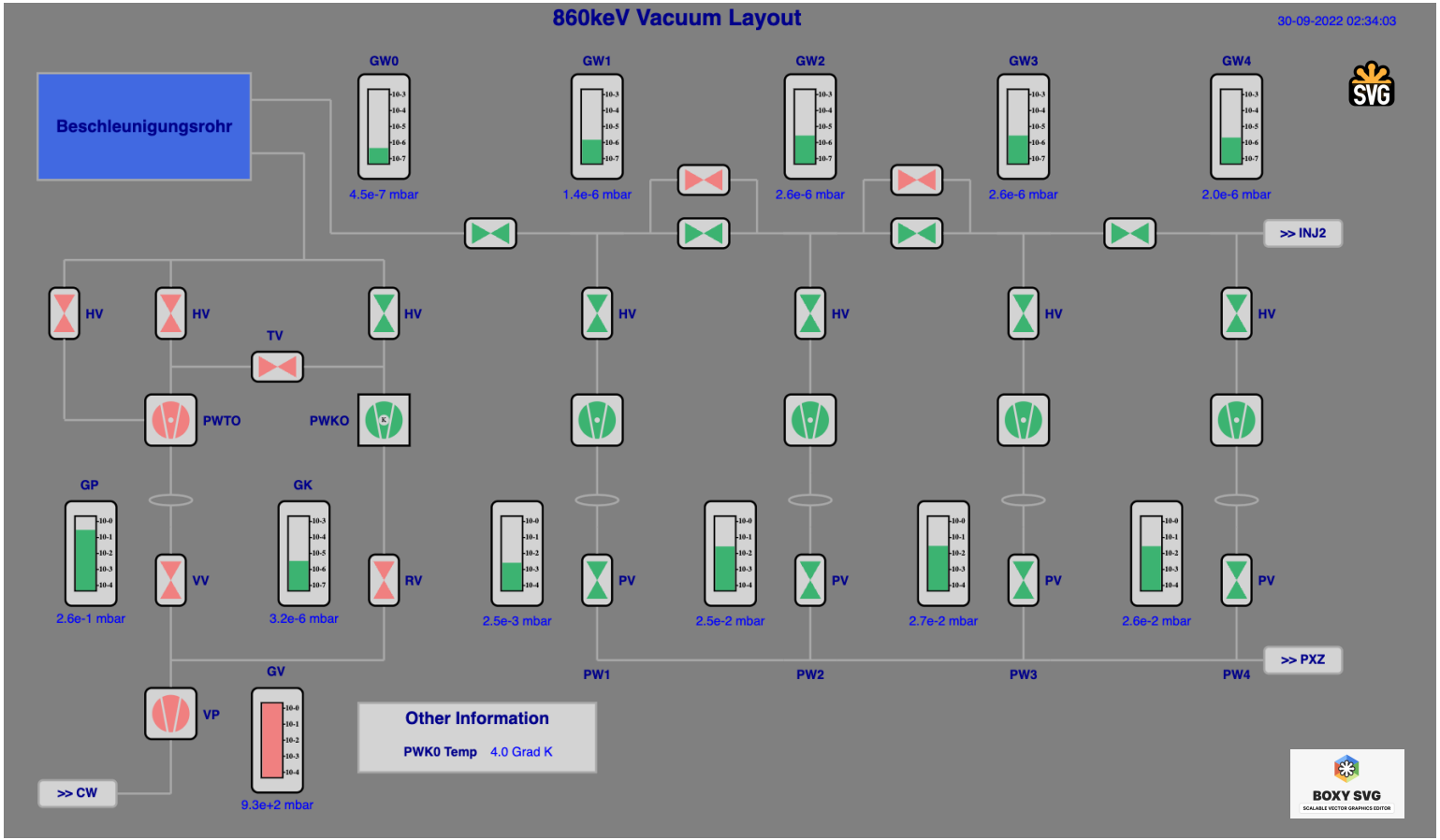
PROSCAN Status Page – responsive design



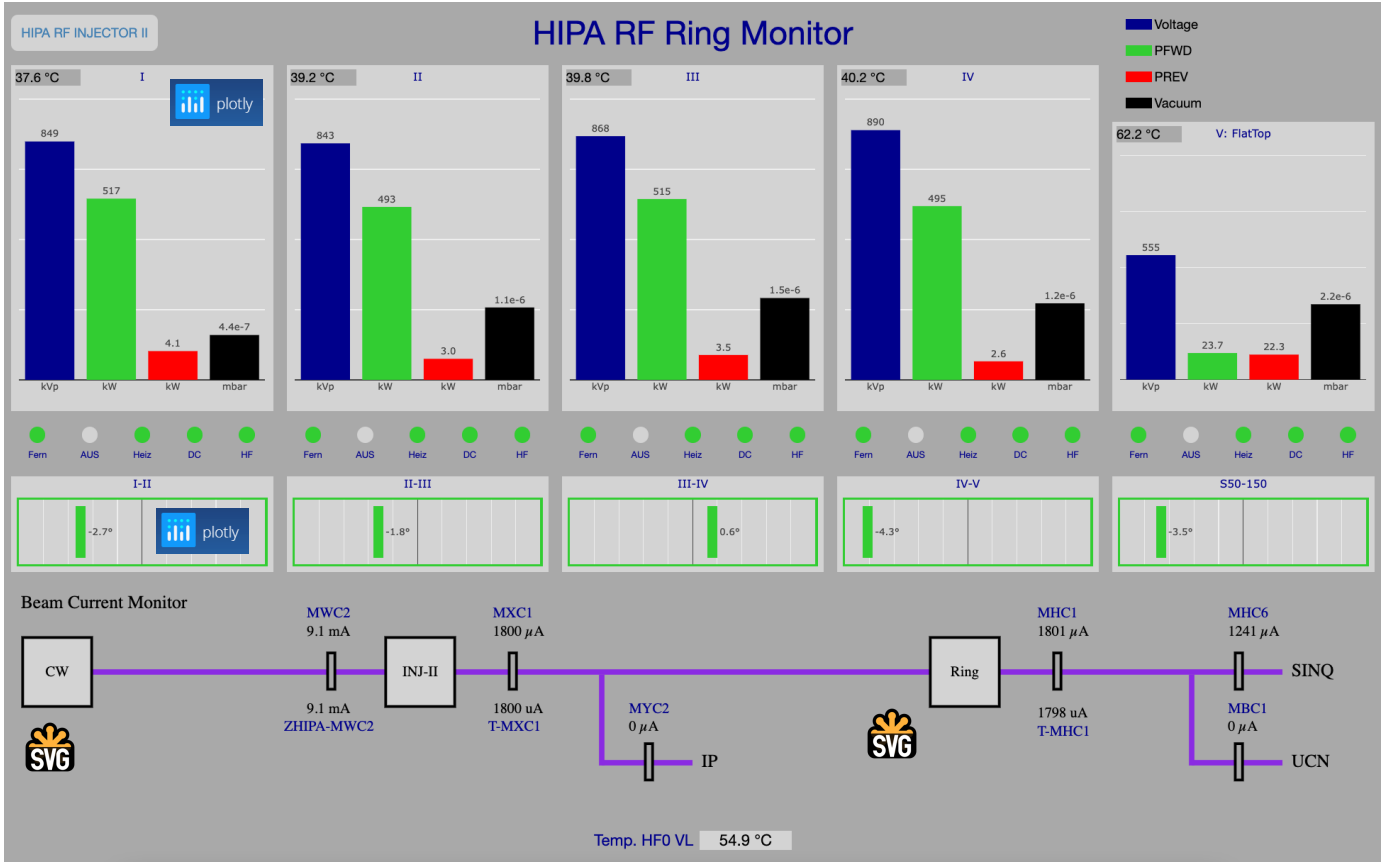
Because the pages just use normal web technology we can use CSS media queries to change the formats according to the features of the viewing device…



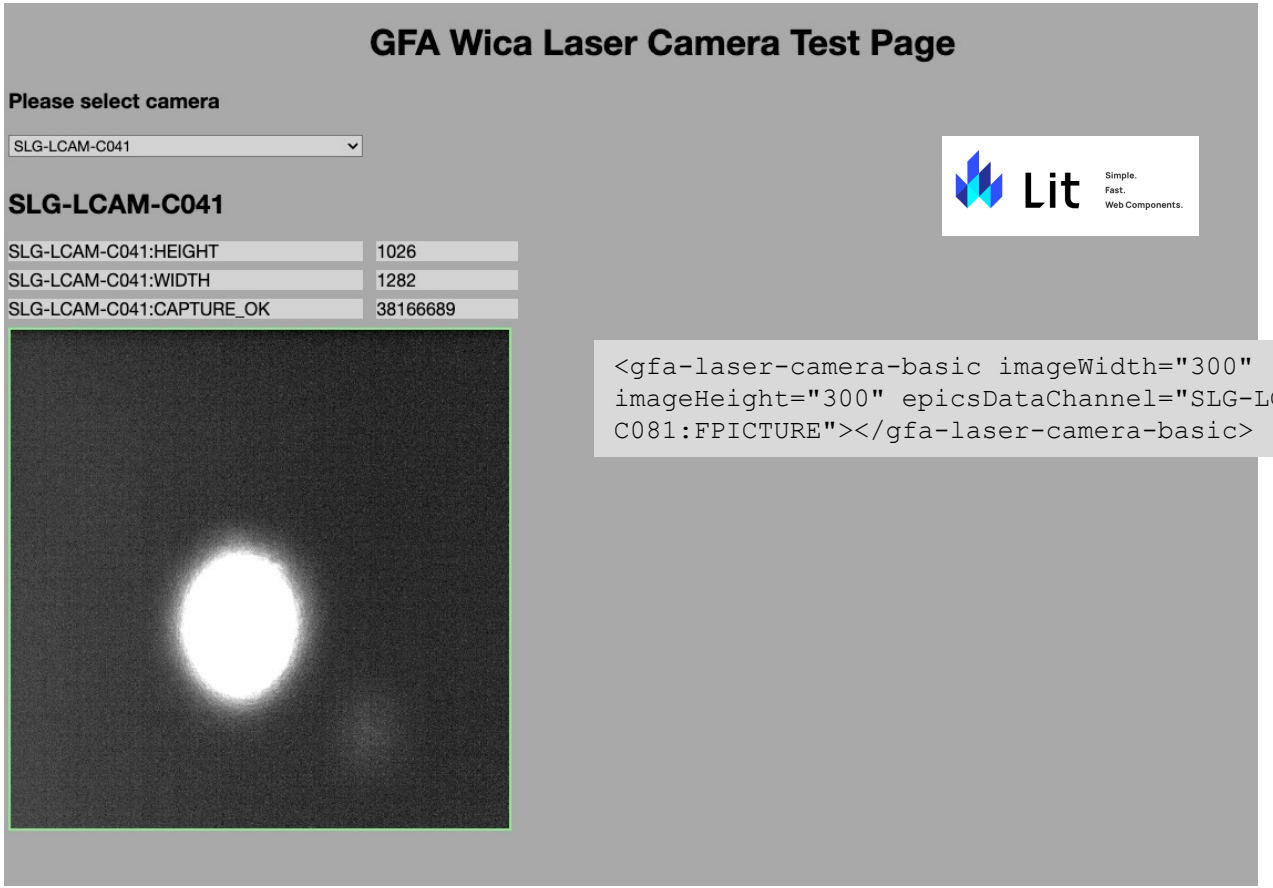
HIPA Vacuum System Page – SVG Example



HIPA RF System Page – SVG + Plot Example



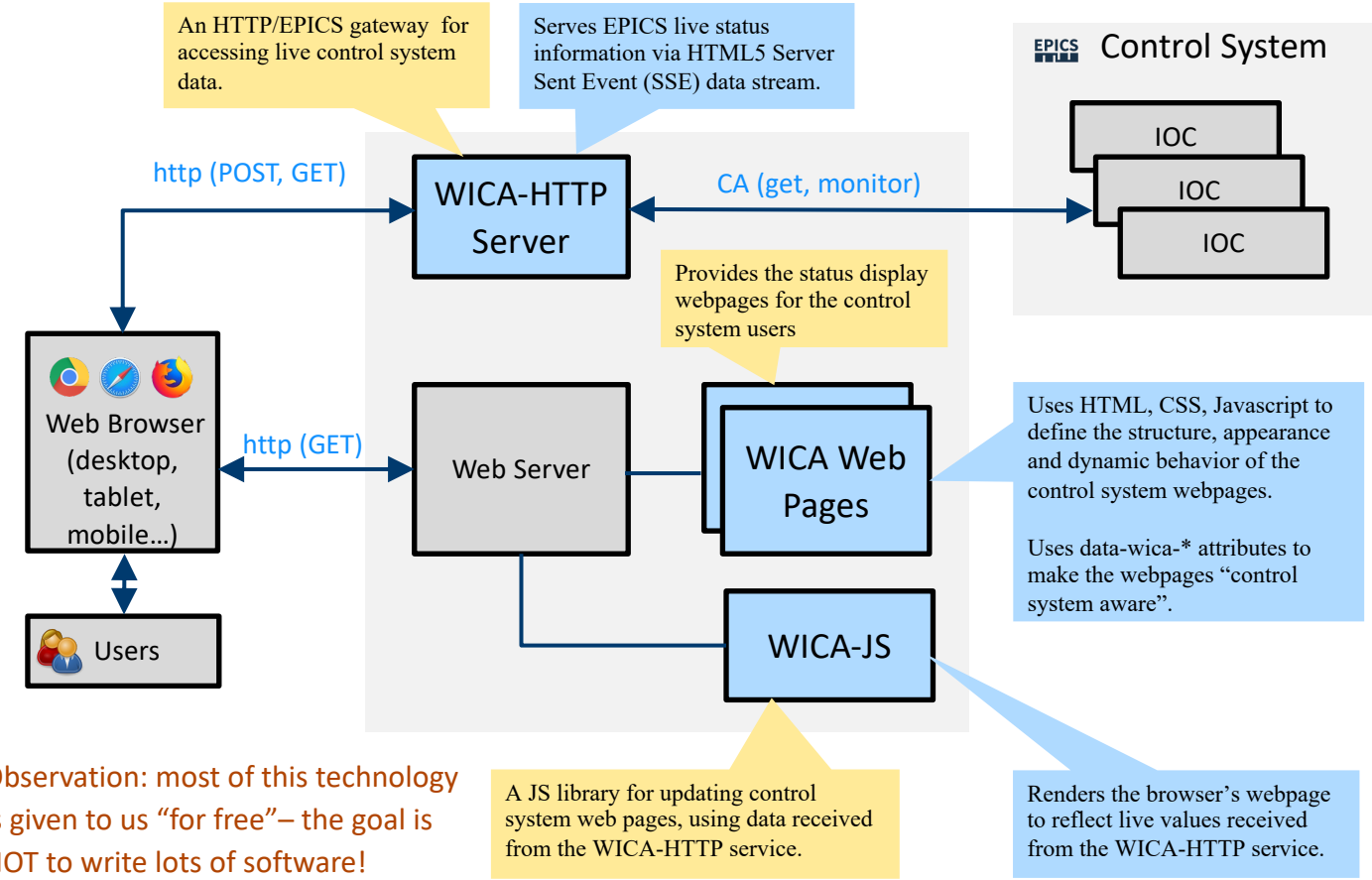
SwissFEL Camera Page – Web Component Example



Using Web Components enable us to create very clean html pages

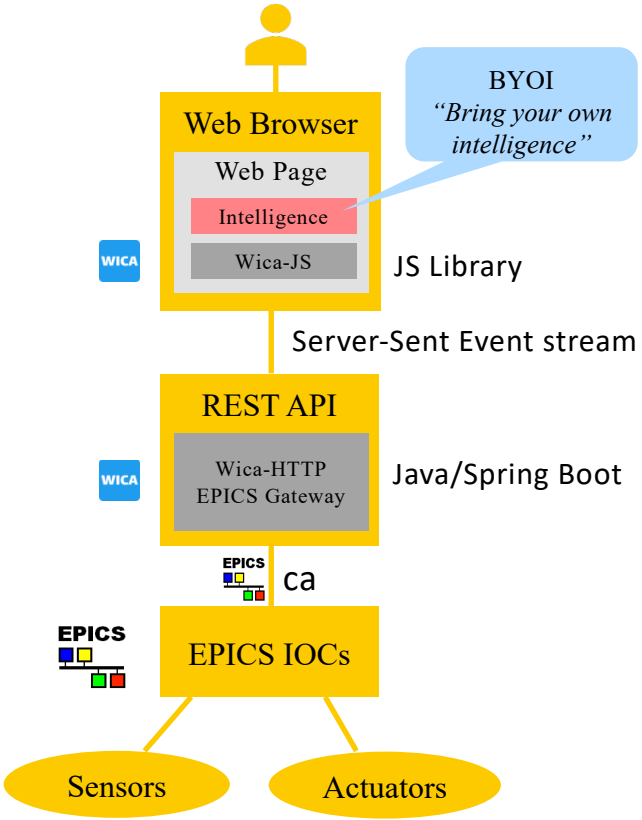
But creating a web component control system library would require significant effort

WICA Overview Picture



WICA – Less is more

WICA = Web Interface for Controls Applications



Main concepts

- Reflect control system *points-of-interest* onto html element data-* attributes
- Use the received data to update **text fields**, **plots**, **SVG displays**, **camera images** etc onto the users display. This can be done in many ways.
- Try to provide the users with some kind of **tool** or **workflow** so they can easily create their own displays

Learning Experiences

- Partial success: WICA is the only display tool that works off-site
- Getting the data to the user's webpage is easy ☺
- Providing a tool or workflow for the users (who are not programmers) has been difficult ☹