References - ideas for the future

OP-Tech Meeting #3 - References
Lukasz Burdzanowski, 9th March 2022
In this talk...

- **What we know, the re-cap**
  state-of-affairs, a brief summary of what was discussed so far.

- **References handover status**
  where we are and how it aligns with our activities.

- **The outlook**
  ideas for the possible evolution towards ”References Service”.

- **Wrap-up**
  summary and proposals on how we could move forwards.
What we know, the re-cap

This talk is based on a few brief talks with Verena (Oct 21), Eva in preparations for IEFC (Nov & Dec 21) and feedback from CO/CSS based on discussions from 2020 and earlier.

Tools for statistics and performance tracking, IEF talk by Eva; Injectors Performance Panel meeting #28 by Chris

Base understanding:

➢ to optimize work of operations there is a need for a system to catalogue structured (e.g. trims, acquisitions) and non-structured (e.g. screenshots) data;

➢ to address this need the “References” application was created by Anthony.

➢ requested functionality is fully provided by the application which was used over LS2; there are valid concerns for its long-term maintenance and evolution.

Last year, Controls system was enriched with: Trims history in NXCALS and LSA TrimTags, both features are related to the references.
References handover status

*From IPP #28 (Chris)* CSS group to get involved from 2021:

- Aim to keep the current solution working.
- No further developments foreseen. Will need to be discussed, planned and staffed.

As of today:

- **No handover took place yet.** Today’s meeting is an opportunity to see what actually is expected ahead (the OP-wide requirements).

- **Technically speaking - the application is not fully aligned with CSS stack:**
  J2EE in place of Spring; binary (blob) storage with eLogBook database in place of structured RDBMS/time-series DB, etc.; not deal-breakers but a concern for the maintenance.

- **Effort to maintain this application will come from LSA team**
  APIs/Controls system/3rd party libs - are changing and will break the application, the question is when.

There are unknowns:

- Who exactly is the actual user in OP, primarily SPS/PS or all of the machines? What if the application does not work, how critical it is?

- How much should we invest into it instead of LSA/InCA?
The outlook

What we can do:

➢ During Q2/Q3 we take over the References application, with the assumption that no any development and bug-fixing will take place. We just keep it running.

➢ Based on OP requirements and priorities, by the end of this year we come up with a concrete design of “References service” and a milestone plan (to know what it would cost).

By storing data in NXCALS service, its tooling becomes available: SWAN, TIMBER, client APIs

Source for documentation auto-generation tools (wikis, etc.)
We have an idea of how the **References service** could look like.

On a related note:

- with input of OP/EA, by the end of Q2 we aim to finalize **CESAR evolution review**:
  - design its future architecture → benefit from LSA, invest into client API;
  - look closer at the notion of **beam files** → reassembles to a degree the **references**.
- to drive the idea of **values tagging in NXCALS** we need clear requirements.

We **need to know Your needs** (OP as a whole):

- to verify if anything is missing in the Controls system to proceed;
- to see how we can **leverage existing Controls services**
  - to limit number of dedicated tools → **lower maintenance** for everyone

The experiences and results of CSS ↔ OP collaborations
(TrimTags, MLP, Fixed Displays, etc.) speak for themselves.

Can we apply them in this scenario?