BI settings management for LHC

E. Bravin

BI TB, 9 June 2016

Present situation

- For most systems settings management is based on FESA persistency
  - i.e. no management
- Sometimes people put "reference" settings in different logbooks/files etc.
- Some settings are logged in LDB
- No coherent approach trough the group
- Only experts can diagnose an error on settings (if they can)
  - Can take considerable time to spot a mistake

How to improve

- Log all settings on change
  - History of settings available and accessible trough TIMBER
  - Tedious to set-up (need to declare fields one by one)
  - Would need expert tools to extract/compare/redrive
  - You can use JPAC and RDA to change settings (JAVA, C++, Python)
- Use LSA/INCA
  - Can import device:property blocks in INCA
  - Tools available to check history and drive settings
  - Inca-fication allows using JAPC to change settings in INCA/LSA
    - Likely all settings have to go this way -> No more C++ or PyRda

INCA/LSA

Direct access

INCAified applications
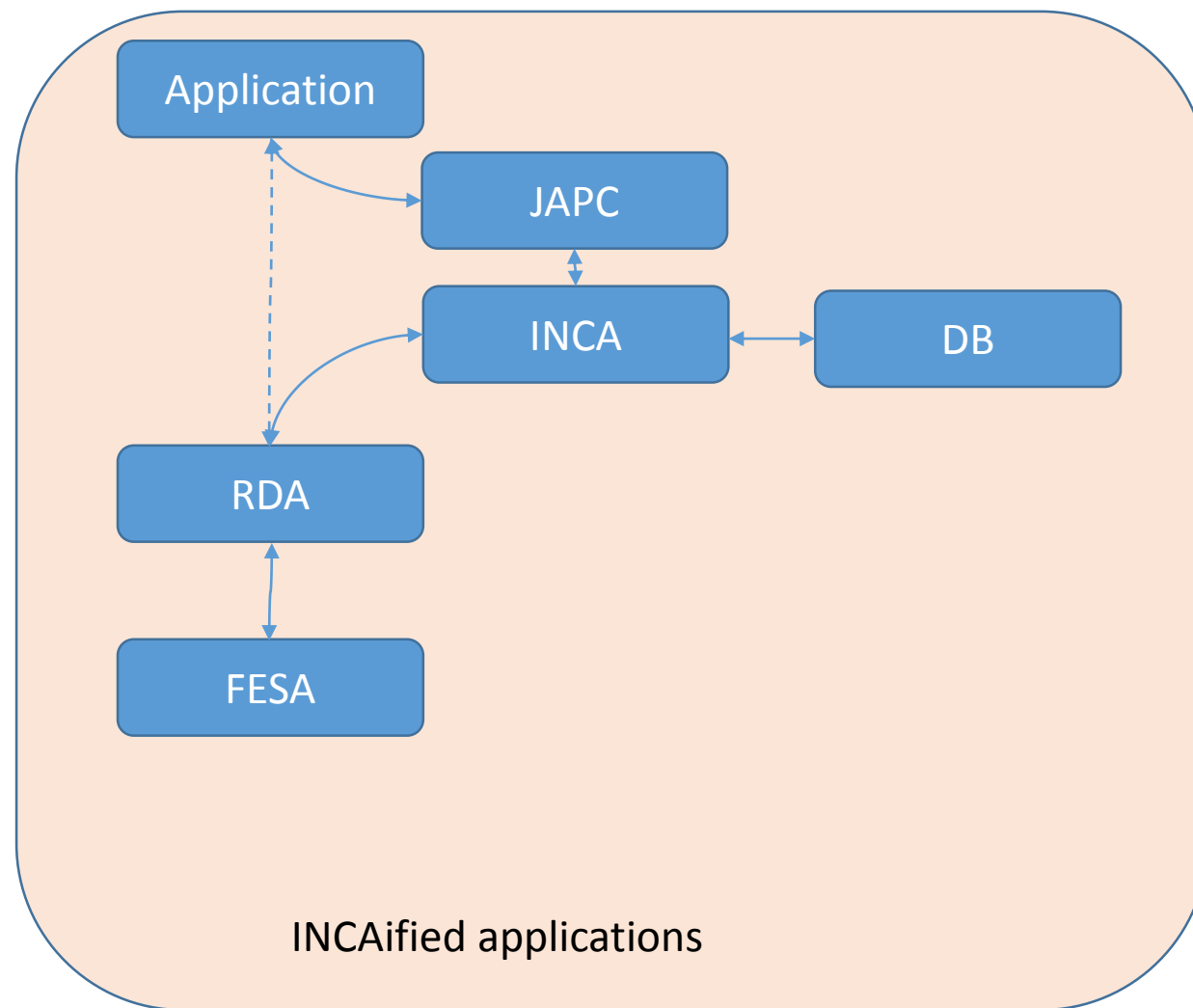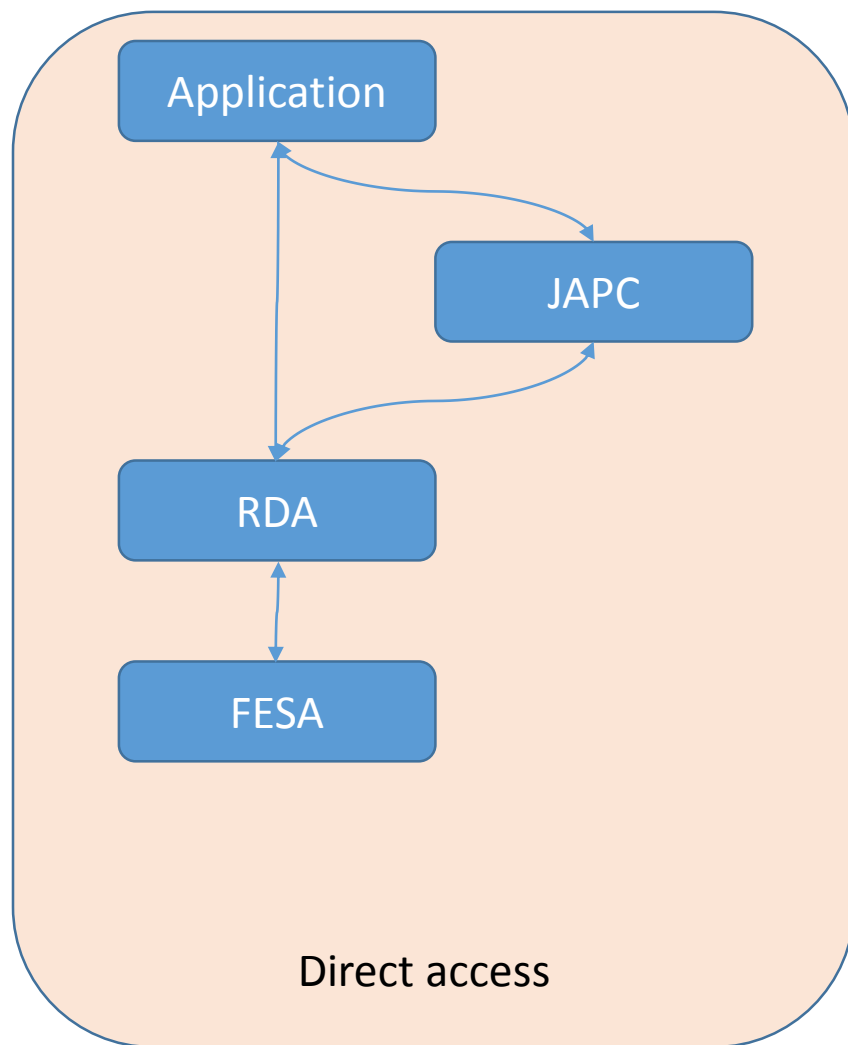
INCA limitations

- Some types/array structures may not be (well) supported in INCA
- INCA provides a DB for settings (history) and tools to check/drive them. Not possible to check against a reference (not easily at least)
  - i.e. if you change using an incaified application by mistake you can not spot it automatically, but you can check the trim history in case of doubts.
- INCA and LSA provide only JAVA libraries