iTesla
Innovative Tools for Electrical System Security within Large Areas

iPST opensource project

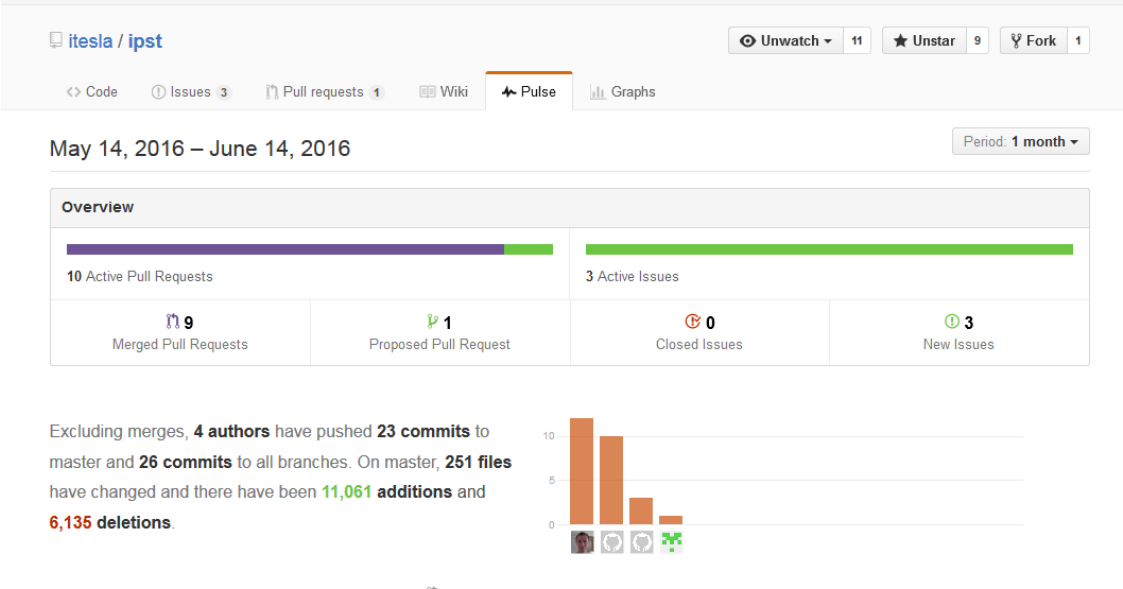
iTesla Power Systems Tools

Nicolas Omont, RTE
nicolas.omont@rte-france.com

PSCC – 23nd June, 2016

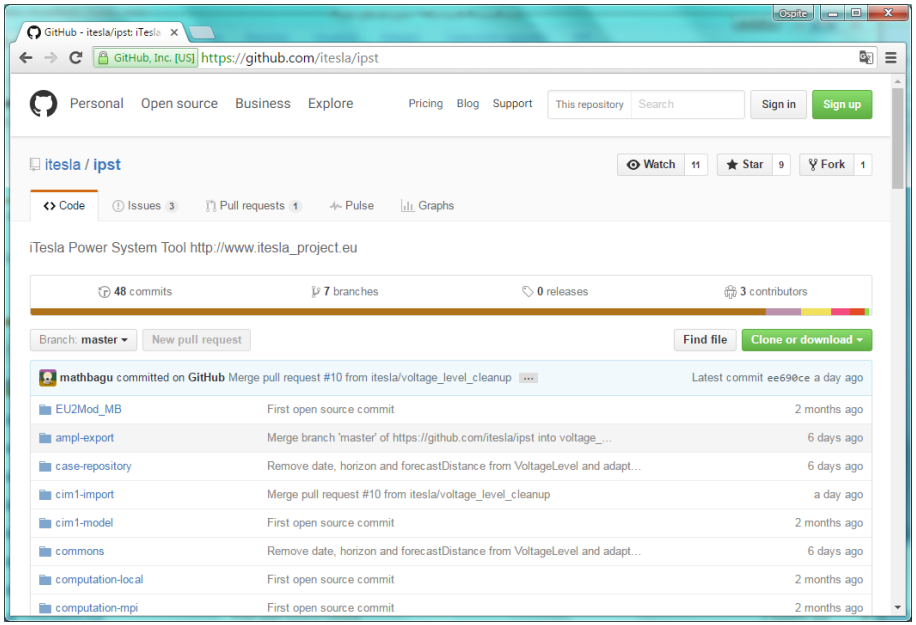
iPST opensource project

- The iPST project was initialized with code produced during the FP7 iTesla project.
- Current size of the iPST project on GitHub:
  - 140000 lines of java code.
  - 10000+ lines of C++/C
  - 8600+ lines of MATLAB code
- 9 developers (with write permissions on the project)



iPST opensource project

- iPST source code freely available on GitHub: https://github.com/itesla/ipst



- License: MPL 2.0

  Permissive license: covered code must remain under the MPL, can be mixed with code under different licenses.

iPST: Basic Functionality

- Basic process: standard N-1 security analysis:
  - Read grid data in UCTE, CIM or internal data model format.
  - Run power flows before and after contingencies.
  - Output the list of constraints.
  - Output grid data in internal data model format.
- Current limitations:
  - No open-source power flow is provided
  - Only CIM 14 is supported
- Planned developments:
  - Free binary versions of power flows (AIA's Helm and RTE's Hades) with limitations (size, restriction to academic use…): end of 2016.
  - CIM CGMES importers and exporters: first trimester 2017.

iPST: Enhanced Functionalities

- Dynamic simulations:
  - Enrichment of static grid data with dynamic models
  - Call of Eurostag or Dymola-based simulators
  - Use of the iPSL (iTesla Power System Library): open-source repository of dynamic models
- Merging: “topological” merging of CIM files
- Machine learning:
  - HistoDB: free of use database for time-series related to historical datasets of forecasted and realized grid data.
  - Uncertainty module to model forecast errors.
  - Decision tree module to learn “rules” (soon).

iPST: computation capabilities

- Highly scalable platform tested up to 10,000 cores.
- Adaptable to most computing environment as it is MPI based .
- Based on standard CPU code compiled as “normal” executable (no rewriting).

Documentation and support

- “Hand book” written by the validators during the FP7
- Javadoc: especially useful for those who want to plug their simulators / OPFs into the platform.
- Wiki and forums on github (to be developed)
- To be developed: tutorials and binary distribution.
- “Best effort” support on the forums. Commercial support by iPST project partners could be possible.

Conclusion

- A toolbox dedicated to innovation in short term grid security analysis processes.
- An help for CGMES implementation: its simple yet powerful internal format “buffers” CGMES CIM complexity.
- An open-source project with a long-term perspective thanks to the multi-year engagement of RTE which will base its operational tools on it.

Questions?

iPST website: https://github.com/itesla/ipst

iTesla FP7 website: http://www.itesla-project.eu/

Contact: nicolas.omont@rte-france.com