Migration of STEAM-SIGMA Inputs from Java to python

One-week work shadowing

Wiktor Liśkiewicz
11.04.2019

TE-MPE-PE

http://www.cern.ch/STEAM
I’m 16 years old and I am currently studying Maths, Science, English, Statistics, French, History and Geography at school.

I’m spending just this week at CERN during my holidays.

As I don’t take Computer Science at school my programming skills before my visit here was very limited.
A story from an R&D engineer’s life

0. A curiosity-driven request to perform a study for: LHC, HL-LHC, FCC*
1. Numerical/Analytical analysis to carefully evaluate various scenarios
2. Selection of a handful of working models out of hundreds trials and errors
3. Double-check to convince oneself that the results do make sense
4. Summary of results as a report, note, paper, presentation*
5. Conscious acceptance of a broken link between 2. and 4.

*choose one that applies

How can we maintain the link between models and analysis?
STEAM Integrated Generator of Magnets for Accelerators

1. INPUT
- User Input
  - Database Material
  - Database Cable
  - Enumeration Type

2. FORMULATION
- Domains []
  - Geometry
  - Domain i
    - label
    - Material
    - Elements []
    - Enum Type
  - Element j
    - label
  - G-Object k
    - Hyper/Area
    - OR
    - Coil

3. IMPLEMENTATION
- Geometry
- Material
- Physics
- API
- Model
- Mesh n’play, for both monolithic and cooperative simulations
- Custom implementation
- GetDP
- LEDET

Mesh & play models
- Construction time of minutes
- Automation against error-prone, manual work

Field extraction based on strands coordinates (input for third-party software)

Other models: 11 Tesla, D1, … and many more!
The Challenge: Make STEAM-SIGMA user-friendly

- Expert tool for Java development
- Multi-stage installation process
- Issues with dependencies
- Big overhead of project files
- Geometry visualization not supported

The initial time investment (to learn Java and IntelliJ) is unacceptable for our users
The Solution: SWAN Notebooks - demo

✓ Environment tailored for scientists
✓ No installation – runs in a browser
✓ No issues with dependencies
✓ Only input files, lean projects
✓ Geometry visualization supported
Java to python parsing

There are 12 STEAM-SIGMA inputs to be migrated (1 done and tested)

Geometry

Point kpc = Point.ofCartesian(0, 0);
Point kp1 = Point.ofCartesian(r, 0);

Cable parameters

this.wInsulNarrow = 1.55e-4; // [m];
this.wInsulWide = 1.35e-4; // [m];

cableD1.setwInsulNarrow(1.55e-4)
cableD1.setwInsulWide(1.35e-4)
Cable Parameters Parser - Demo

```java
inputString = 'this.wInsulNarrow = 1.55e-4;'
```

A parser would change the java input into the python output

Expected Output

```
cableD1.setWInsulNarrow ( 1.55e-4)
```
Magnet Geometry Parser - Demo

```java
inputString = 'Element ho3_3 = new Element("IY_HOLE3_3", ar3_3);'
```

A parser would change the java input into the python output

```
Output python statement: ho3_3 = Element("IY_HOLE3_3", ar3_3)
```

A similar code can be used for all geometry inputs
Summary

Following the Model-Based System Engineering Methodology, we do believe that:
1. Models are repositories of data and the link with reports should be maintained for the future use
2. Models should originate from a “single-source-of-truth”, so a single notebook
3. Python notebooks can be enhanced with simple analytical calculation (MIITs curve, RL circuit analysis, etc.)

Java to python parser is partially completed (cable parameters done, geometry in progress) and will be applied to migrate the remaining 12 STEAM-SIGMA inputs
1. What I liked - How simple it is, I came with basically no programming experience yet I was able to contribute and work
2. What I learnt – I learnt some programming and python skills (for loops, reading/writing files, etc.)