DIPARTIMENTO DI
INGEGNERIA
INDUSTRIALE

SCAD2015 — 5th Symposium on Collaboration in Aircraft Design
12-14 October, Naples, Italy

jPAD — A Java Toolchain of Computer Programs for Aircraft Design.

Software Engineering Best Practices Applied to Aerospace Sciences

A. De Marco

Università degli Studi di Napoli Federico II, Dept. Industrial Engineering, Aerospace Div.

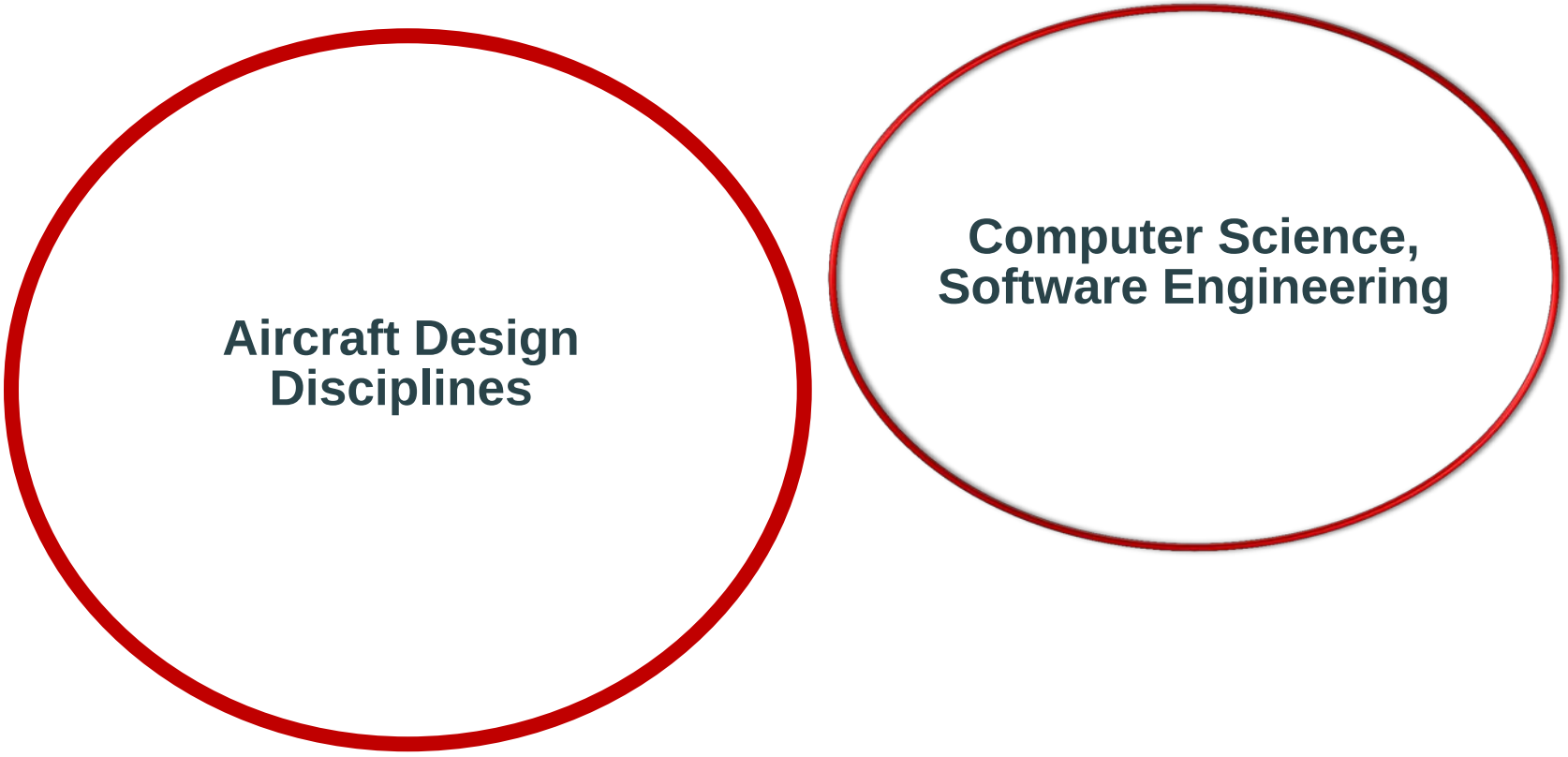
jPAD — Java Programs for Aircraft Design

- A software toolchain for aircraft preliminary design and MDO.
- A modern, user friendly, modular framework.
- Support for simultaneous management/analysis of several aircraft and/or ‘varied’ configurations of the same aircraft.
- Conceived for collaborative design activities.
- Interoperability with other tools/disciplines (CAD/CFD/FEM analysis).
- Started in early 2014. Approx. 120k lines-of-code.
- Planning to go open source.

What you can do with jPAD

- Define parametric representations of wings, fuselages and nacelles with XML configuration/input files (similar to OpenVSP).
- Generate CAD geometries of aircraft assembly and sub-components (Open CASCADE, JNI Technology). Measure lengths, areas, volumes. Export in CAD formats (Brep, STEP, IGES, STL, Collada).
- Vary geometric parameters and regenerate internal representation of geometries programmatically.
- Import CPACS aircraft configurations files (Tigl Native Interface) and extract relevant properties. (Work in progress)
- Perform various types of analysis (L0, L0.5, L1): Aerodynamics, Stability & Control, Performance, Weight, Costs. (Structural TBD)
- Exports analysis results in XML (native/CPACS) and Excel formats.

Competency spectrum



We are aerospace engineers ... Should we extend our capabilities?

Software engineering principles

- *Software design patterns*: formalized best practices that the programmer can use to solve common problems when designing an application or system.
- Design patterns speed up the development process by providing tested, proven development paradigms.
- Examples of patterns:
  - Builder, Factory, Dependency Injection,
  - Strategy, Separation-of-Concerns,
  - Observer (Publish/Subscribe)
- Micro patterns (design decisions in code): DRY (Don't Repeat Yourself), Sampler (Controlled Creation),

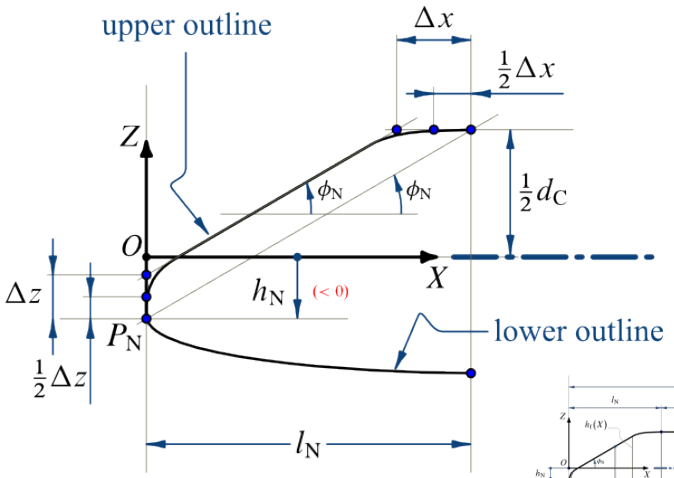
Example of Java code

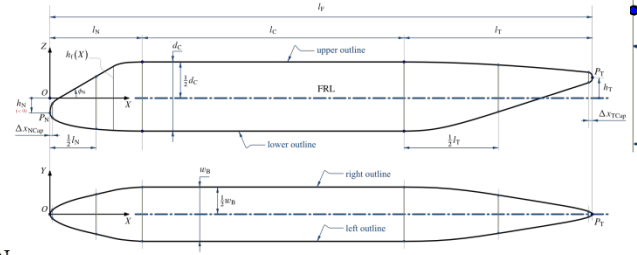

```
Aircraft aircraft = new Aircraft("Baseline_AC_AGILE.xml");
OperatingPoint op = OperatingPointFactory.getPoint("OP_001.xml");
ACAnalysisManager analysis = new ACAnalysisManager(op, aircraft,
                                                   AnalysisTypeEnum.AERODYNAMIC,
                                                   AnalysisTypeEnum.BALANCE,
                                                   AnalysisTypeEnum.WEIGHTS,
                                                   AnalysisTypeEnum.PERFORMANCE,
                                                   AnalysisTypeEnum.COSTS);
analysis.calculateGeometryAuxiliaryData();
analysis.runAnalysis(AnalysisTypeEnum.AERODYNAMIC,
                             AnalysisTypeEnum.PERFORMANCE);

DataWriter dataWriter = new DataWriter(analysis);
dataWriter.exportToXML("Analysis_Baseline_AGILE_op_001.xml");
dataWriter.exportToXLS("Analysis_Baseline_AGILE_op_001.xlsx");

CADBuilder cadBuilder = new CADBuilder(aircraft);
cadBuilder.build().exportToSTEP("Baseline_AGILE_op_001.stp");

CPACSBuilder cpacsBuilder = new CPACSBuilder(aircraft); // TBD
cpacsBuilder.build().export("Baseline_AGILE_op_001.xml");

WriteUtils.serializeObject(aircraft, aircraft.getName());
```

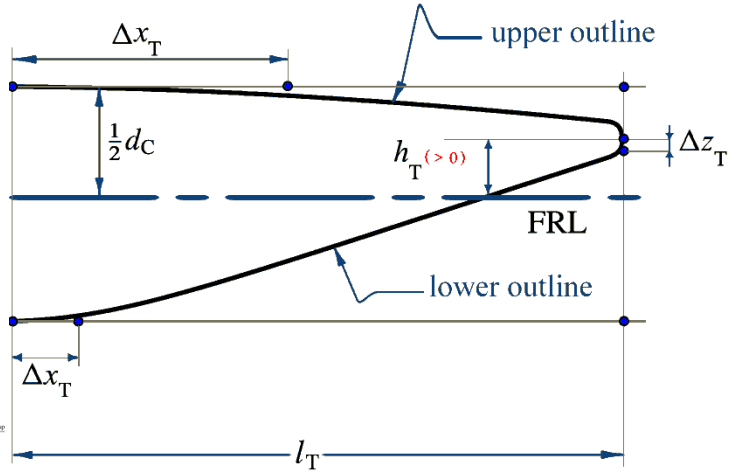
Java. Why?

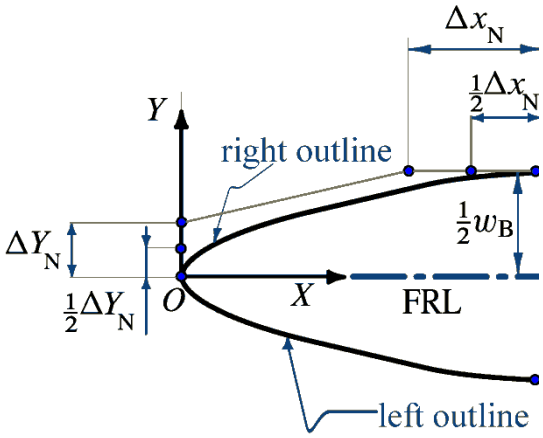
- “Compile once. Run it everywhere.” (well, almost)
- Widely supported, continuously updated and improved.
- Many open source libraries available, especially for I/O tasks and for complex mathematical operations.
- Widely supported GUI frameworks (SWT/JFace and JavaFX) and a GUI visual builders.
- Object-Oriented paradigm is naturally applied in the abstraction of typical Aircraft Design problems.
- Promotes modularity: easier to work with in an ever changing team.

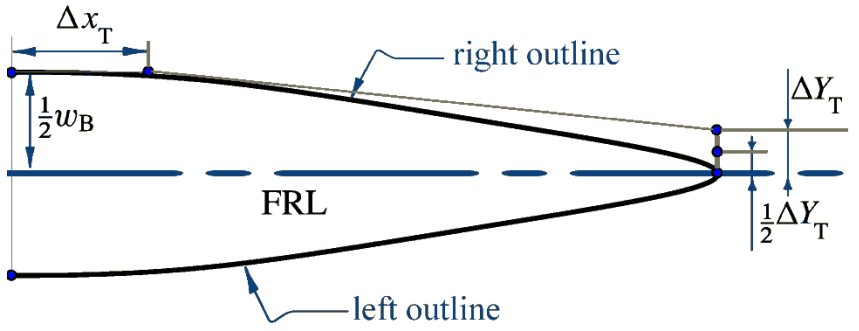
Parametric Fuselage in jPAD



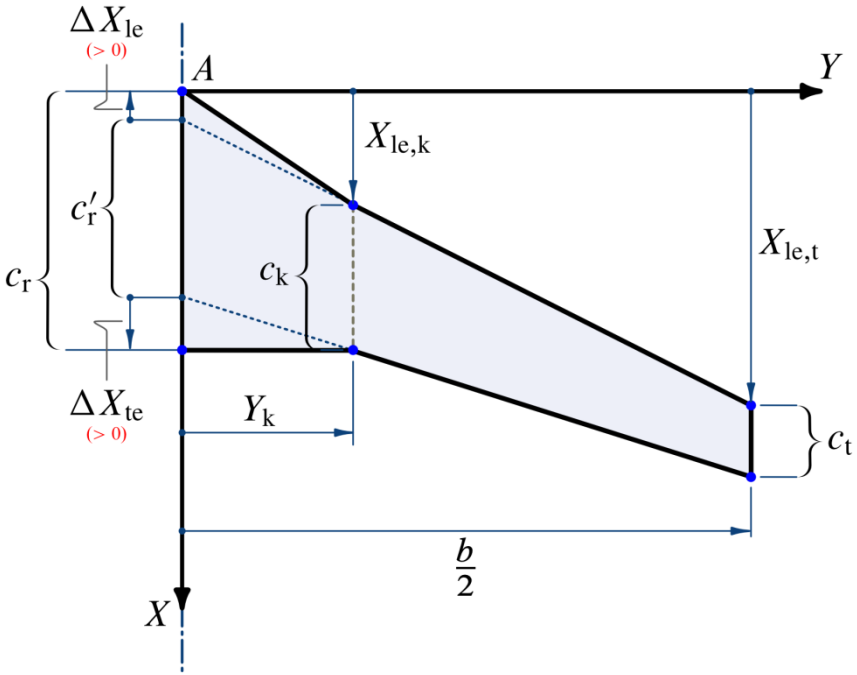


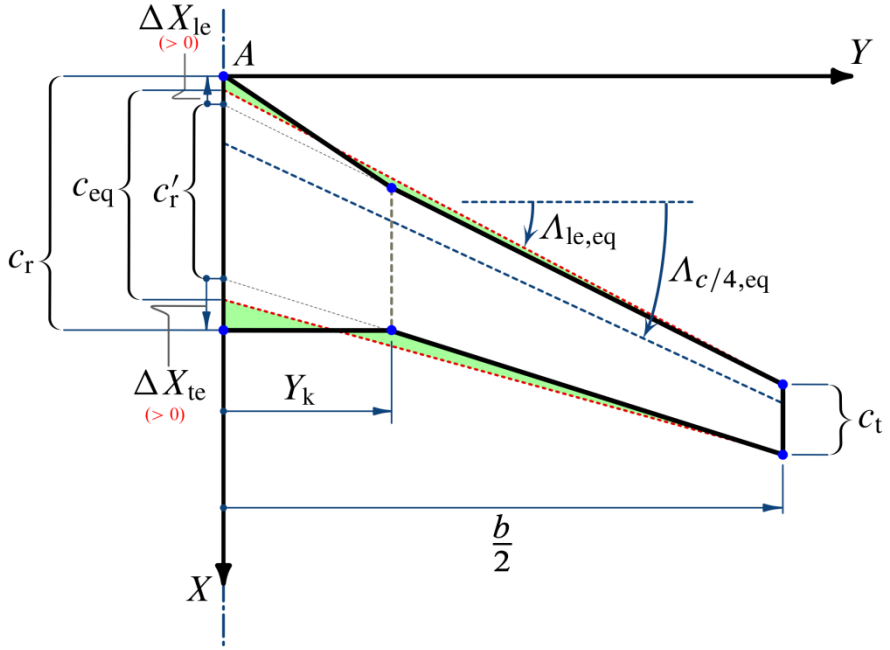






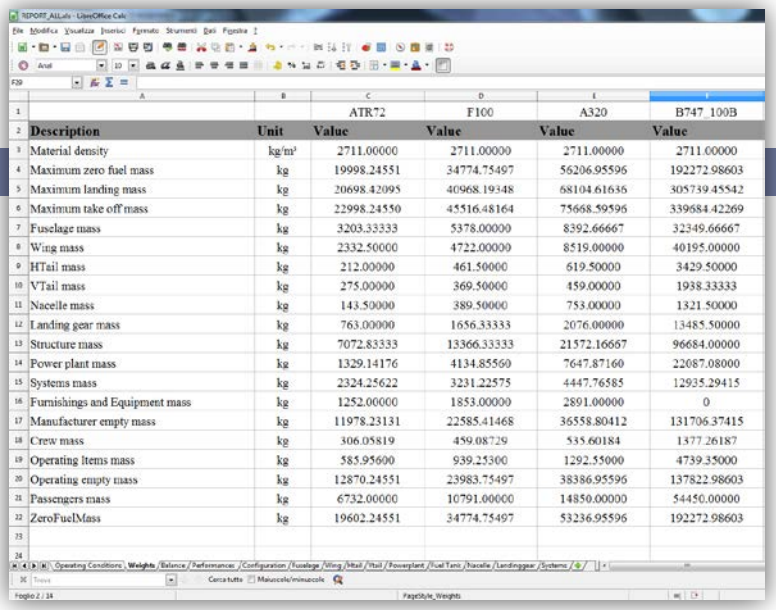
Parametric wing in jPAD



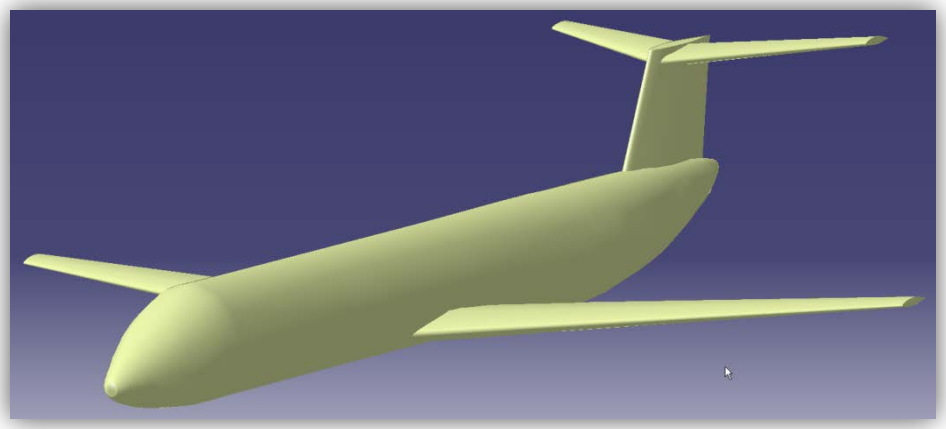


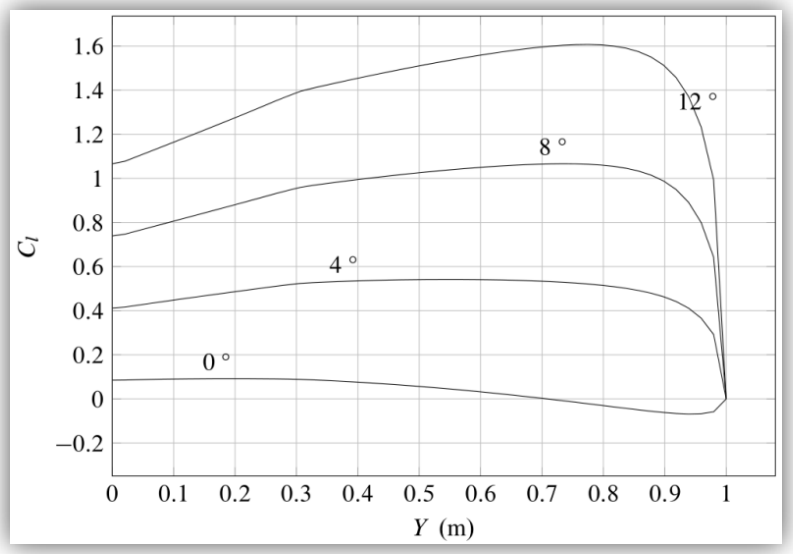
Output files

- XML
- Microsoft Excel
- Charts
- CAD model ready to be meshed

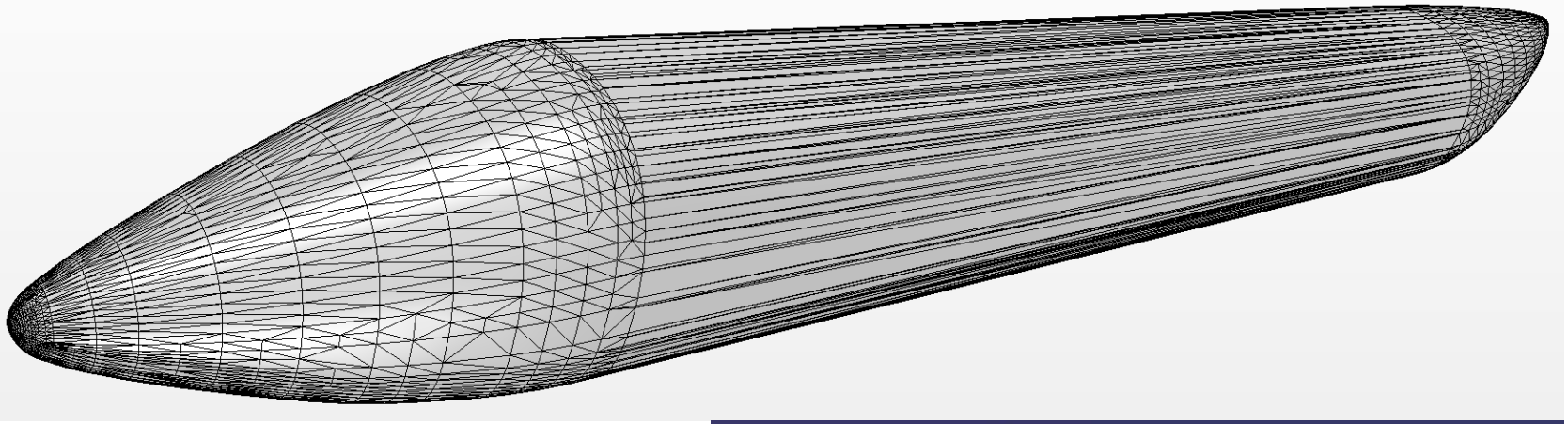


		ATR72	F100	A320	B747_100B
Description	Unit	Value	Value	Value	Value
Material density	kg/m³	2711.00000	2711.00000	2711.00000	2711.00000
Maximum zero fuel mass	kg	19998.24551	34774.75497	56206.95596	192272.98603
Maximum landing mass	kg	20698.42095	40968.19348	68104.61636	305739.45542
Maximum take off mass	kg	22998.24550	45516.48164	75668.59596	339684.42269
Fuselage mass	kg	3203.33333	5378.00000	8392.66667	32349.66667
Wing mass	kg	2332.50000	4722.00000	8519.00000	40195.00000
HTail mass	kg	212.00000	461.50000	619.50000	3429.50000
VTail mass	kg	275.00000	369.50000	459.00000	1938.33333
Nacelle mass	kg	143.50000	389.50000	753.00000	1321.50000
Landing gear mass	kg	763.00000	1656.33333	2076.00000	13485.50000
Structure mass	kg	7072.83333	13366.33333	21572.16667	96684.00000
Power plant mass	kg	1329.14176	4134.85560	7647.87160	22087.08000
Systems mass	kg	2324.25622	3231.22575	4447.76585	12935.29415
Furnishings and Equipment mass	kg	1252.00000	1853.00000	2891.00000	0
Manufacturer empty mass	kg	11978.23131	22585.41468	36558.80412	131706.37415
Crew mass	kg	306.05819	459.08729	535.60184	1377.26187
Operating Items mass	kg	585.95600	939.25300	1292.55000	4739.35000
Operating empty mass	kg	12870.24551	23983.75497	38386.95596	137822.98603
Passengers mass	kg	6732.00000	10791.00000	14850.00000	54450.00000
ZeroFuelMass	kg	19602.24551	34774.75497	53236.95596	192272.98603

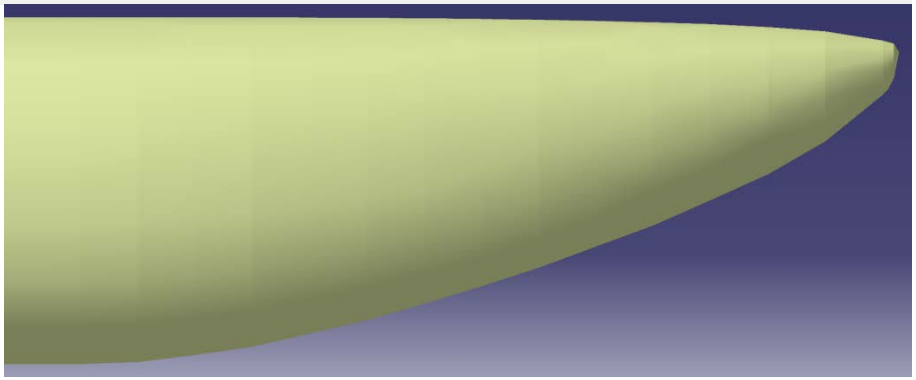




Example of jPAD output



External fuselage shape exported as STEP file



Example of jPAD output

Multiple aircraft analysis result comparisons

REPORT_ALL.xls - LibreOffice Calc

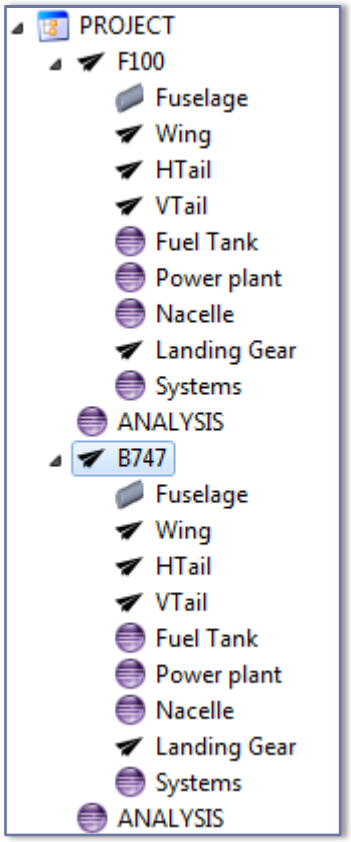
File Modifica Visualizza Inserisci Formato Strumenti Dati Finestra ?

	Description	Unit	ATR72 Value	F100 Value	A320 Value	B747_100B Value
3	Material density	kg/m³	2711.00000	2711.00000	2711.00000	2711.00000
4	Maximum zero fuel mass	kg	19998.24551	34774.75497	56206.95596	192272.98603
5	Maximum landing mass	kg	20698.42095	40968.19348	68104.61636	305739.45542
6	Maximum take off mass	kg	22998.24550	45516.48164	75668.59596	339684.42269
7	Fuselage mass	kg	3203.33333	5378.00000	8392.66667	32349.66667
8	Wing mass	kg	2332.50000	4722.00000	8519.00000	40195.00000
9	HTail mass	kg	212.00000	461.50000	619.50000	3429.50000
10	VTail mass	kg	275.00000	369.50000	459.00000	1938.33333
11	Nacelle mass	kg	143.50000	389.50000	753.00000	1321.50000
12	Landing gear mass	kg	763.00000	1656.33333	2076.00000	13485.50000
13	Structure mass	kg	7072.83333	13366.33333	21572.16667	96684.00000
14	Power plant mass	kg	1329.14176	4134.85560	7647.87160	22087.08000
15	Systems mass	kg	2324.25622	3231.22575	4447.76585	12935.29415
16	Furnishings and Equipment mass	kg	1252.00000	1853.00000	2891.00000	0
17	Manufacturer empty mass	kg	11978.23131	22585.41468	36558.80412	131706.37415
18	Crew mass	kg	306.05819	459.08729	535.60184	1377.26187
19	Operating Items mass	kg	585.95600	939.25300	1292.55000	4739.35000
20	Operating empty mass	kg	12870.24551	23983.75497	38386.95596	137822.98603
21	Passengers mass	kg	6732.00000	10791.00000	14850.00000	54450.00000
22	ZeroFuelMass	kg	19602.24551	34774.75497	53236.95596	192272.98603

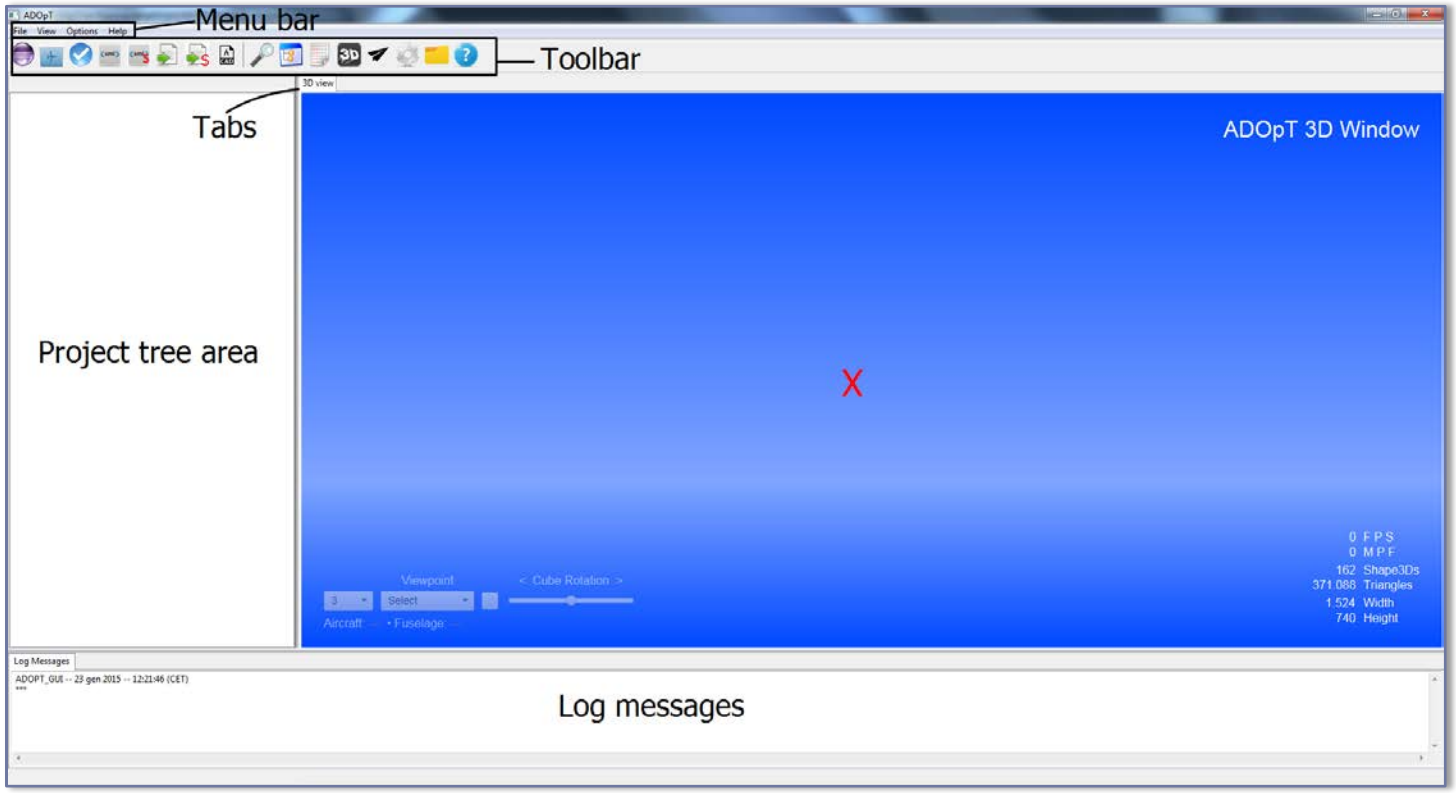
Operating Conditions | Weights | Balance | Performances | Configuration | Fuselage | Wing | Htail | Vtail | Powerplant | Fuel Tank | Nacelle | Landinggear | Systems

Foglio 2 / 14 PageStyle_Weights

ADOpT: a GUI for jPAD

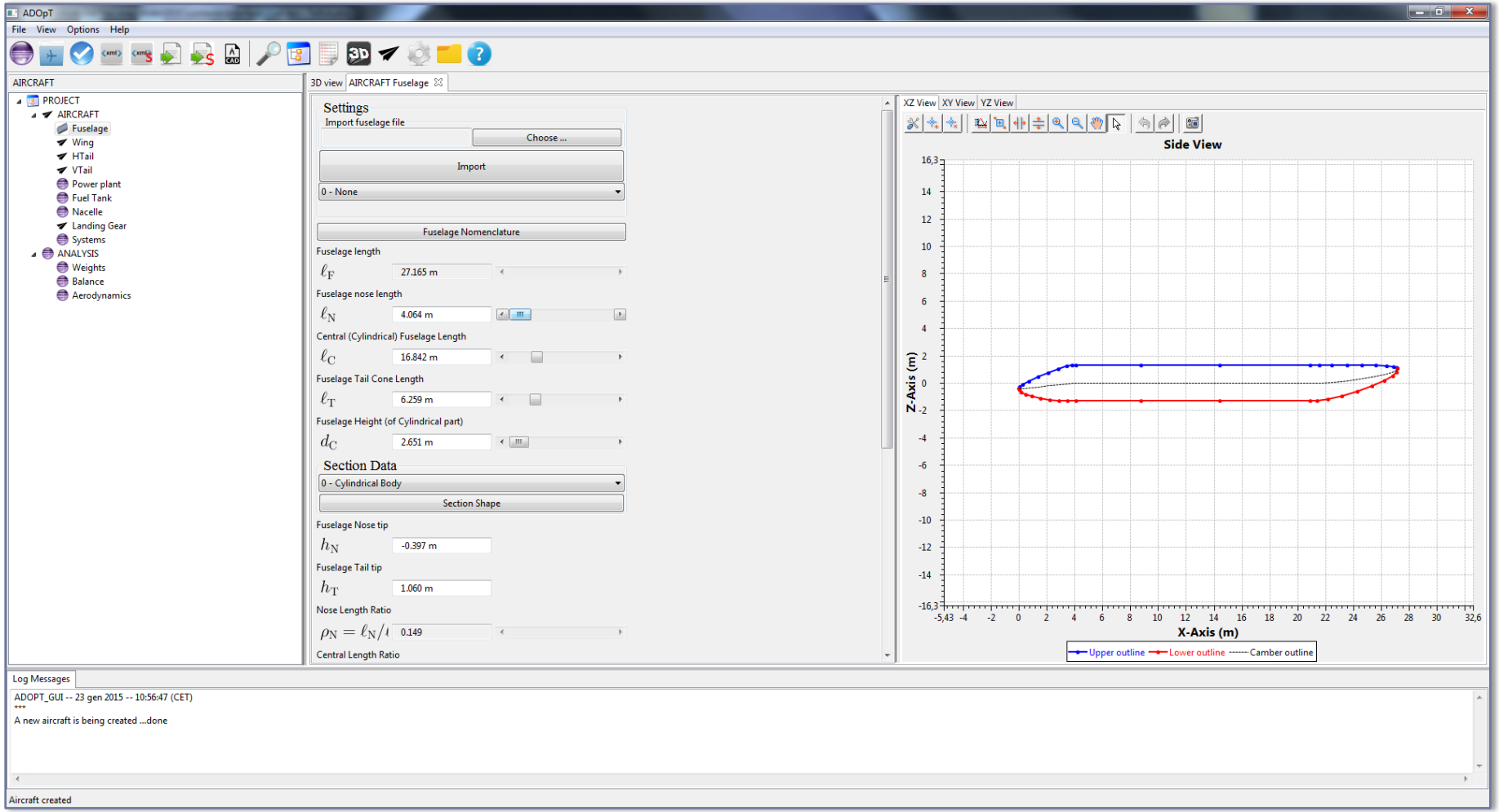


Example of project tree



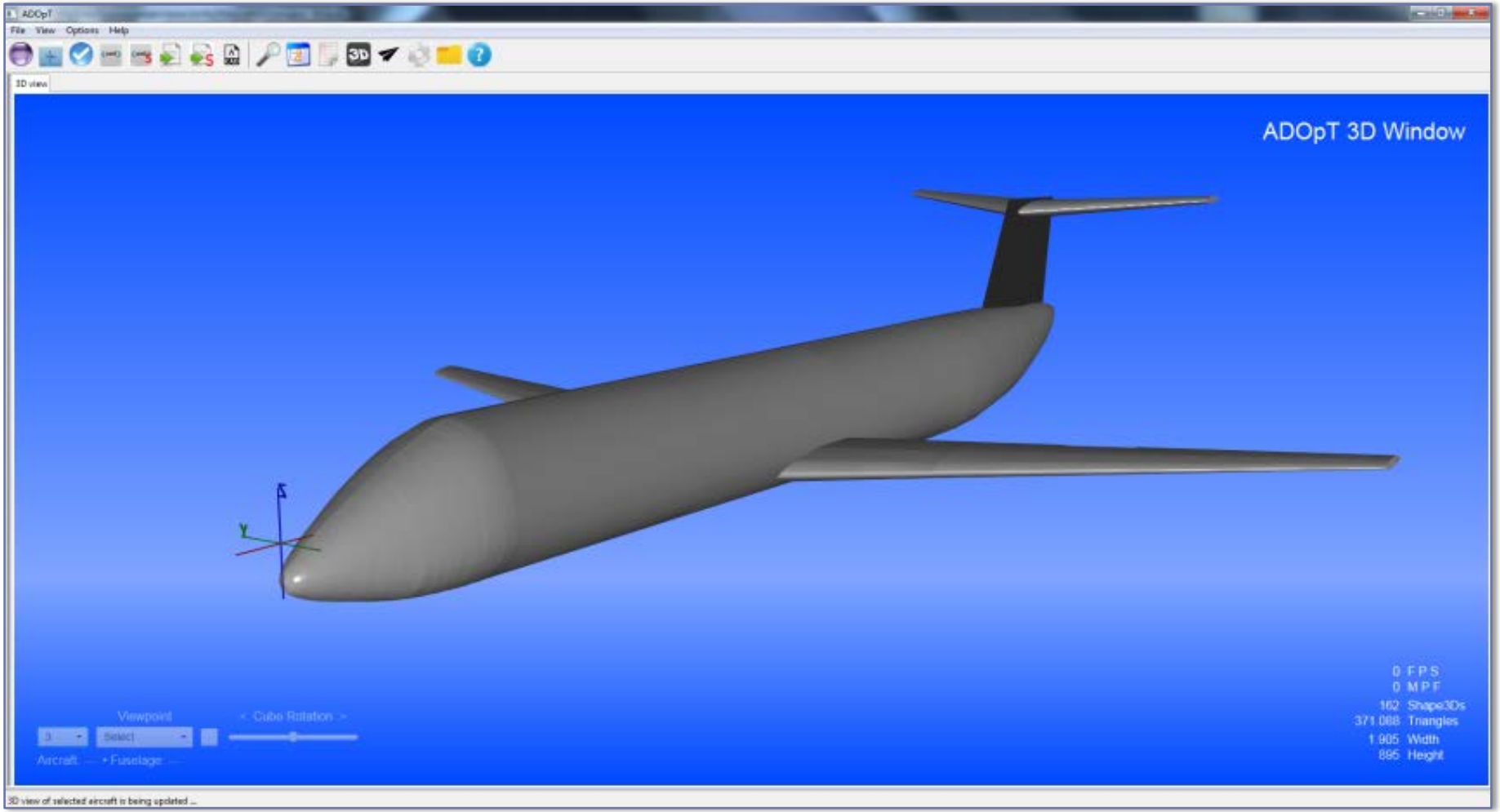
The GUI – Eclipse SWT/Jface technology, JavaFX for the 3D view

ADOpT: a GUI for jPAD



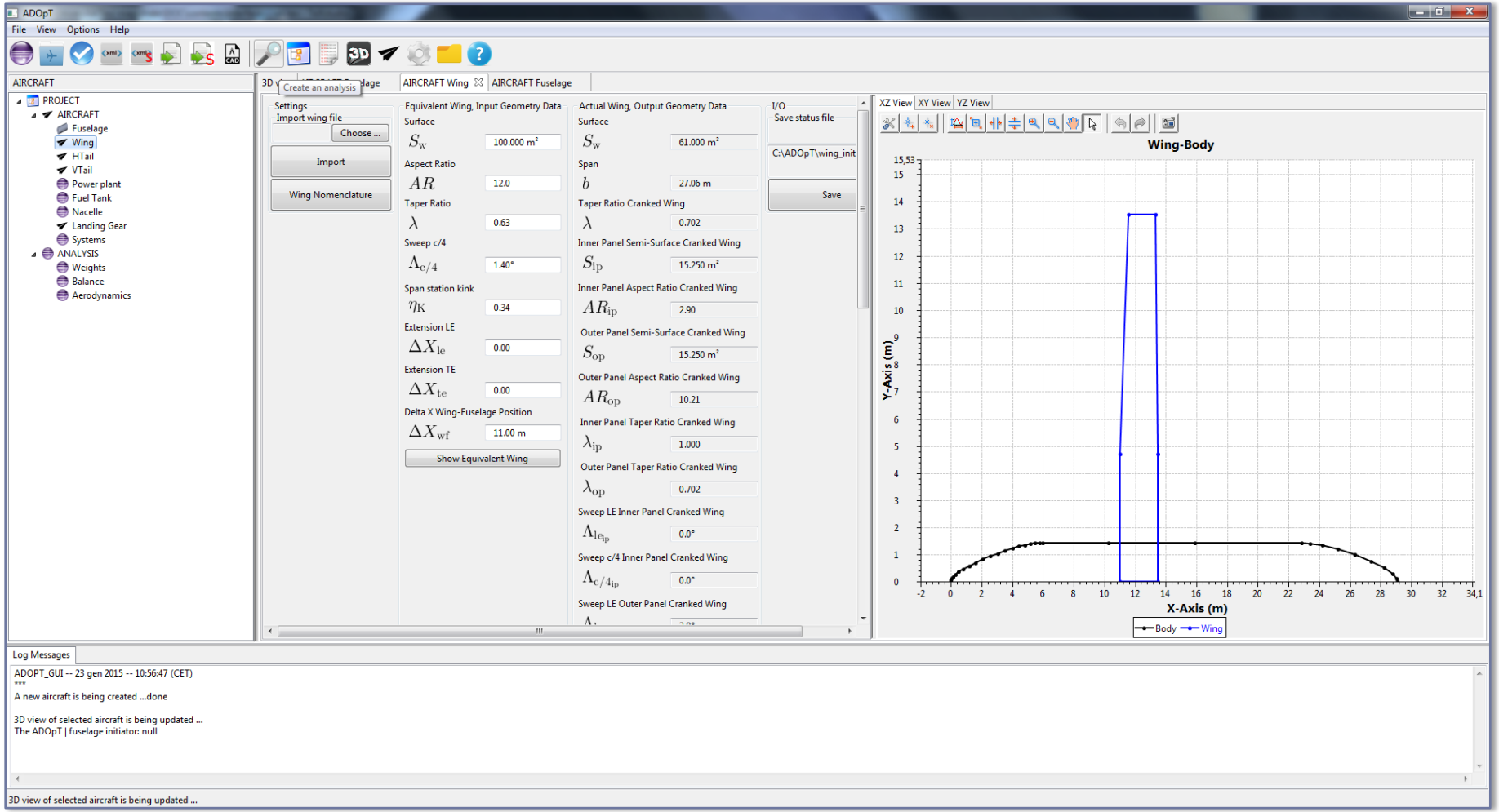
Varying geometric parameters

ADOpT: a GUI for jPAD



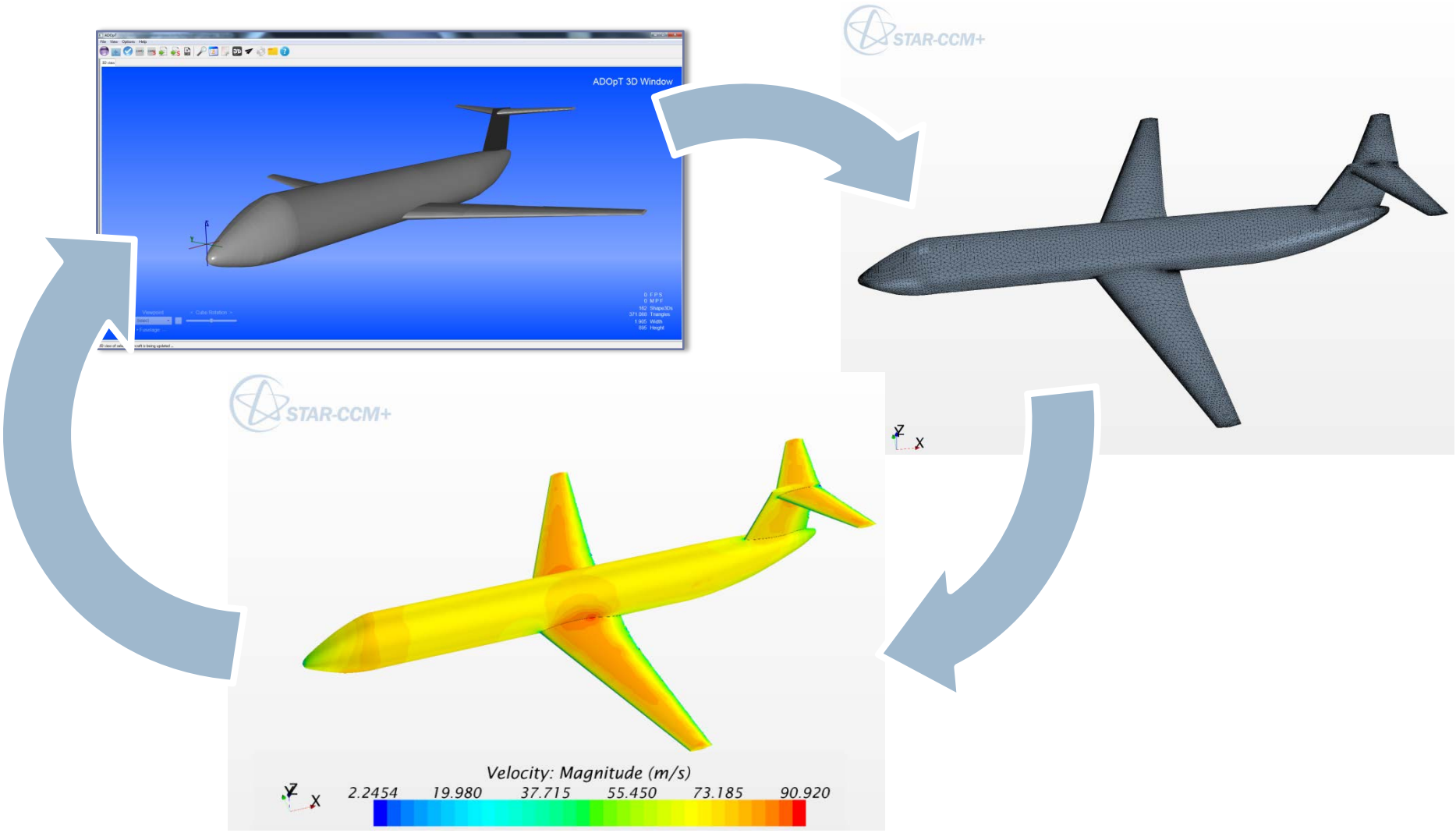
CAD generated from parametric representation

ADOpT: a GUI for jPAD



Configuring the aerodynamic analysis

Interface with external CFD tools



Key Points

- Java programming language (JDK ≥1.8)
- Eclipse IDE
- Great effort to make code easily maintainable
- In-house solutions for:
  - Importing/exporting data (including CPACS via Tigl interface)
  - Array handling
  - GUI design (SWT/JFace libraries, JavaFX)
  - Building the CAD model (via Open CASCADE libraries/Java Native Interface JNI)
  - CAD 3D viewer (JavaFX)
  - DATCOM-like database for Aerodynamics/Performances/Stability & Control analysis modules (Hierarchical Data Format, HDF)

Conclusions

- Main features and general arrangement complete
- Work in progress:
  - Loads
  - Stability & Control module
  - Collaborative features:
    - CPACS
    - United States Air Force Stability and Control Digital DATCOM
    - FlightGear simulator input XML file