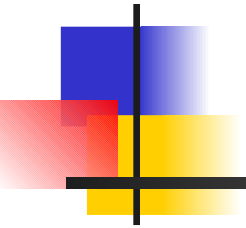
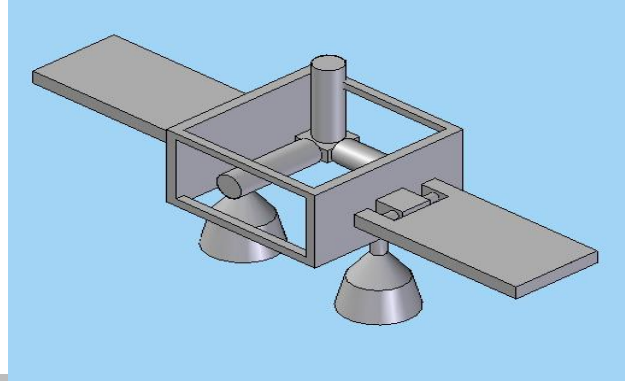
Integrating CAD models and Multi-disciplinary Simulation models

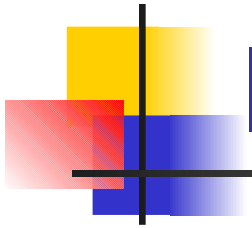


By
Chahé Adourian

Introduction



- CAD models are generally used to describe only mechanical aspects of a design
  - List of parts
  - Assembly information
  - Constraints between parts
    - axial joints
    - rigid links …
  - Dimensions, weight, material properties of parts
  - Etc.

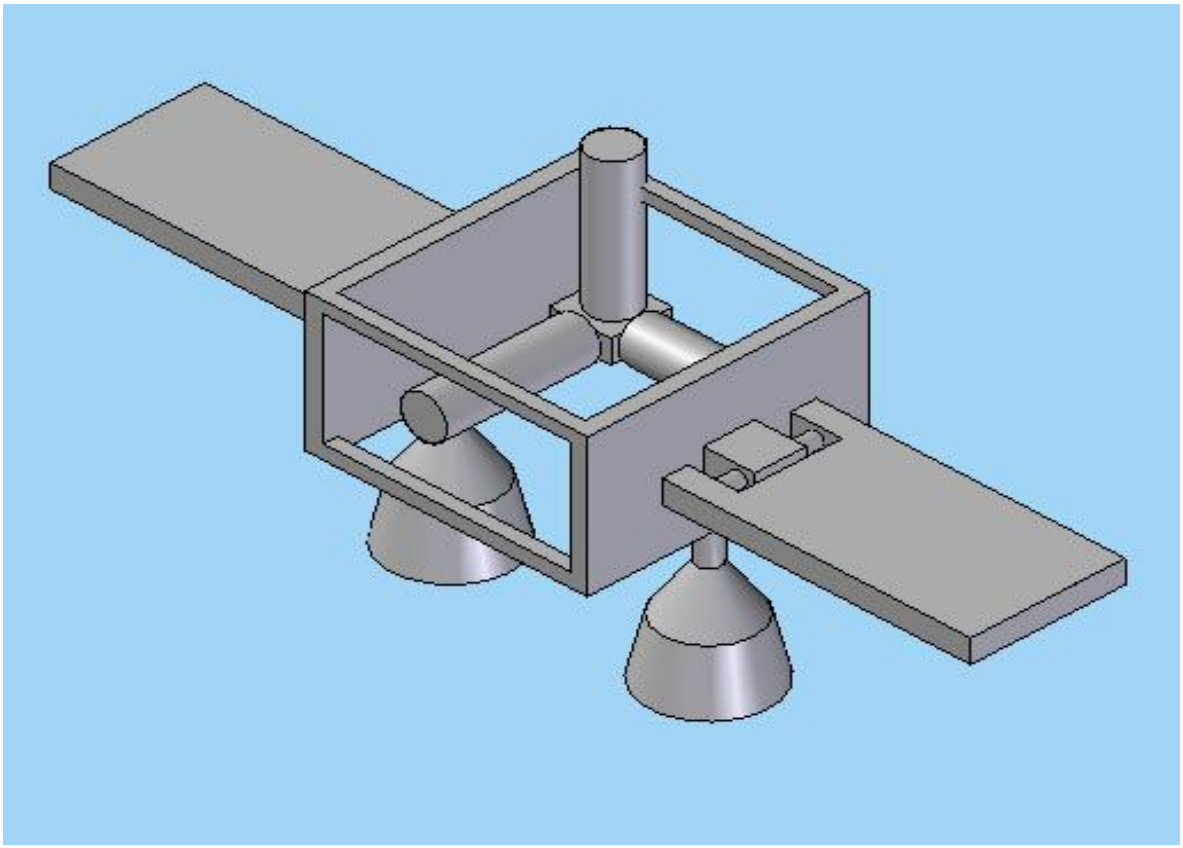


Introduction

- Want to make better use of the CAD model
- Go beyond what **traditionally** CAD models are used for in generating simulators
  - Ex. CAD model → Mechanical simulations
  - Ex. CAD model → Thermal simulations, etc.
- Use model to generate a **multidisciplinary** simulation of whatever the model represents

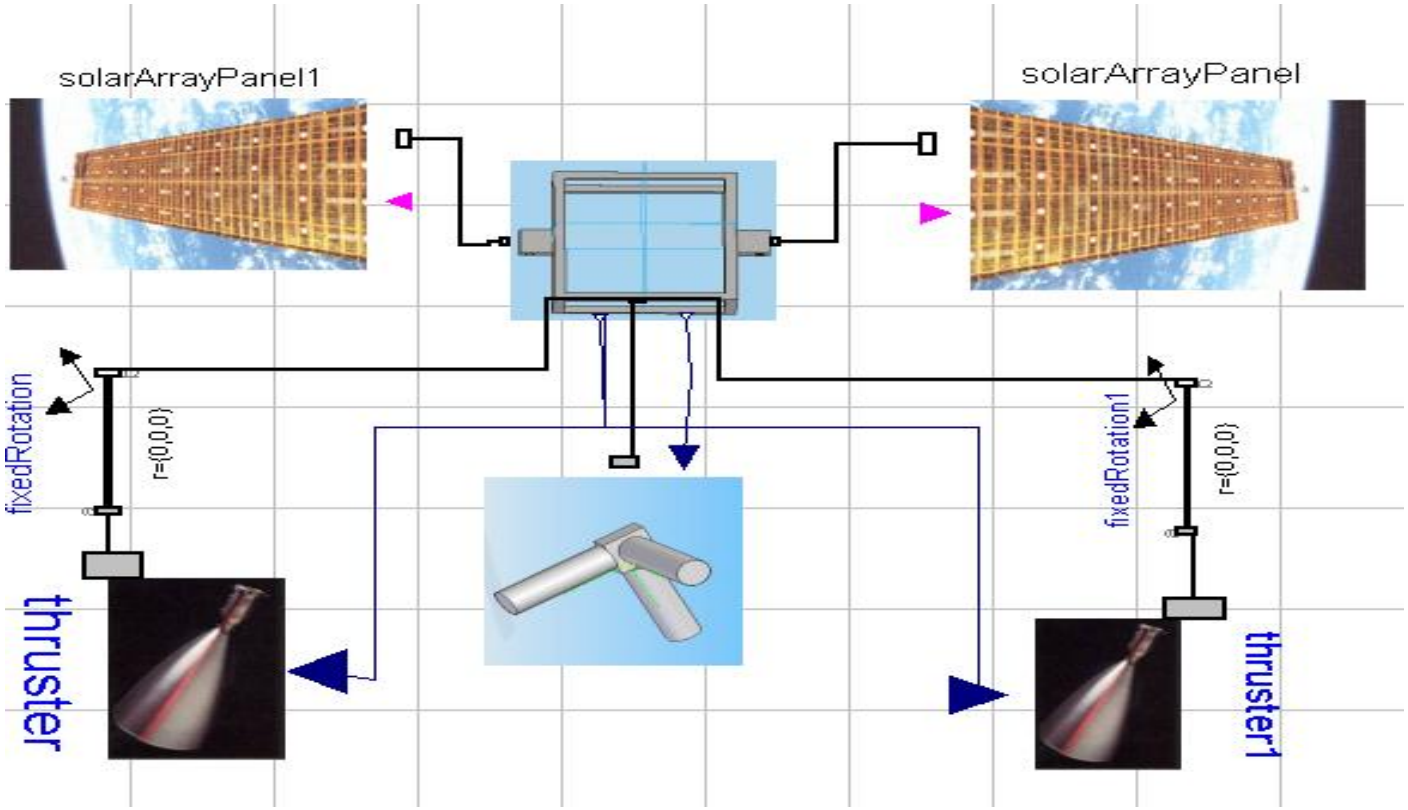
In Short!

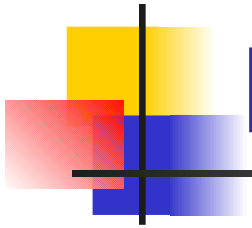
- We have drawn CAD model



In Short!

- We want a simulator!

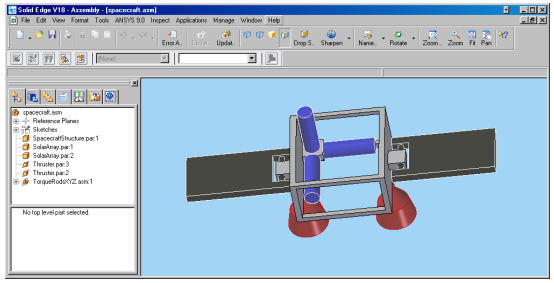




Process

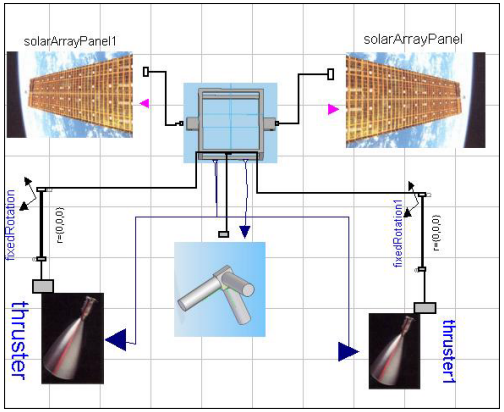
- Associate to each CAD part, a model in the simulator
- Each simulation model can have:
  - Mechanical behaviour
  - Electrical behaviour
  - Thermal behaviour
  - Control behaviour
- Each Simulation model must have
  - The behaviour intrinsic to the real part
  - An Equivalent Interface

Conceptual Design

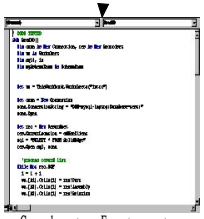


CAD model

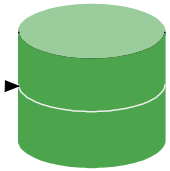
Code to generate mechanical simulation model



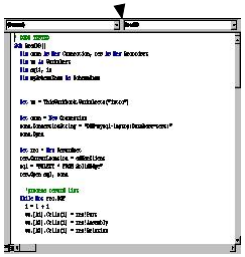
Mechanical Simulator model



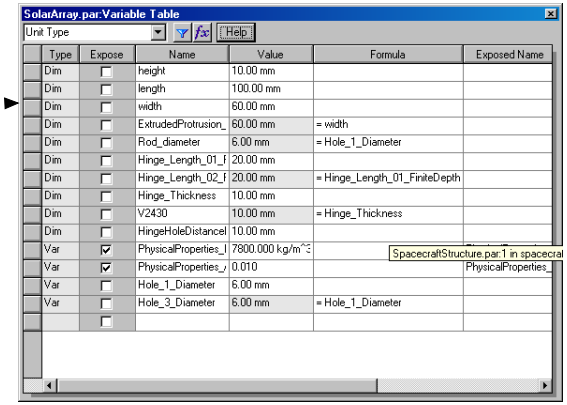
Code to Extract Mechanical Assembly Information



Database Storage



Code to generate User-Interface for parameter control



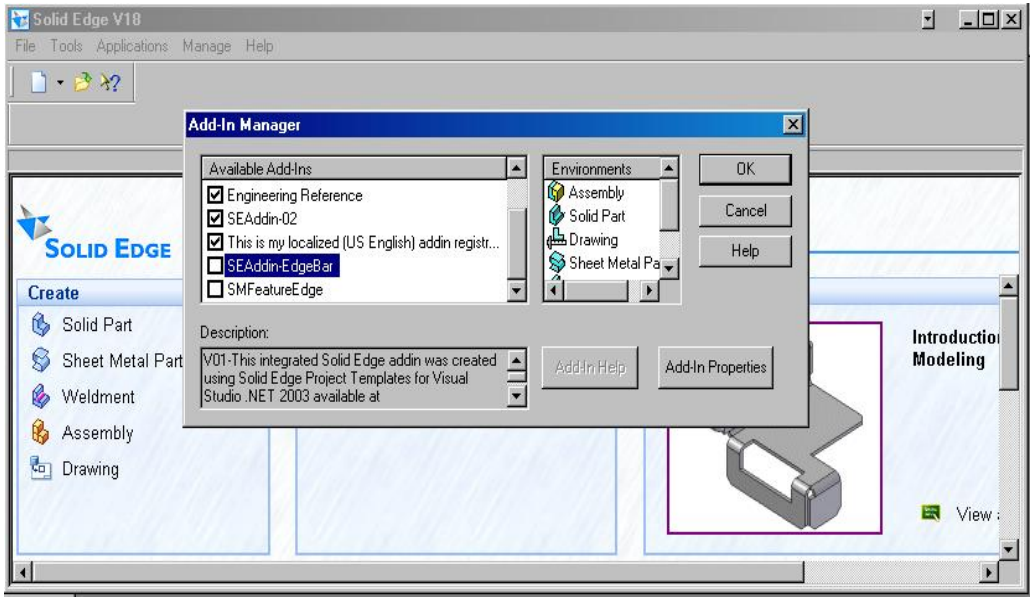
Customized User-Interface for Parameter Access and Control

Conceptual Design

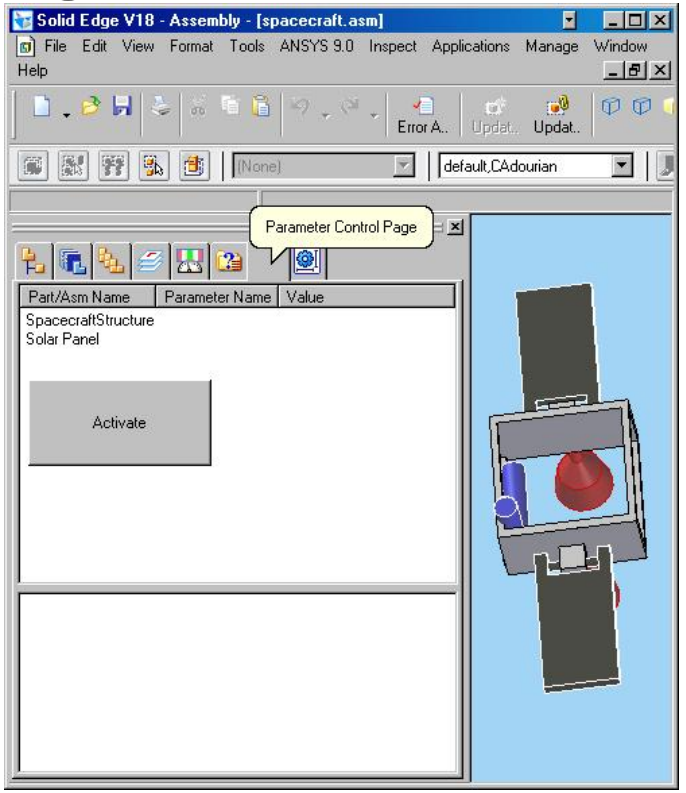
- There are four main parts
  1. CAD
     - A UI to save CAD model assembly and part information, into the database
     - A UI to control the CAD model parameters
  2. DB: design the database tables
  3. CAD to Simulation Mapping
  4. SIM (not presented)
     - Interface to Access the DB
     - Extract Assembly information
     - Generate simulation files
     - Load model and parameter table

CAD Component
Integrate User-interfaces into CAD

- Adding a UI to SolidEdge



Adding the created UI to SolidEdge



SolidEdge, with a new UI

CAD component
Extracting Assembly information

Occurrence: SolarArray.par:1
Origin (m)= 2.99999999999998E-02, -7.98718597857838E-02, 3.37396753273139E-02
Transformation Matrix (Rotation + Translation)
1, -1.77993775588808E-14, 1.79856129989277E-14, 2.99999999999998E-02
1.86235440046908E-16, -1, 1.22460635382239E-16, -7.98718597857838E-02
-1.7626110851032E-30, 1.78631523635452E-14, -1, 3.37396753273139E-02
0, 0, 0, 1

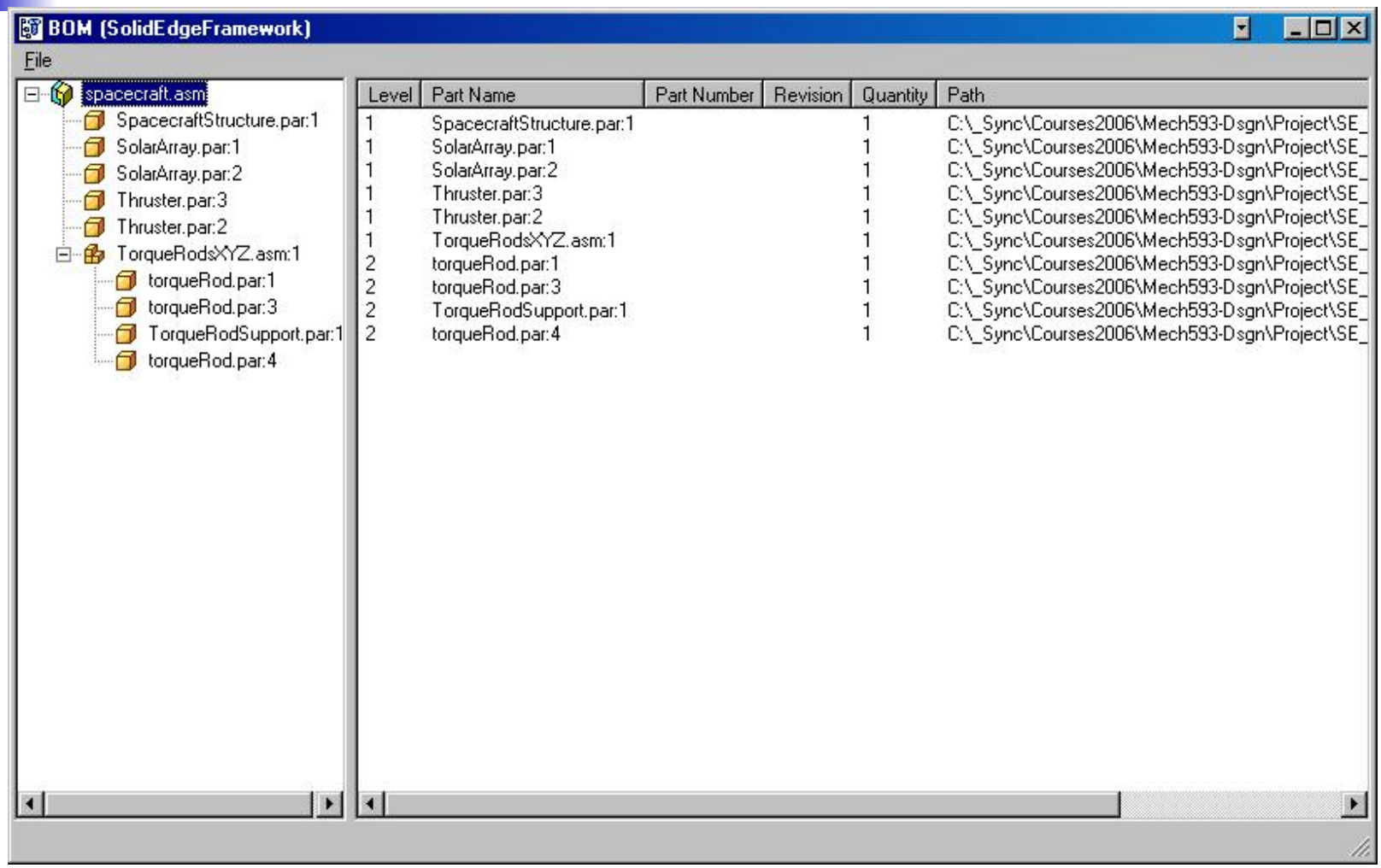
Physical Properties:
Mass (Kg):= .50802159217128
Volume (m^3):= 6.51309733552924E-05
Area (m^2):= 1.74974335690969E-02
CenterOfMass in Part Frame (m):= -3.00000000000007E-02, 4.52732412021825E-02, .005
CenterOfVolume in Part Frame (m):= -3.00000000000007E-02, 4.52732412021825E-02, .005
Moments in Part Frame(Kg m^2):=
Ixx= 1.58194239618618E-03 Iyy= 6.35436033630346E-04 Izz= 2.18361765304319E-03
Ixy=-6.89993522348629E-04 Ixz=-7.62032388257062E-05 Iyz= 1.14998920391449E-04
PrincipalAxis:=
x: 0, 0, 1
y: 1, 0, 0
z: 0, 1, 0
PrincipalMoments (Kg m^2):= 6.85123447914599E-04, 5.2796708420748E-04, 1.6551606087189E-04
RadiiOfGyration (m):= 3.67234377861003E-02, 3.22375729674831E-02, 1.80500739431201E-02

Relations:
Axial : SolarArray.par:1 SpacecraftStructure.par:1
Geometry1 Info
Position1: -0.030, -0.070, 0.029
Vector1: 1.000, 0.000, 0.000
Geometry2 Info
Position2: -0.185, -0.070, 0.029
Vector2: 1.000, 0.000, 0.000

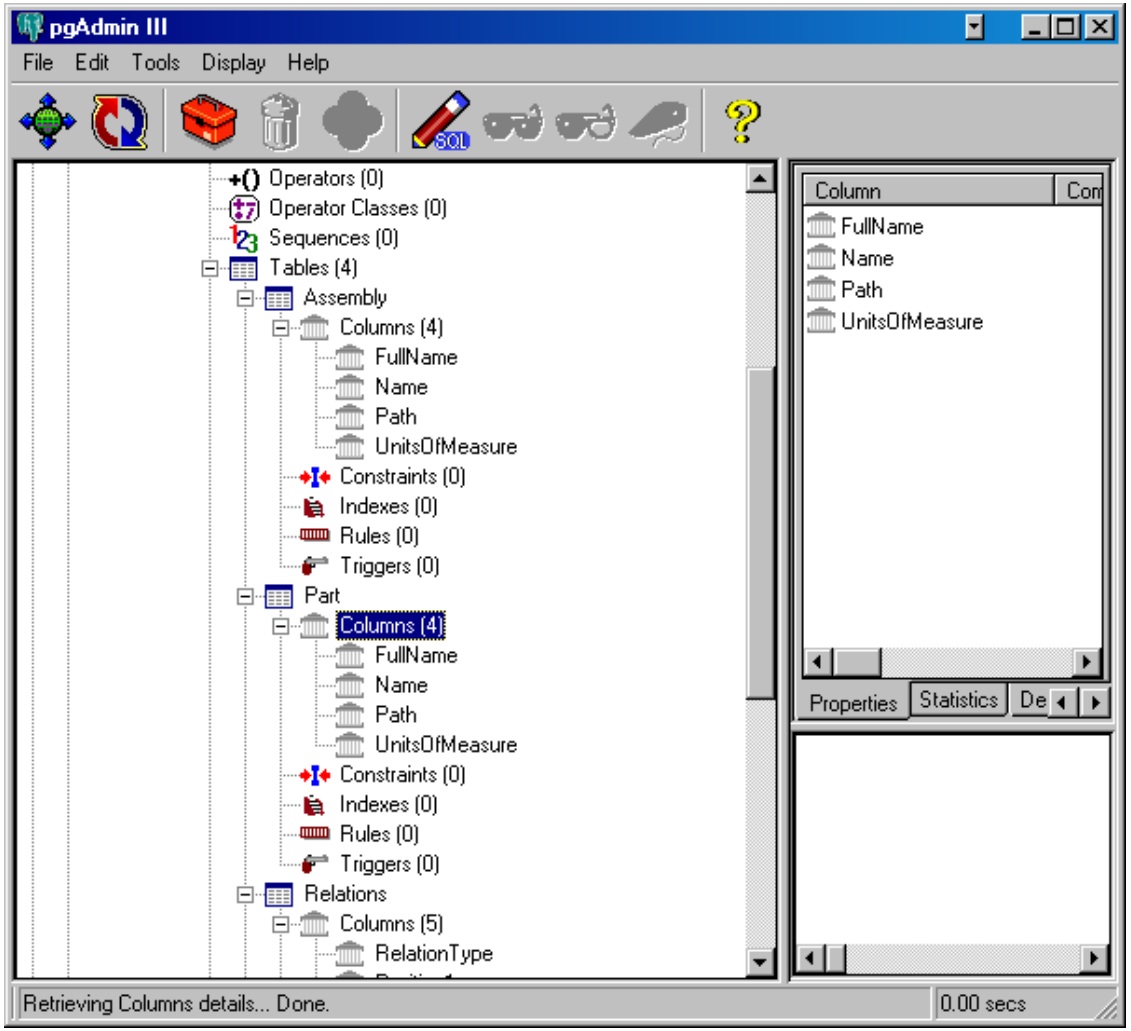
Planar : SolarArray.par:1 SpacecraftStructure.par:1
Geometry1 Info
Position1: -0.020, -0.060, 0.034
Vector1: 1.000, 0.000, 0.000
Geometry2 Info
Position2: -0.010, -0.076, 0.024
Vector2: -1.000, 0.000, 0.000

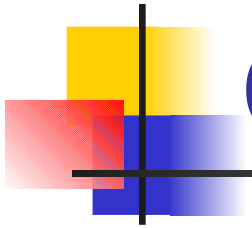
- Part:
  - Name, position, rotation, …
- Physical Properties
  - Mass, Volume…
- Relations
  - Axial joint

CAD Component
User-Interface to control CAD parameters



DB Component
Brief look at the Database tables



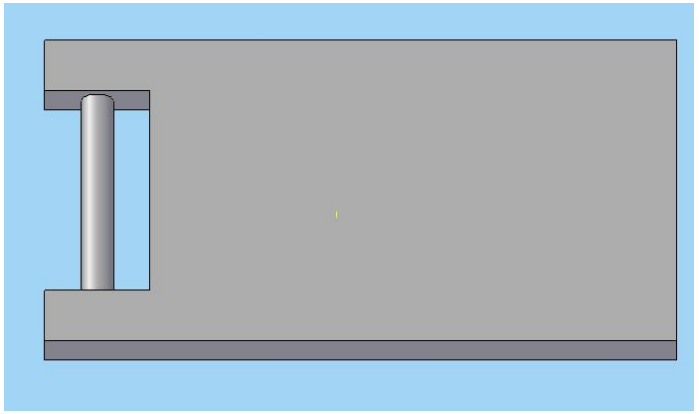


CAD to Simulation Mapping

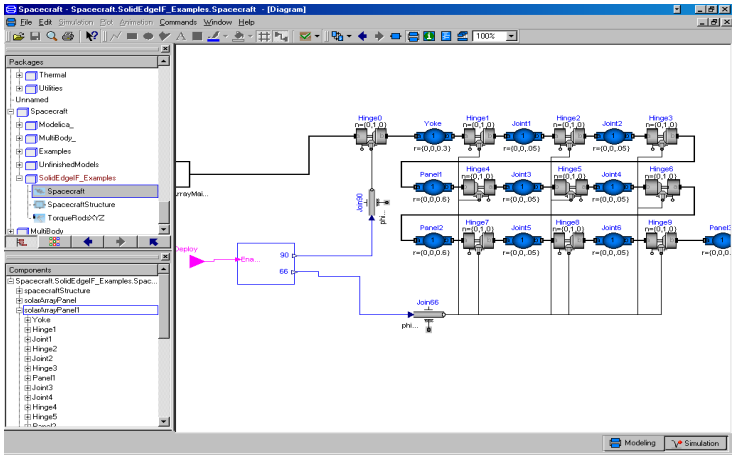
- The mapping is an association between
  1. CAD Assemblies and Parts on the one hand
  2. And Modelica models on the other
- Must be careful when both Assemblies and Part have modelica equivalents

Example Mapping

- CAD Solar Panel

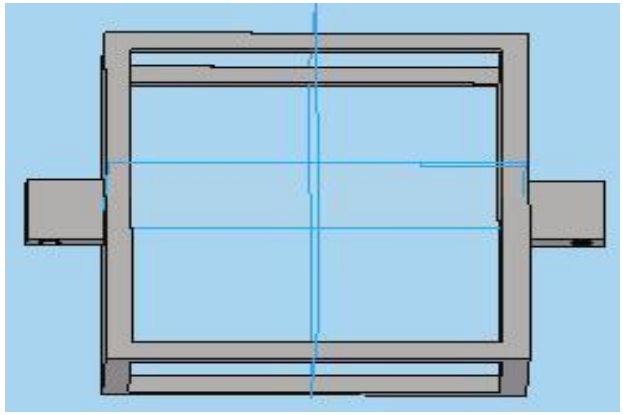


- Simulator Solar Panel

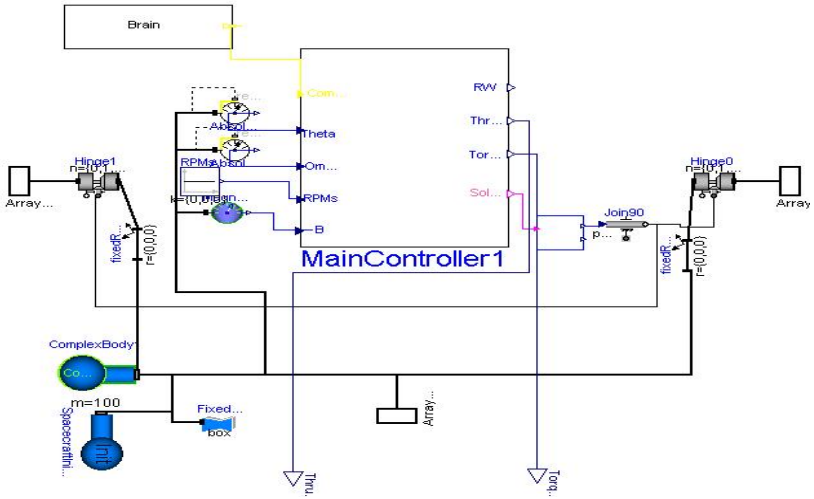


Example Mapping

- CAD Spacecraft Structure

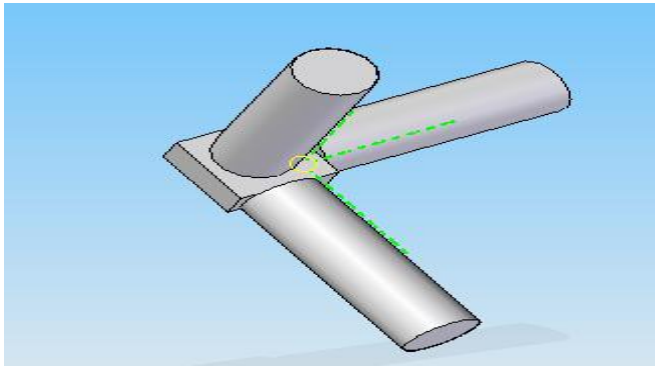


- Simulator S/C Structure

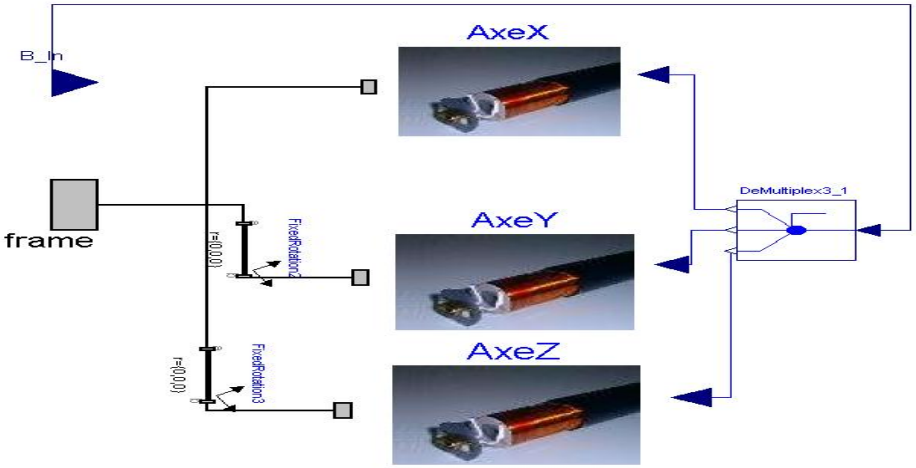


Example Mapping

- CAD Torque Rod Assembly

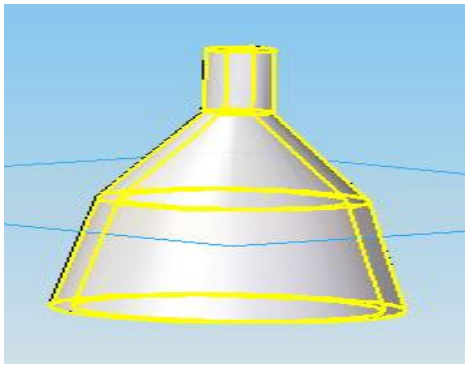


- Simulator Torque Rod Assembly

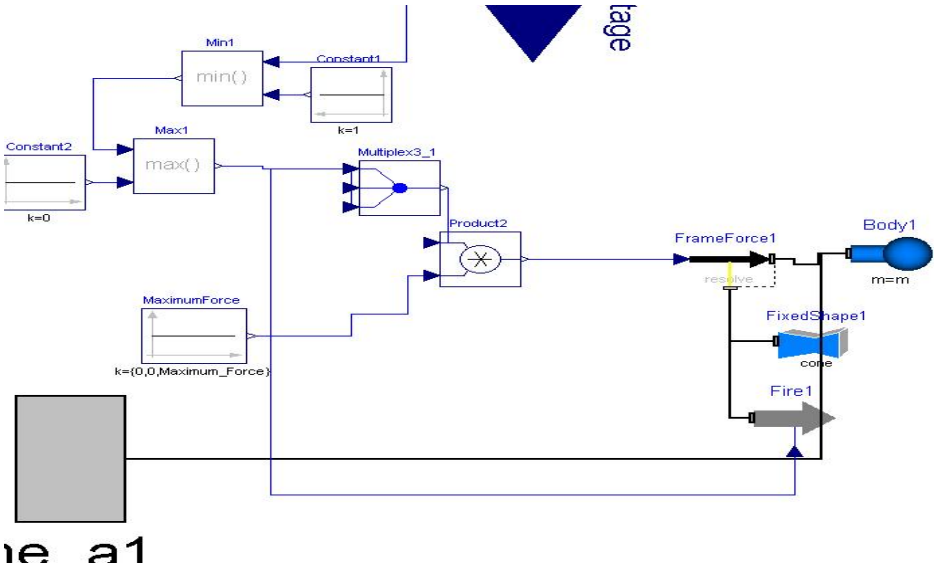


Example Mapping

- CAD Thruster

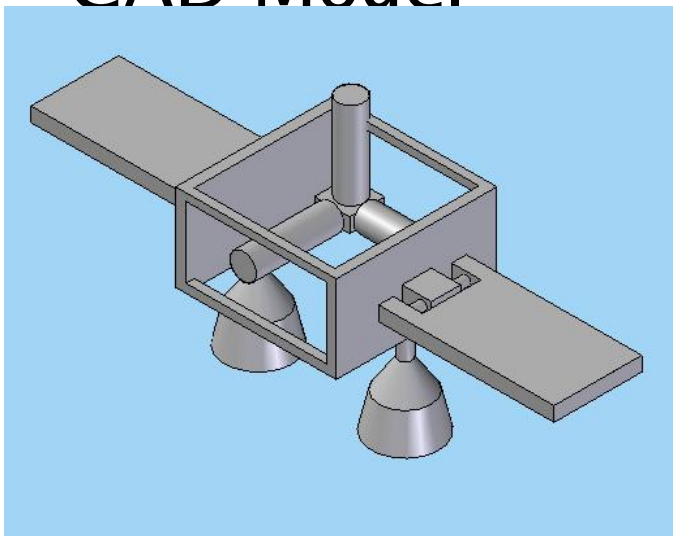


- Simulator Thruster

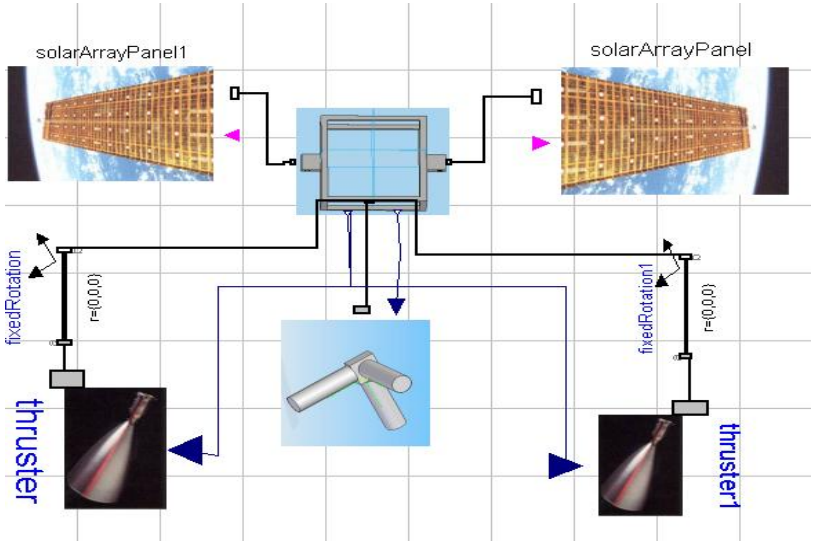


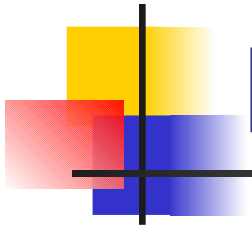
Expected Conversion Result

- CAD Model



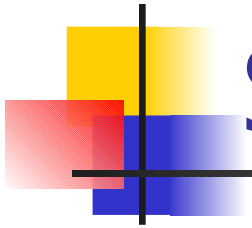
- Modelica Model





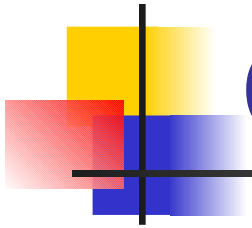
Expected Conversion Result

- CAD Demo
- Modelica Demo



Status

- All major technical components of the design have been exercised
- No foreseeable technical difficulties
- Conversion from CAD to Modelica now possible, however reverse process requires an upcoming release of Dymola (or MSDL compiler)



Conclusion

- Approach looks promising
  - CAD model translated to multidisciplinary simulation model
  - Will eventually allow to completely link CAD model to all related simulation models
- Much work still required
  - Modeling all possible user behaviors on both the CAD and Modelica tools
  - Reevaluate system architecture accordingly
  - Continue developing the interfaces