COMPLETE MODEL STORAGE – A SOLVER-AGNOSTIC HIERARCHICAL APPROACH

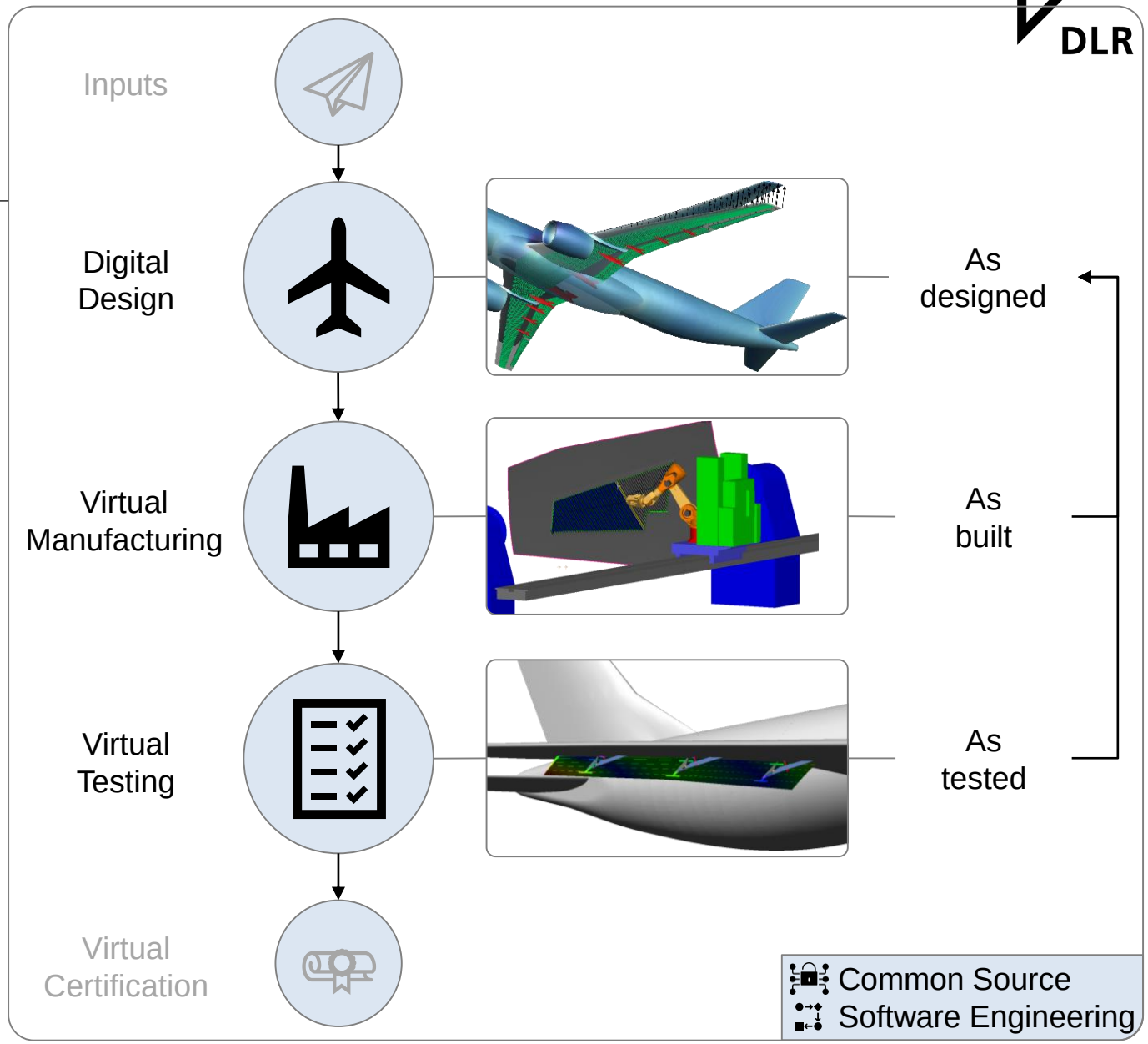
VMAP Working Group „Complete Model Storage“
19.09.2022
Martin Rädel

DLR

TASK/MOTIVATION

Task

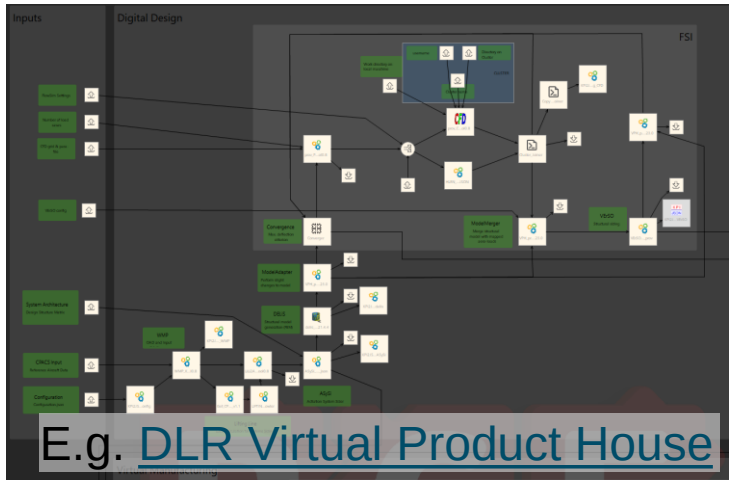
- Virtual product development
- Multiple simulation steps
- Methods e.g.:
  - Numerical
    - FEM
    - Peridynamics
    - …
  - Semi-analytical
    - Ritz
    - …
  - Analytical
- Central for each method: Model



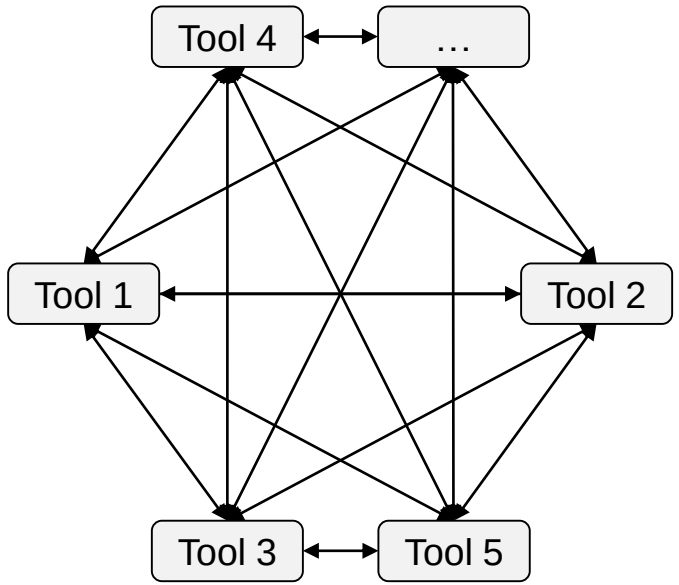
Task
Example: Digital design process aircraft moveable

- Each tile
  - Assessment capabability → Process
  - Requires input, delivers output → I & O

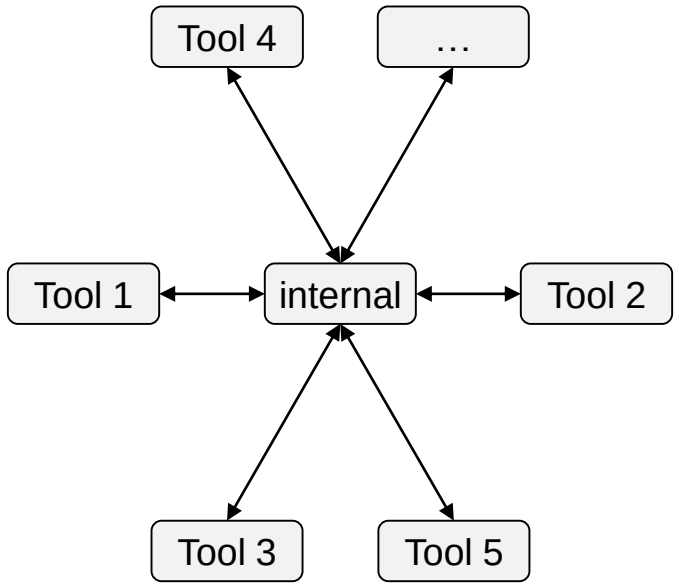
  IPO
- Basically same information, vendor-specific formats



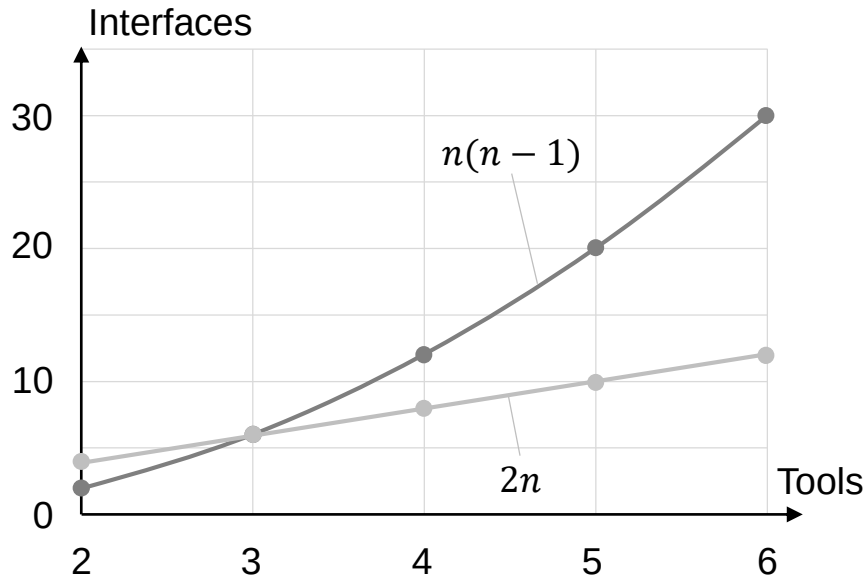
E.g. DLR Virtual Product House



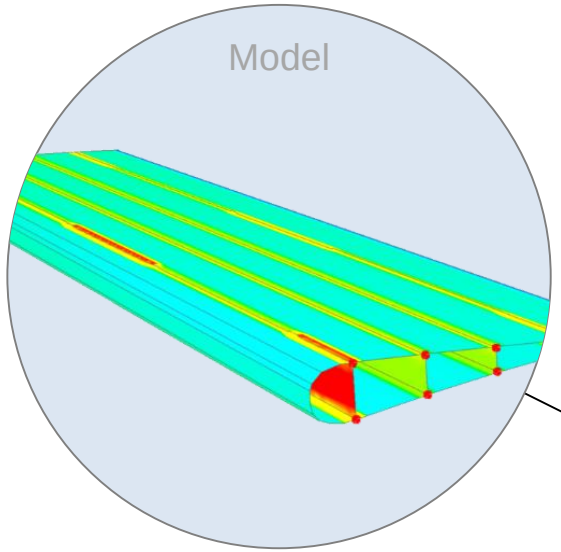
Individual data standards



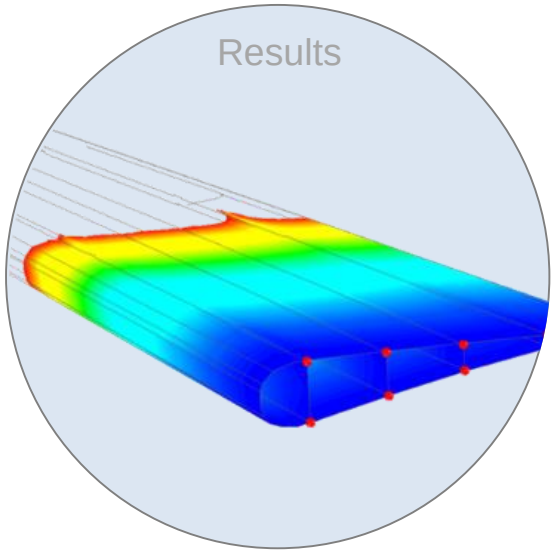
Common data standard



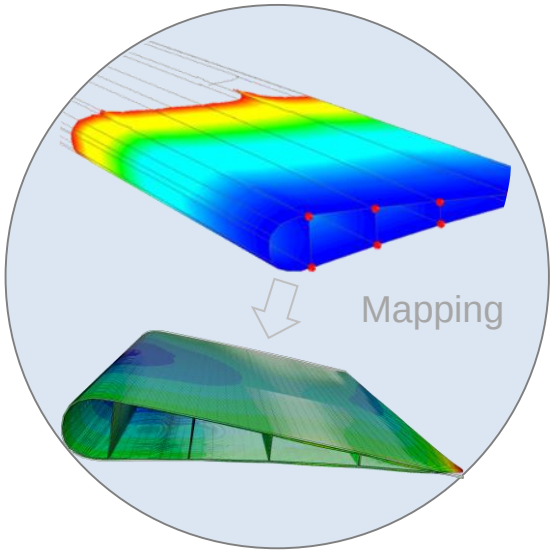
Task
Steps



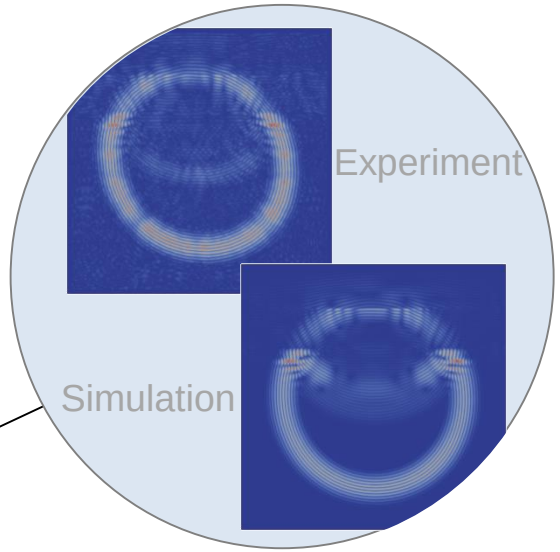
Parametric model generation
→ Input



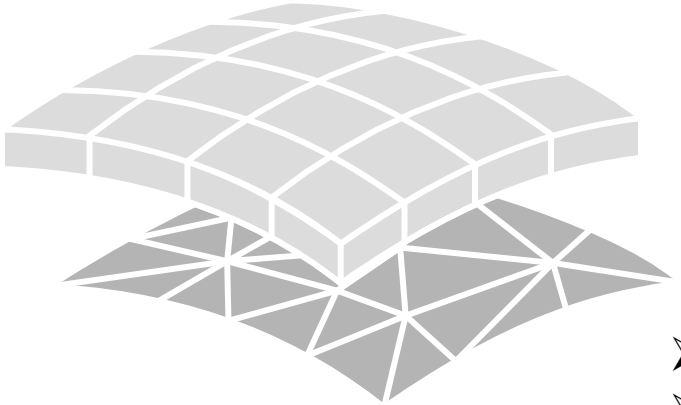
Postprocessing
→ Output



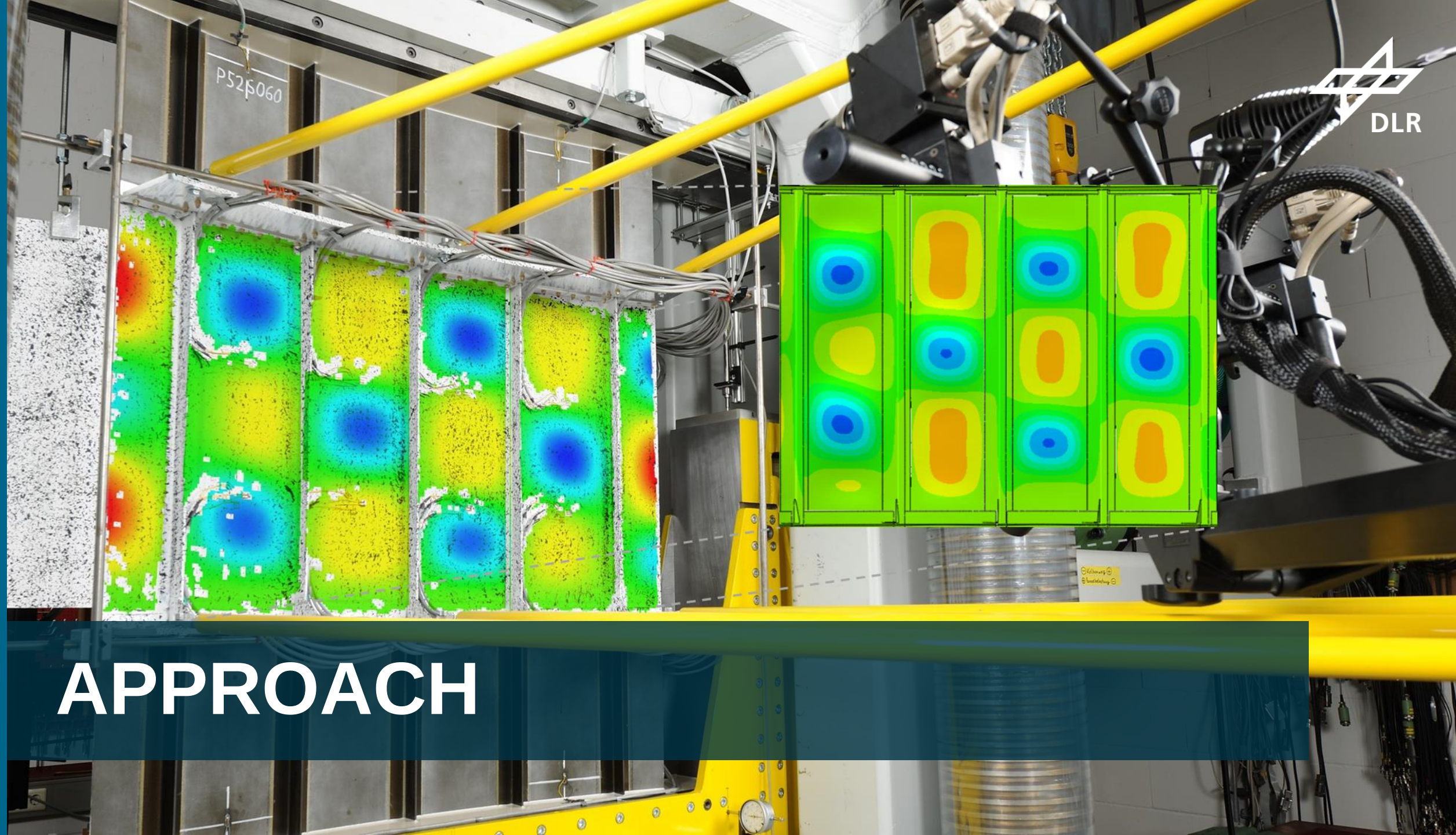
State transfer
→ Next input



Spatial & temporal measurements



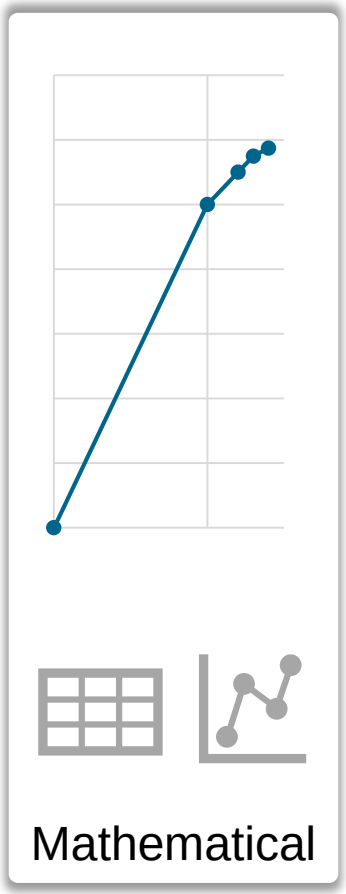
- Tool/solver-agnostic
- Standardized

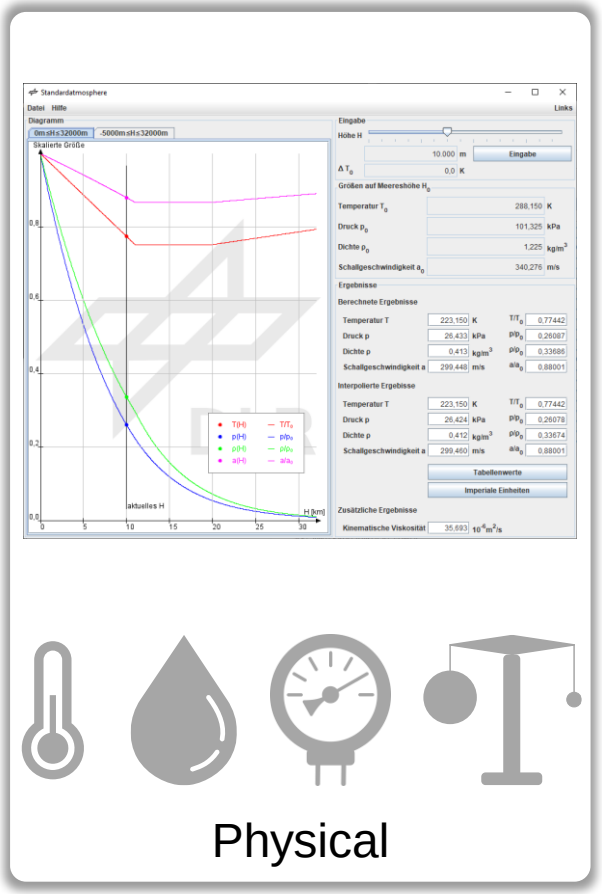


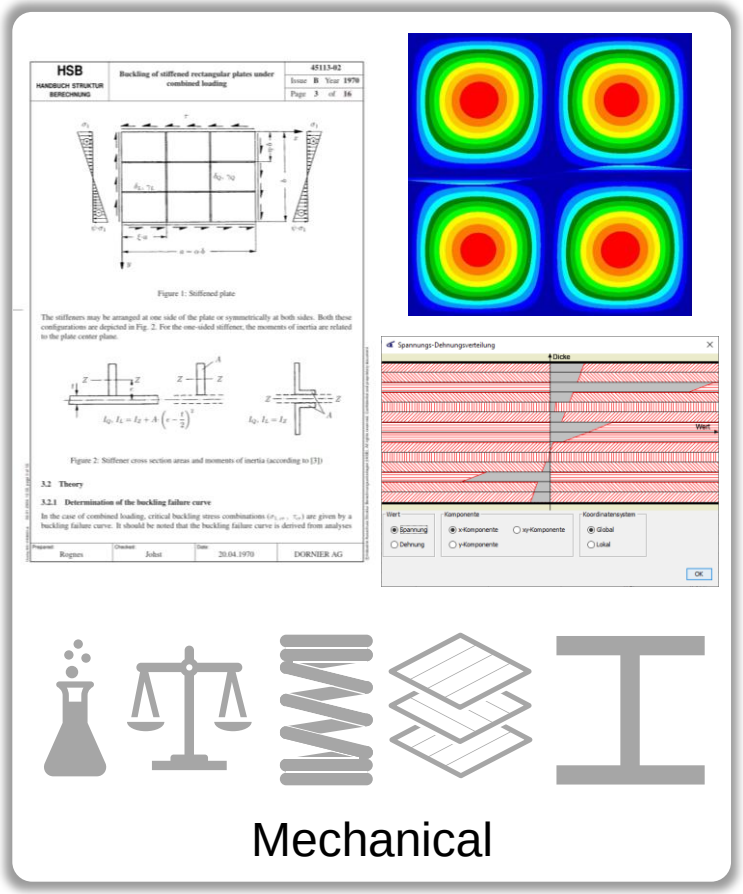
APPROACH

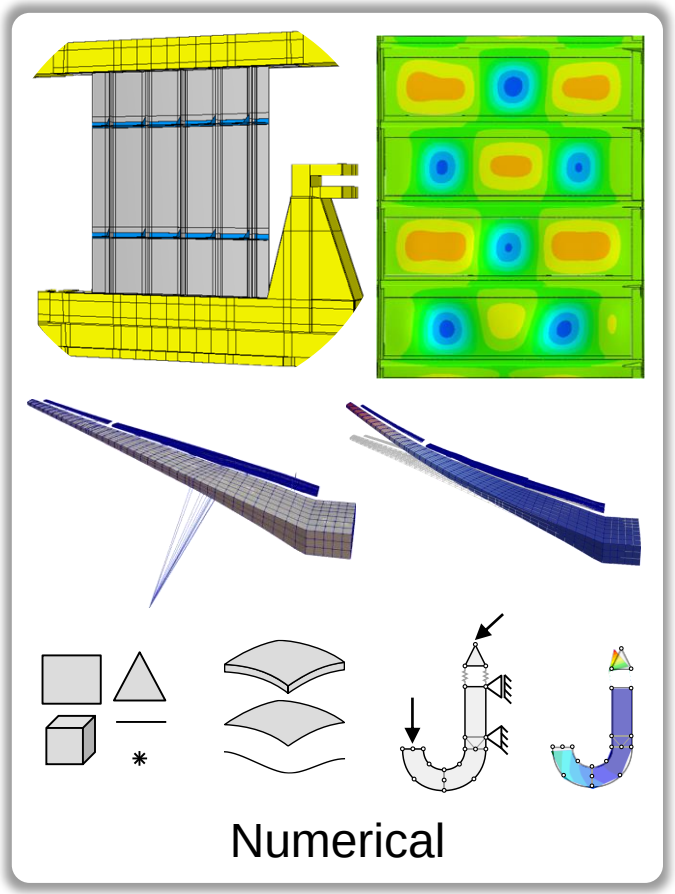
Approach
Hypothesis

- Subdivision of information that makes up a model possible → based on level of assumptions & theories → object oriented approach



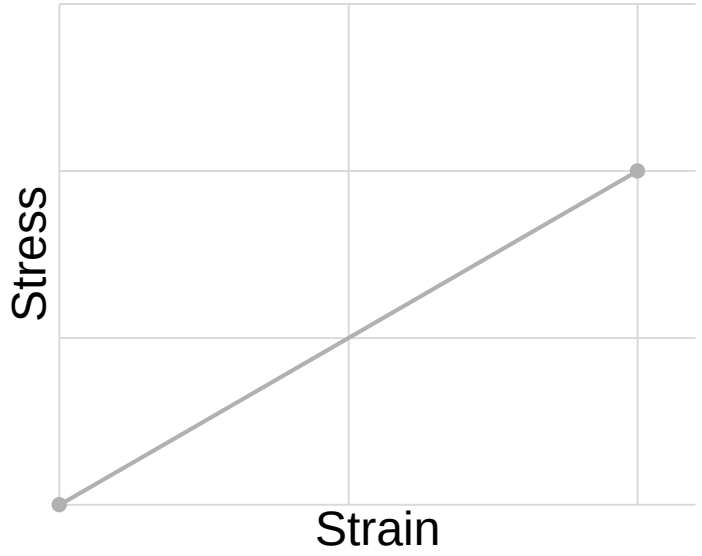




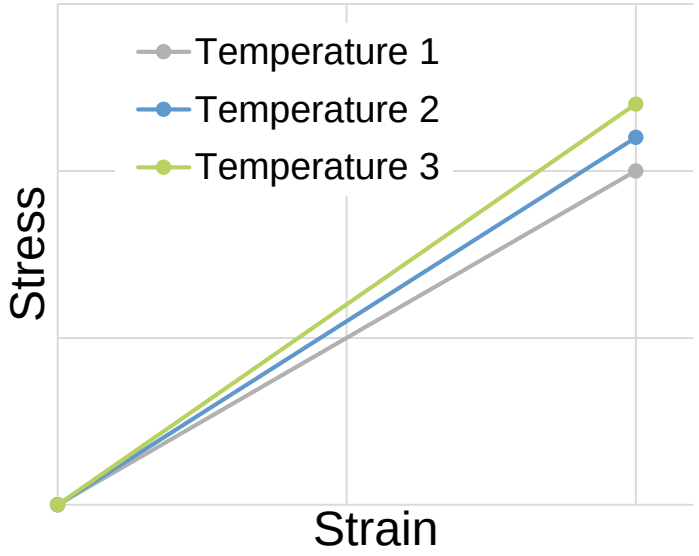


Approach
Base entity

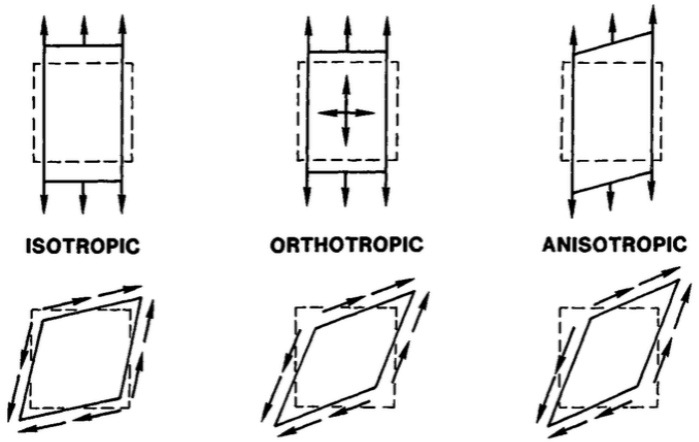
- Description:
  - Entity
  - Model
  - Value
- Example
  - ElasticityEvolution
  - Subtypes not complete
- Schema also applicable to nonlinear or non-cauchy value types



→ ConstantModel

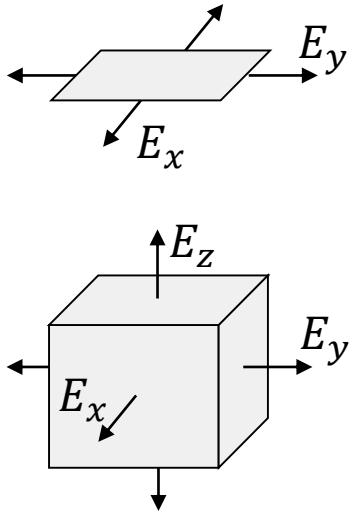


→ ConditionDependentModel



DOI: 10.13140/RG.2.2.31723.59682

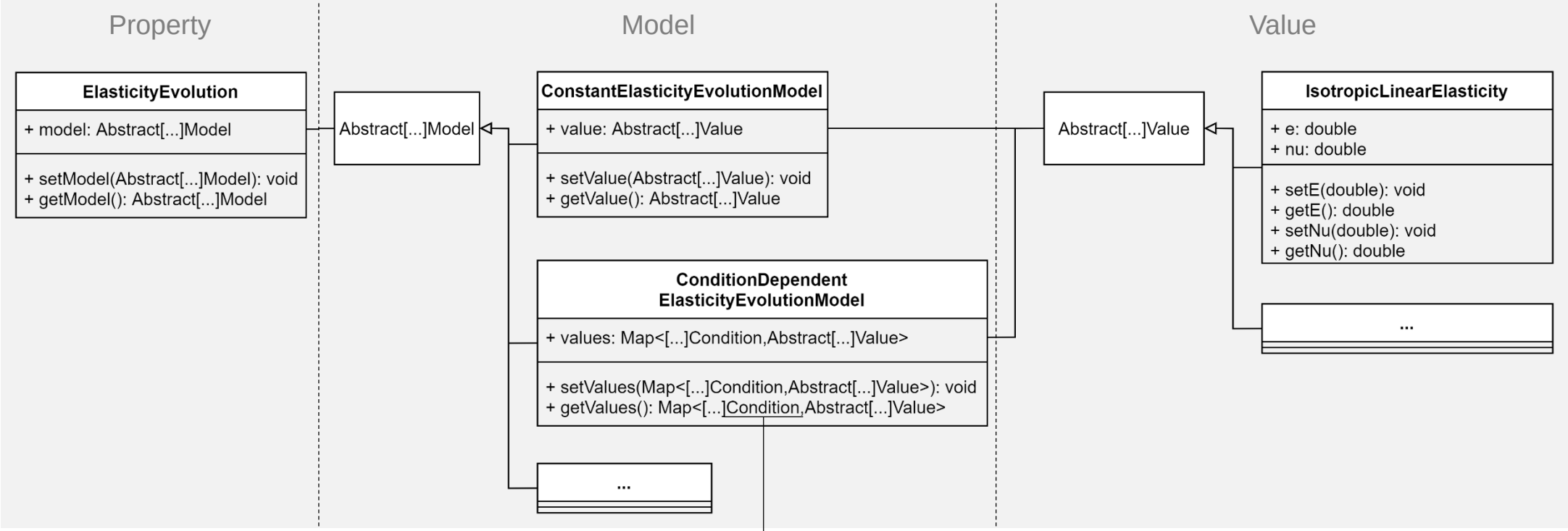
Behaviours



Idealizations

Approach
Base entity

- Property-Model-Value example



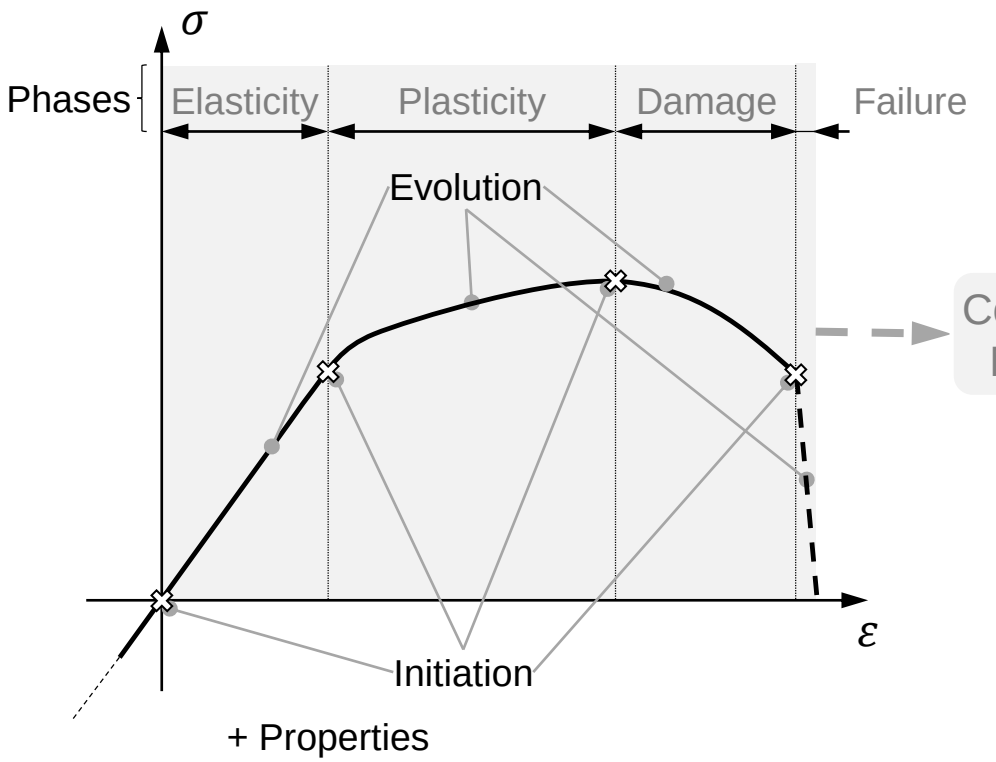
$$= f(T, p, \dot{\varepsilon}, \cdots)$$

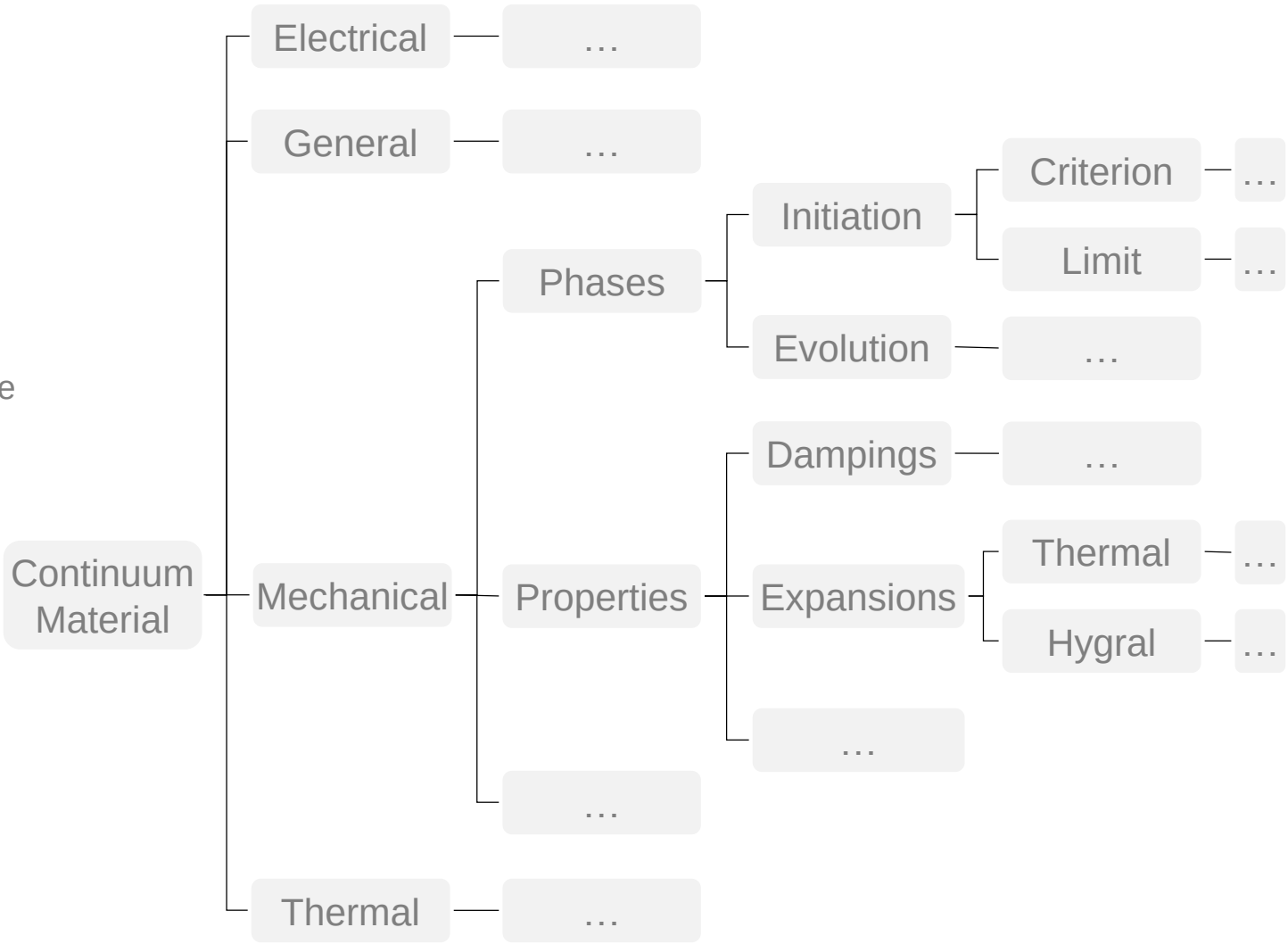
Approach
Compound entity

- Collect base entities
- Do not have models, just values, because models are contained in attributes
- Example: Material
  - Current value implementations:
    - CohesiveMaterial
    - ContinuumMaterial
    - DatabaseMaterial

Approach
Hierarchy development

- Search of model components
- Goal: expandable, modular
- Example: ContinuumMaterial



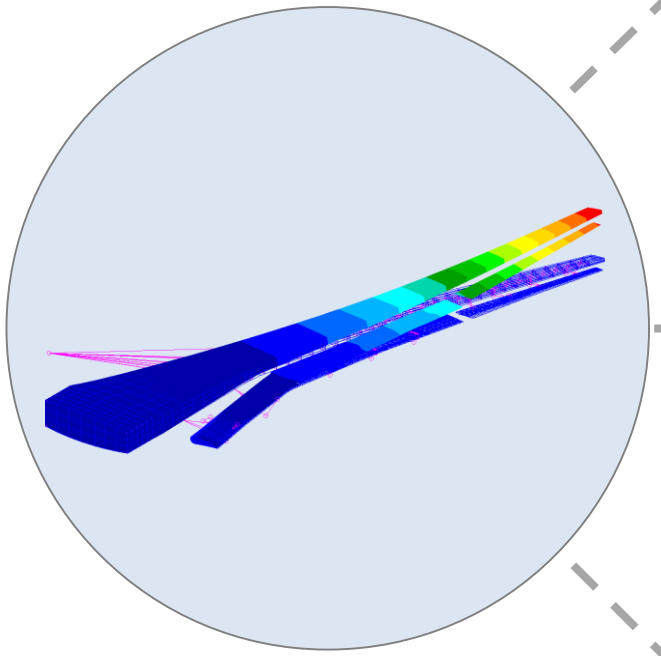


IMPLEMENTATION

```java
@EqualsAndHashCode(callSuper=true)

// JAXB
@XmlRootElement
@XmlSeeAlso({
    ConditionDependentElasticityEvolutionModel.class
   ,ConstantElasticityEvolutionModel.class
})

// Jackson
@JsonSubTypes({
    @JsonSubTypes.Type(value = ConditionDependentElasticityEvolutionModel.class)
   ,@JsonSubTypes.Type(value = ConstantElasticityEvolutionModel.class)
})

// -------------------------------
// Class
// -------------------------------

public class ElasticityEvolution<
            M extends AbstractElasticityEvolutionModel
        >
        extends
            AbstractMechanicalPhaseEvolution<
                ElasticityEvolution<M>
               ,M
            >
{

    /**
     * Create a new empty instance
     */
    public ElasticityEvolution(){}

    /**
     * Create a new instance
     * @param model {@link M}
     */
    public ElasticityEvolution(
            M model
```

```
BUILD SUCCESS
------------------------------------------------------------------------
Total time:  05:34 min
Finished at: 2022-09-15T18:23:18+02:00
------------------------------------------------------------------------
```

Implementation
Data types



Numerical implementation data

- Nodes
- Elements
- …

- Results

Connecting data

- Sections
- Loads & BC
- …

Math, physics & mech. idealization data

- Materials
- Composites
- Crosssections
- …

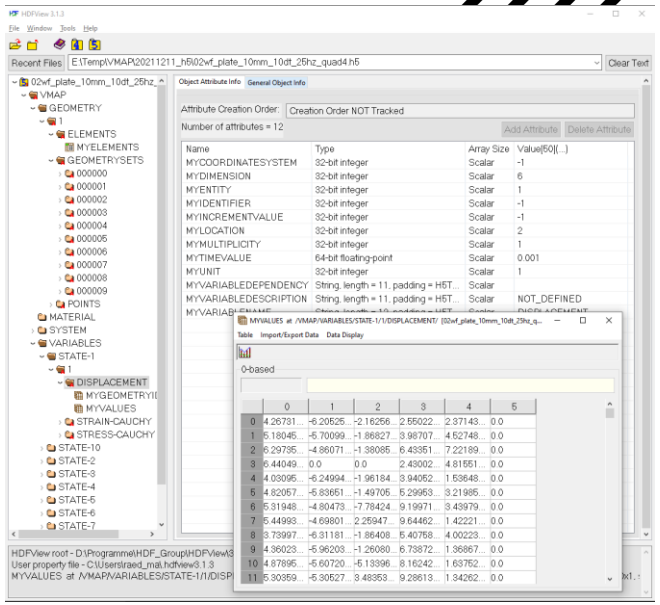
Heavy data

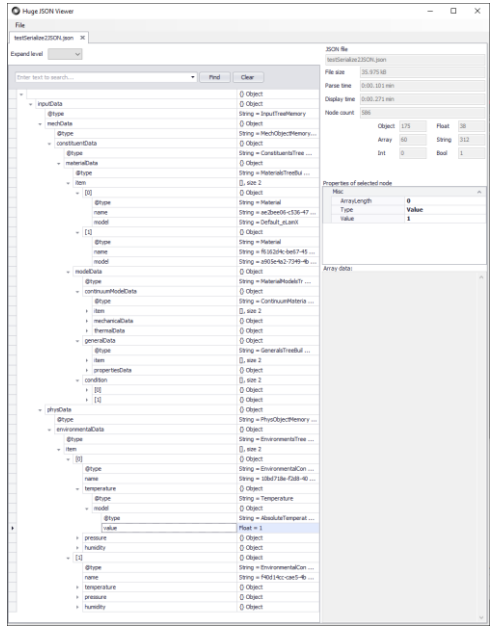
VMAP

References

Light data

jMeS





Implementation
Goals

- Solver-agnostic CSM model & result data format
- Differentiation between
  - Numerical discretization-dependent information → „Heavy“ data
  - Model information → „Light“ data
- Heavy data:
  - Usage existing format
  - Preferrably based on well-known format, supported by libraries
  - Visualization
- Light data:
  - Programming language independent
  - Machine readable
  - Human readable
  - Single-source of truth

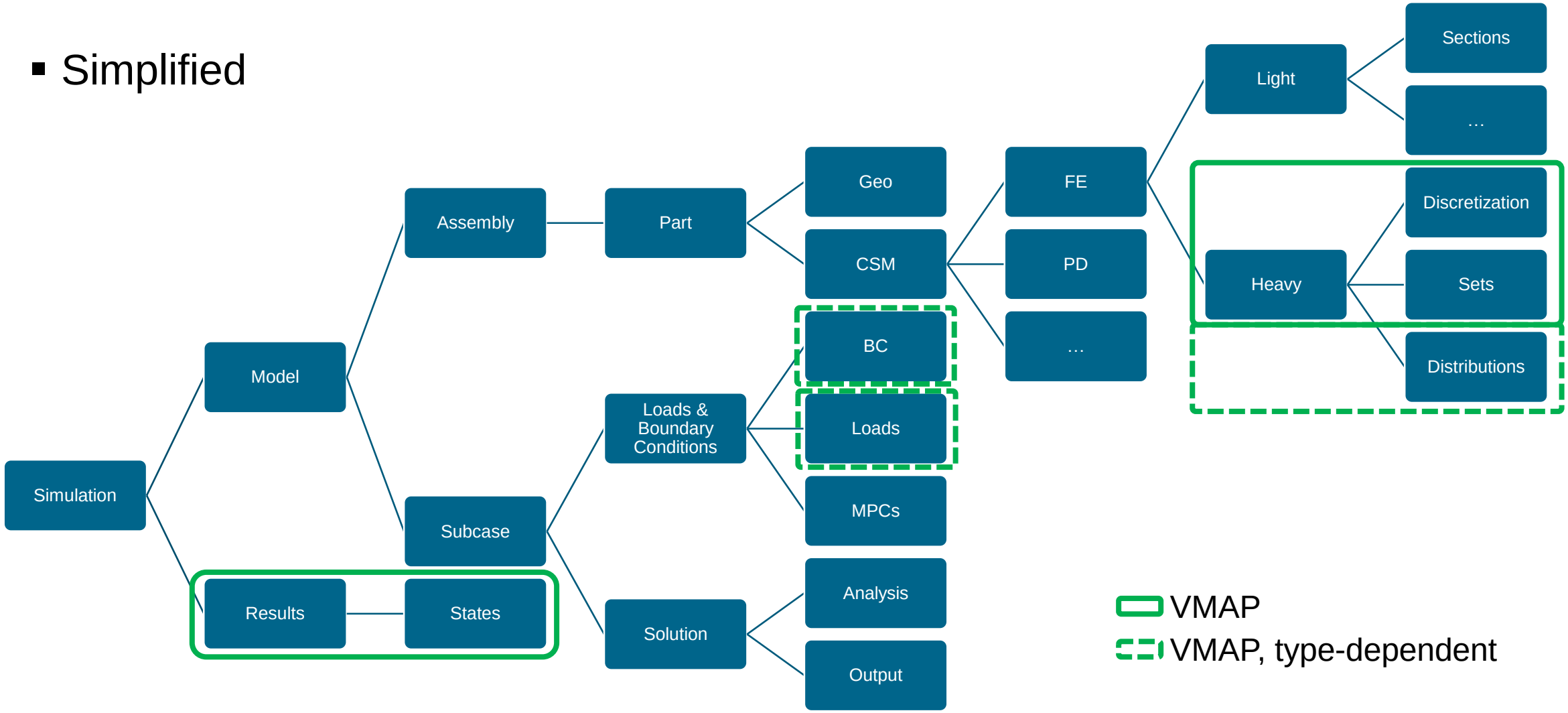
Serialization format → JSON, XML, YAML, …
→ REST ready

- FEM:
  - Exodus
  - MOAB
  - Silo
  - VMAP
  - XDMF
- Particles:
  - H5Part
- CFD:
  - CGNS

Alternatives

Implementation
Hierarchy

- Simplified



Implementation
Approach

- Hierarchical, object-oriented data structure
  - Implementation in Java-based backend jMeS (Java Mechanics Suite)
  - Usage of JAXB annotations across hierarchy to denote
    - Abstraction
    - Attribute types
- Annotations understood by de-/serialization libraries
  - E.g. Jackson, MOXy, …
  - Allows creation of schema, e.g. XML
  - Creation of XML, JSON, YAML, …

Abstract superclass

```
// JAXB
@XmlTransient
@XmlAccessorType(XmlAccessType.FIELD)

// Jackson
@JsonTypeInfo(use = JsonTypeInfo.Id.NAME, include = JsonTypeInfo.As.PROPERTY)
public abstract class AbstractMechObjectProperty<
            T extends AbstractMechObjectProperty<T,M>
           ,M extends AbstractMechObjectModel
        >
```

Implementation

```
// JAXB
@XmlRootElement
@XmlSeeAlso({
    ConditionDependentElasticityEvolutionModel.class
   ,ConstantElasticityEvolutionModel.class
})

// Jackson
@JsonSubTypes({
    @JsonSubTypes.Type(value = ConditionDependentElasticityEvolutionModel.class)
   ,@JsonSubTypes.Type(value = ConstantElasticityEvolutionModel.class)
})
public class ElasticityEvolution<
```

Attributes

```
public abstract class AbstractMechObjectProperty<
            T extends AbstractMechObjectProperty<T,M>
           ,M extends AbstractMechObjectModel
        >
    @XmlAttribute
    @XmlID
    private String name = DEFAULT_NAME;

    @XmlIDREF
    protected M model;

    protected MetaInfo<? extends AbstractMetaInfoValue> metaInfo;
```

Expand level 10

Enter text to search... Find Clear

	{} Object
model	{} Object
@type	String = ModelTreeBuilder ...
name	String = e15288f8-ba12-4c ...
assemblyData	{} Object
@type	String = TreeBuilderAssem ...
name	String = 17d2bef7-64a6-4f ...
part	[], size 1
[0]	{} Object
@type	String = TreeBuilderPart
name	String = d6250751-34db-4f ...
geometry	{} Object
@type	String = GeoMemoryTreeBui ...
r0Data	{} Object
@type	String = R0GeometriesTree ...
item	[], size 8
[0]	{} Object
@type	String = PointR3
name	String = f2d28093-e2a2-49 ...
topology	{} Object
@type	String = PointR3Topology
coordinate1	Float = 0
coordinate2	Float = 0
coordinate3	Float = 0
preferences	String = 5eb070f7-b2d8-4f ...
[1]	{} Object
@type	String = PointR3
name	String = fda44dc9-b546-46 ...
topology	{} Object
@type	String = PointR3Topology
coordinate1	Float = 1
coordinate2	Float = 0
coordinate3	Float = 0
preferences	String = 5eb070f7-b2d8-4f ...
[2]	{} Object
@type	String = PointR3
name	String = 3eb63f8f-0601-4d ...
topology	{} Object
@type	String = PointR3Topology
coordinate1	Float = 1
coordinate2	Float = 1
coordinate3	Float = 0
preferences	String = 5eb070f7-b2d8-4f ...
[3]	{} Object
@type	String = PointR3
name	String = add07b42-fe7f-49 ...
topology	{} Object
@type	String = PointR3Topology
coordinate1	Float = 0
coordinate2	Float = 1
coordinate3	Float = 0

JSON file

testSerialize2JSON.json

File size	83.737 kB
Parse time	0:00.121 min
Display time	0:00.286 min
Node count	1510

Object	381	Flo	5
Array	130	String	751
Int	153	Bool	37

Properties of selected node

Misc	
ArrayLength	**0**
Type	**Object**
Value	

Array data:

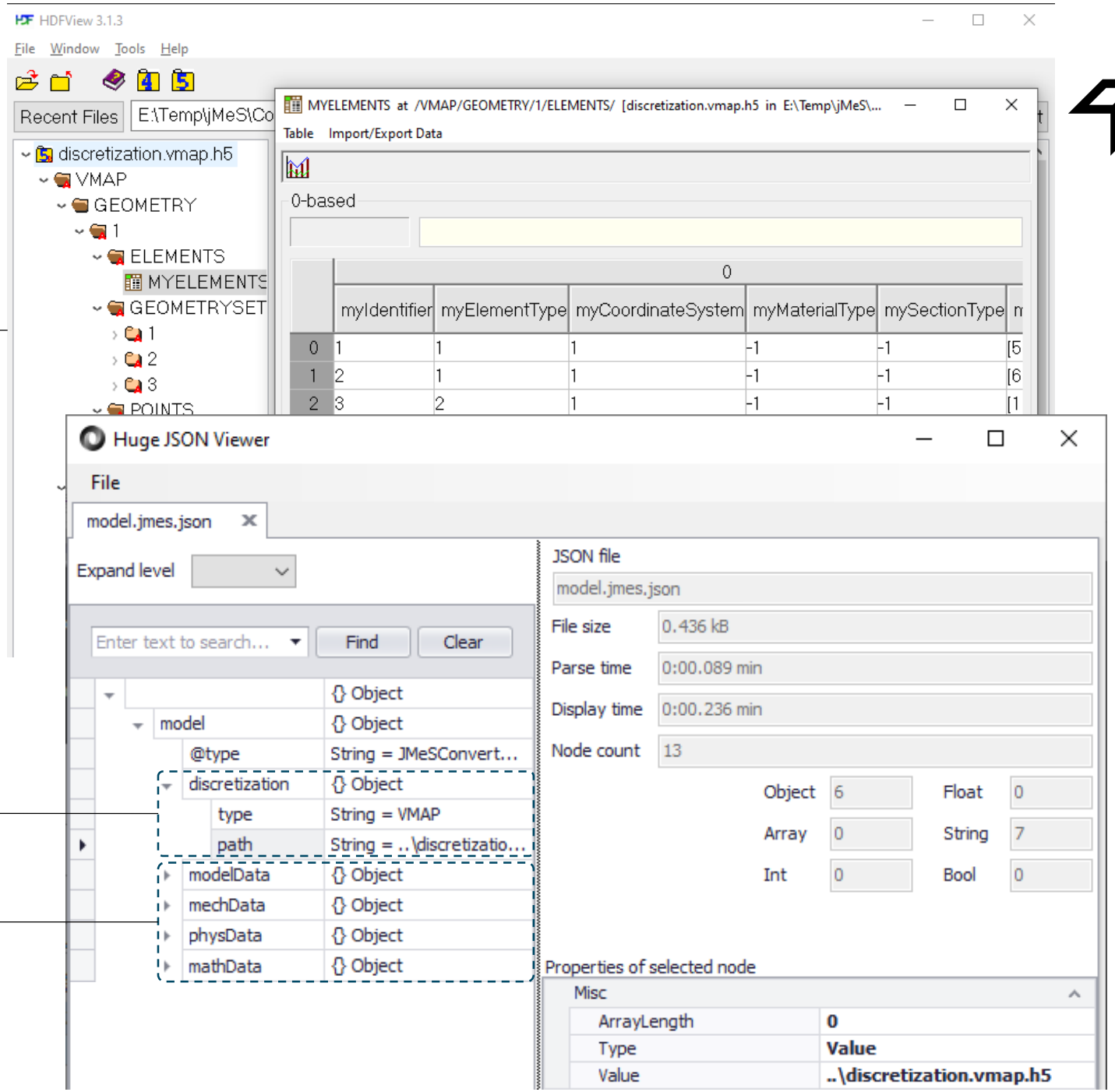
EXAMPLES

Examples

- Implementation

Link to heavy data file

Light data



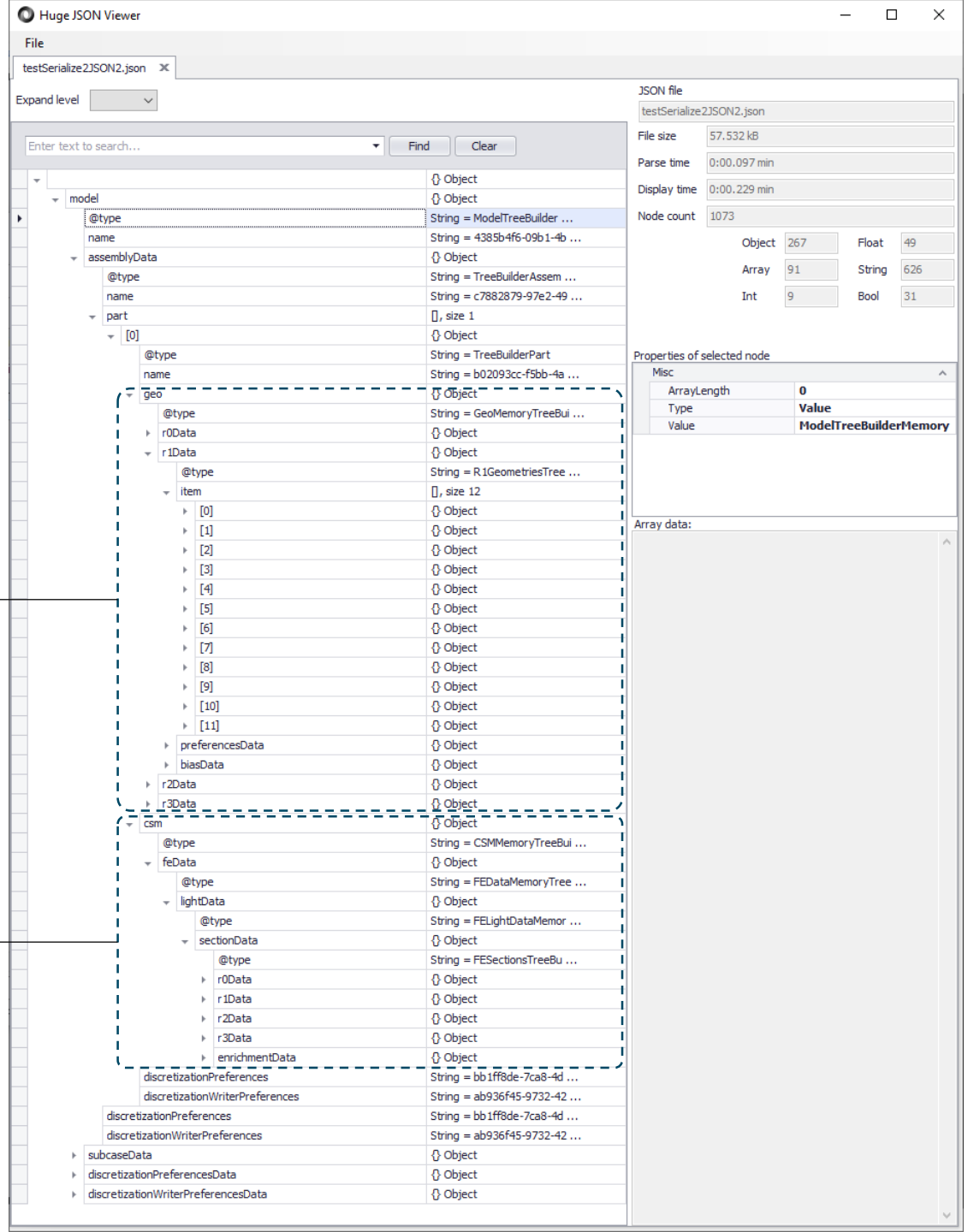
- Demonstration

Examples

- Implementation

Geometry

CSM



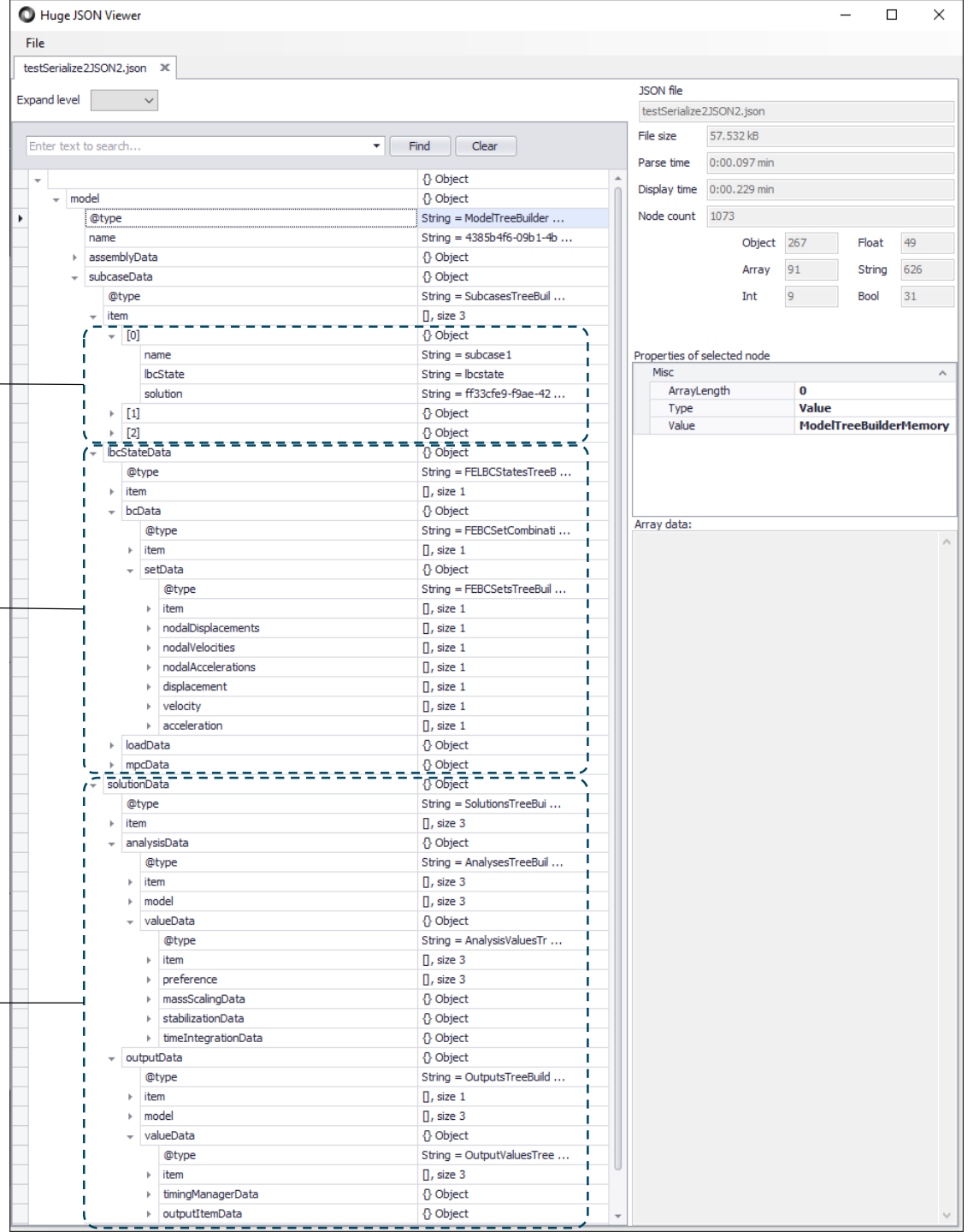
Examples

- Implementation

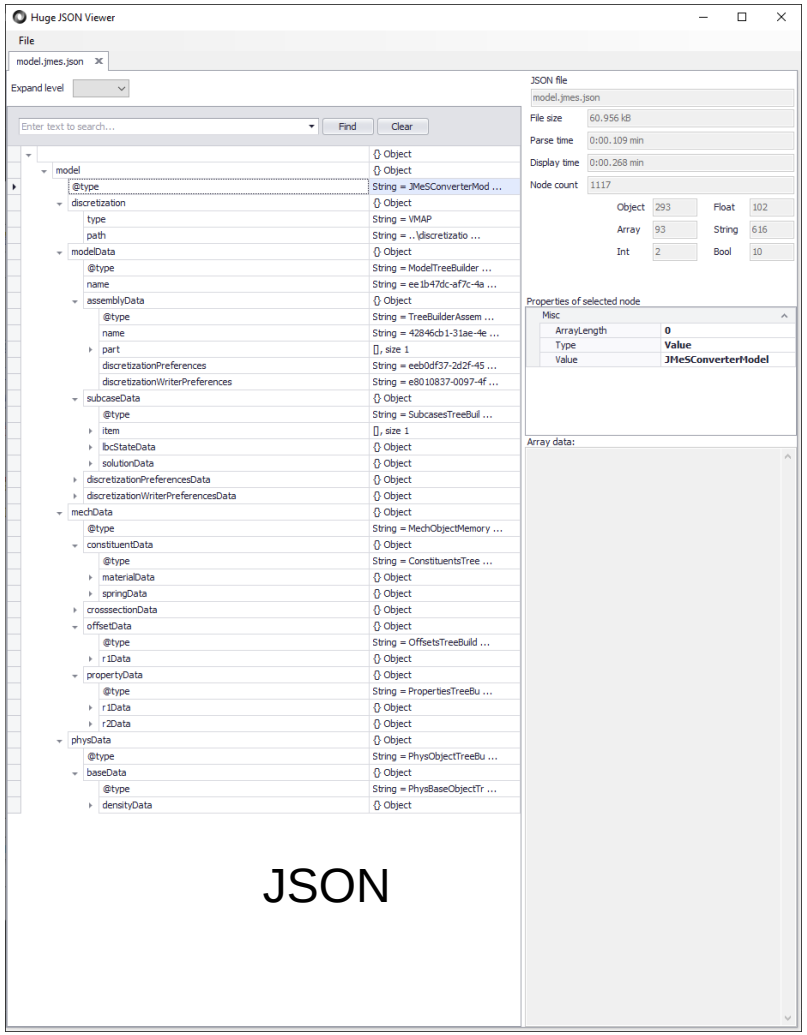
Subcases

Load & boundary condition (LBC) state

Solution

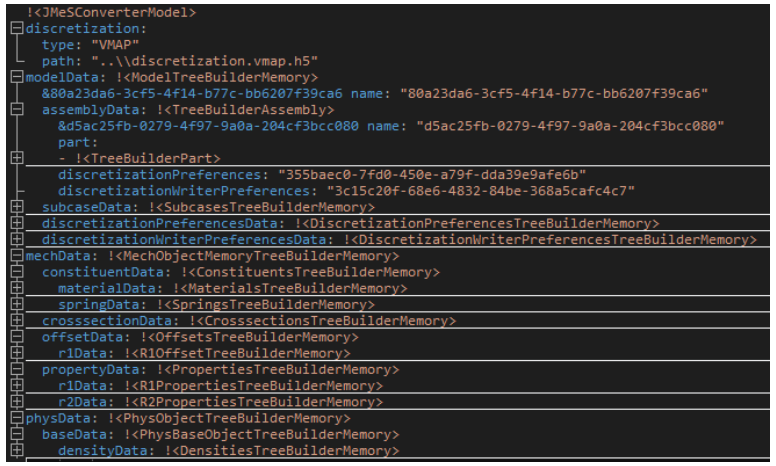


Examples
Machine readable formats



JSON

XML

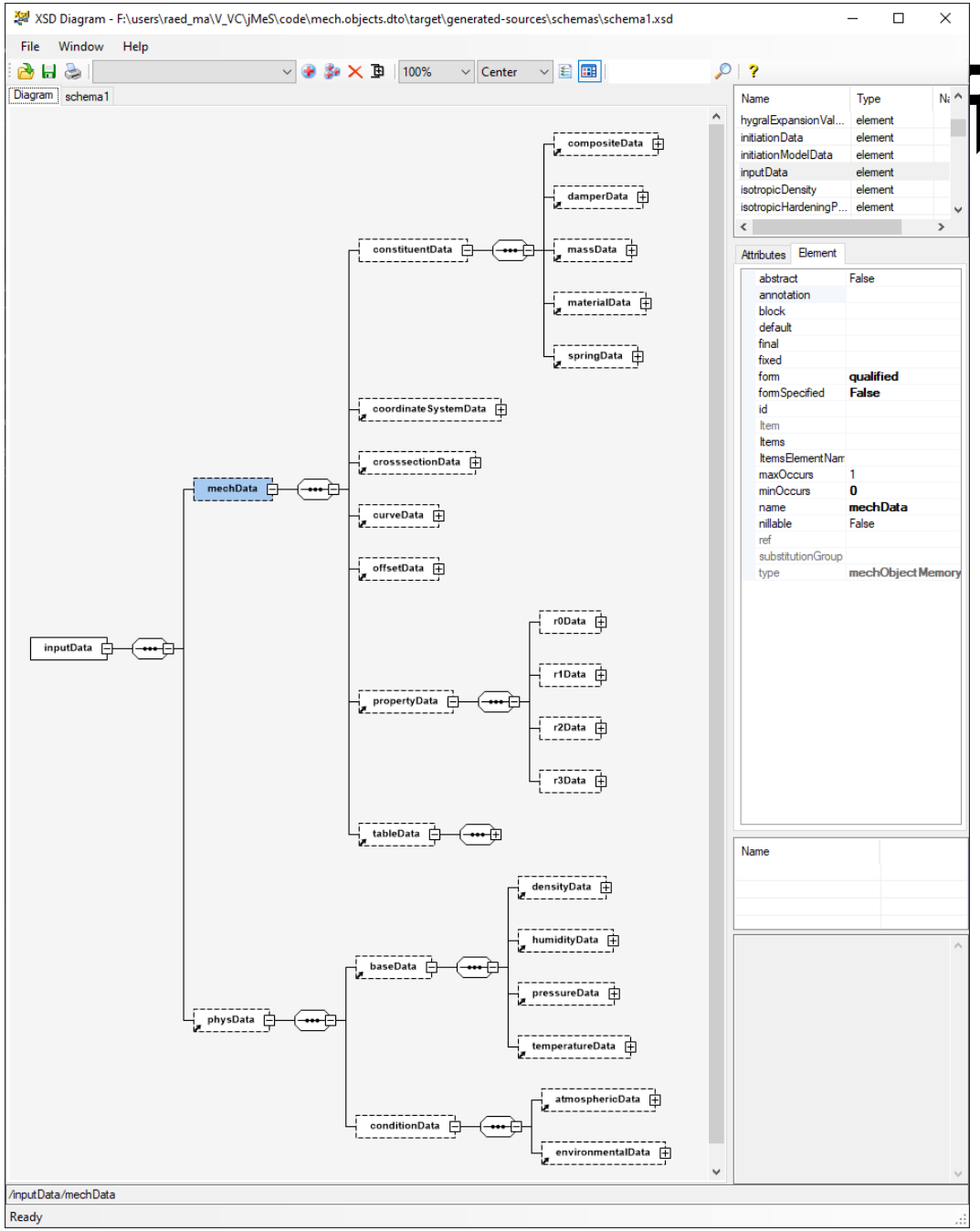


YAML

- Fully automatic generation by selection of serialization goal

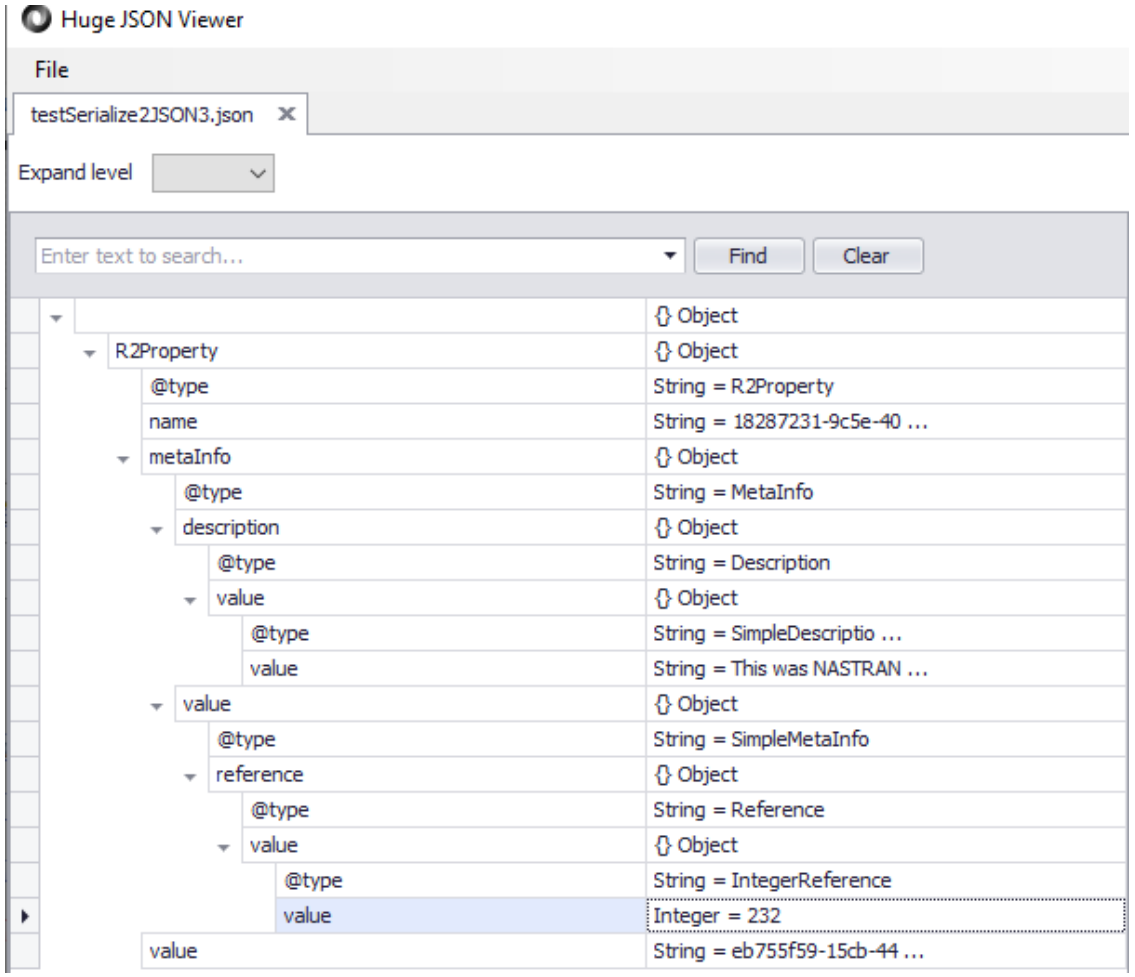
Examples
Machine readable formats

- Automatic XSD schema generation during code compilation
- Automatic translation to code:
  - C++: codesynthesis
  - Python: generateDS
  - Java: Jackson
  - …



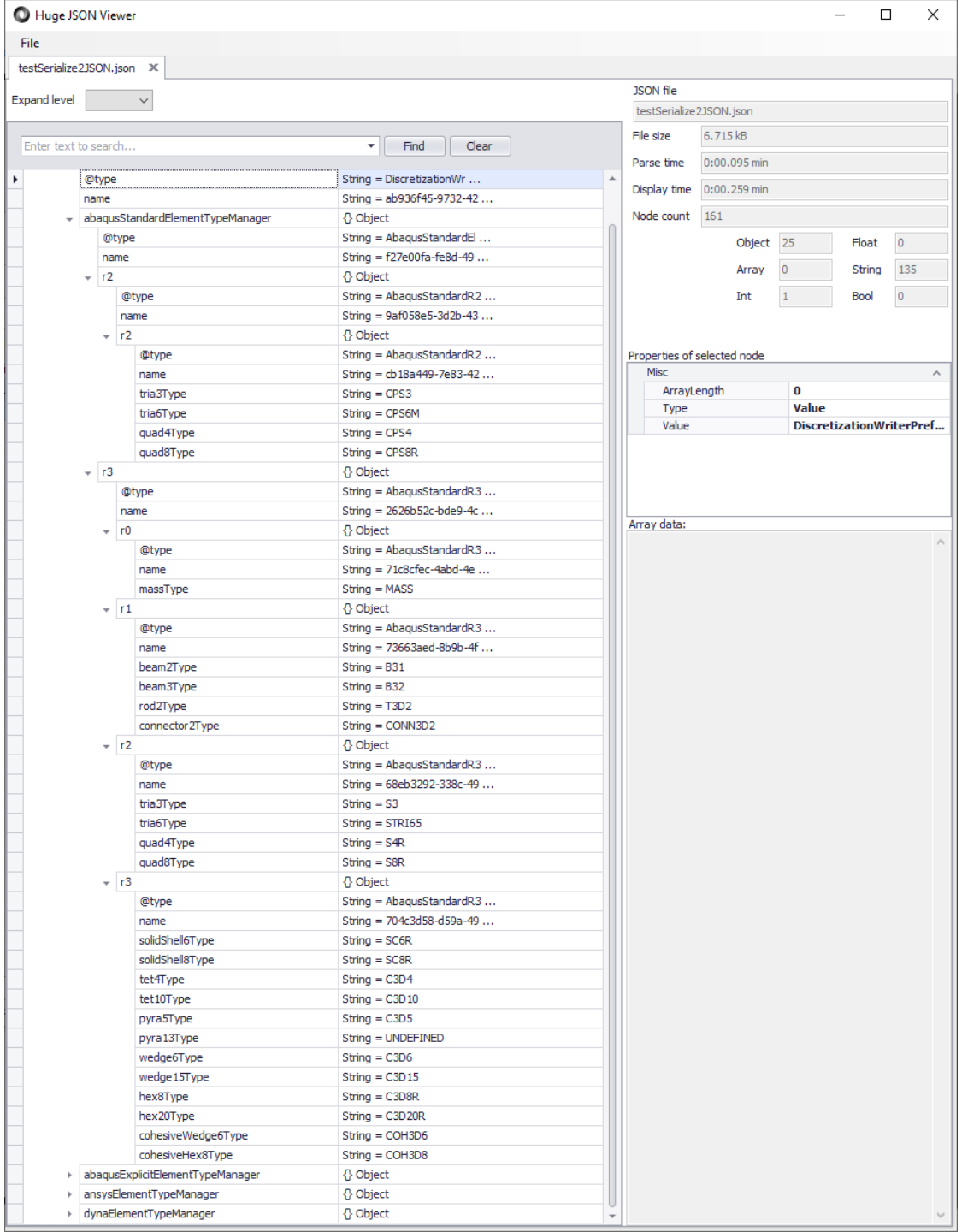
Examples
Meta information

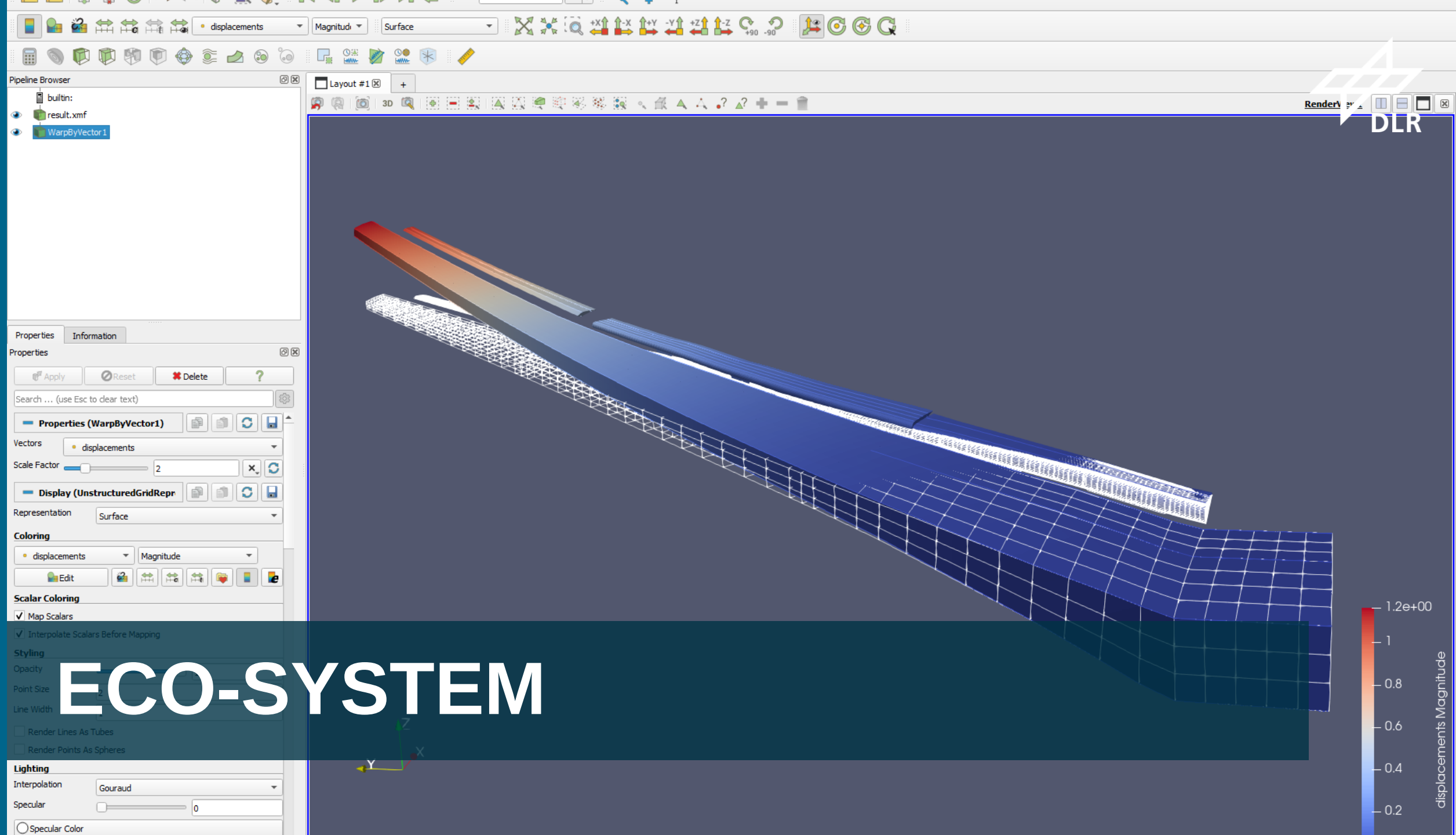
- E.g. reference to ID‘s from number-based formats (e.g. Nastran, LS-Dyna) for internal String UUID approach
- Currently:
  - Description
  - Reference
  - Source
- Modularly expandable
- Applicable to most DTO‘s



Examples
Solver-dependent information

- Will never achieve complete solver-independence
- Solver-independent information addable
- Example: ElementTypeManagers
- Fully expandable w.r.t.
  - Solution method preferences
  - …

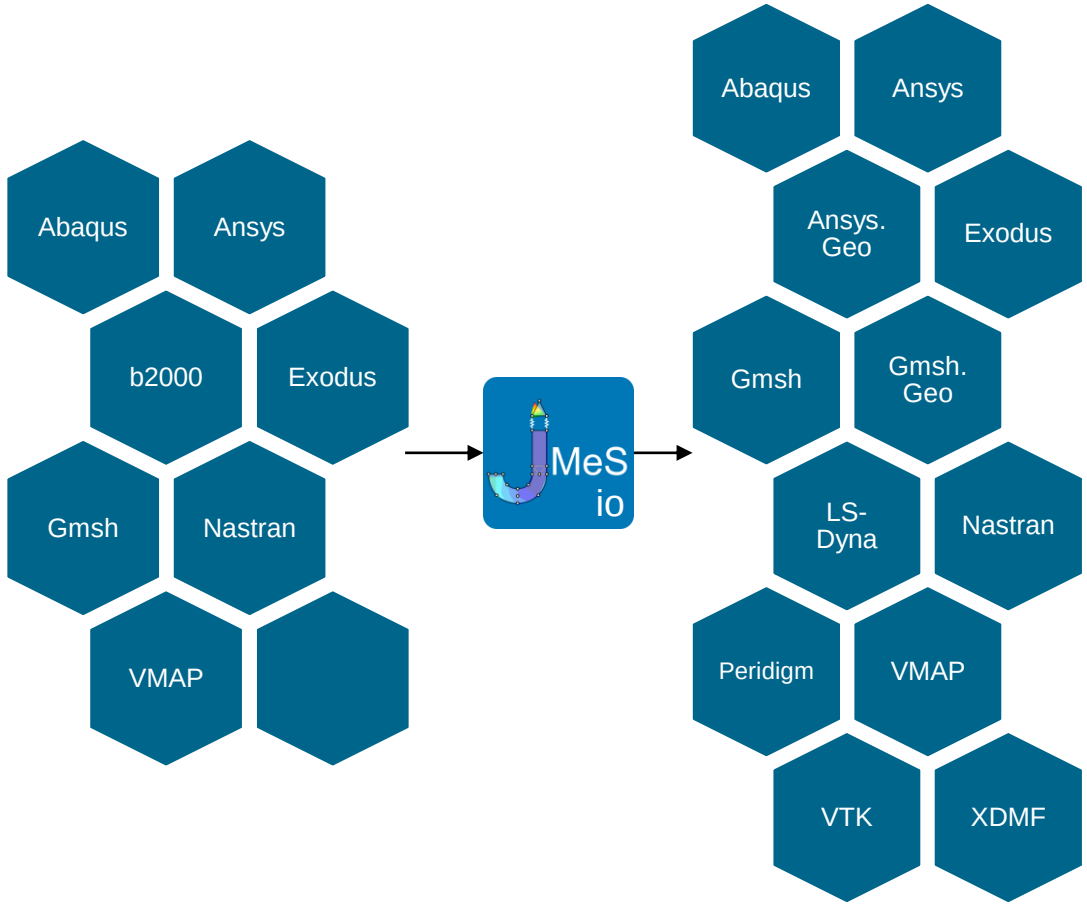



displacements
Magnitude
Surface
Pipeline Browser
builtin:
result.xmf
WarpByVector1
Layout #1
Properties
Information
Apply
Reset
Delete
Search ... (use Esc to clear text)
Properties (WarpByVector1)
Vectors
displacements
Scale Factor
2
Display (UnstructuredGridRepr
Representation
Surface
Coloring
displacements
Magnitude
Edit
Scalar Coloring
Map Scalars
Interpolate Scalars Before Mapping
Styling
Opacity
Point Size
Line Width
Render Lines As Tubes
Render Points As Spheres
Lighting
Interpolation
Gouraud
Specular
0
Specular Color
1.2e+00
1
0.8
0.6
0.4
0.2
displacements Magnitude

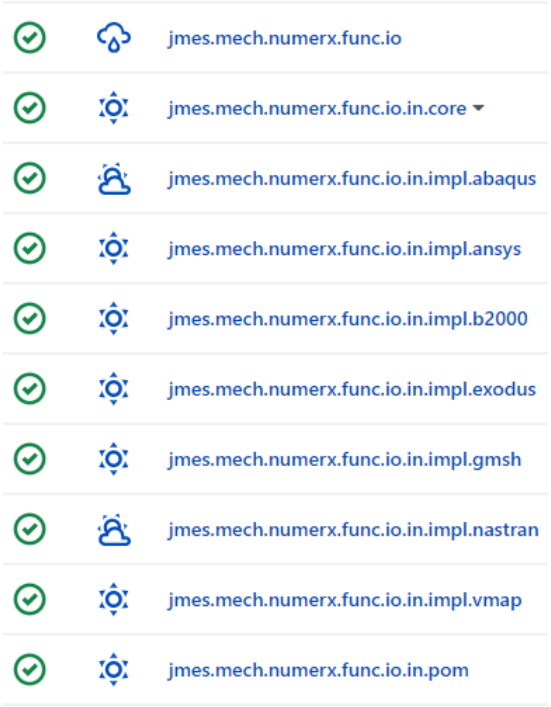
ECO-SYSTEM

Eco-system
IO plugins

- Conversion from & to solver-specific formats using io-plugins

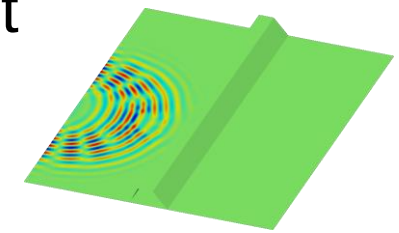


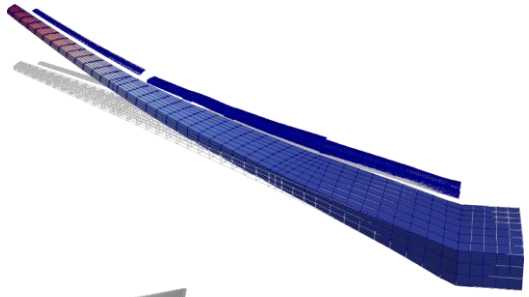
- Usage from python with Py4J interface
- Software engineering:
  - Testing
  - Documentation
  - CI/CD

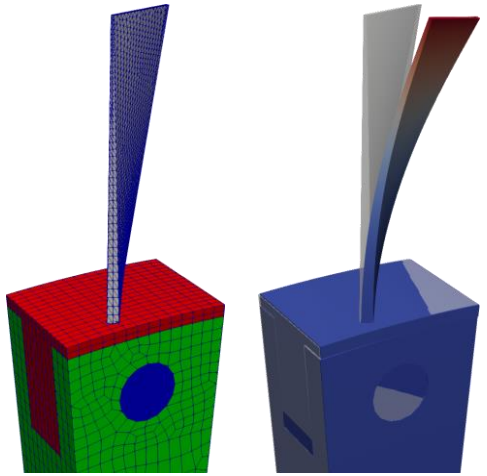


Eco-system
Visualization

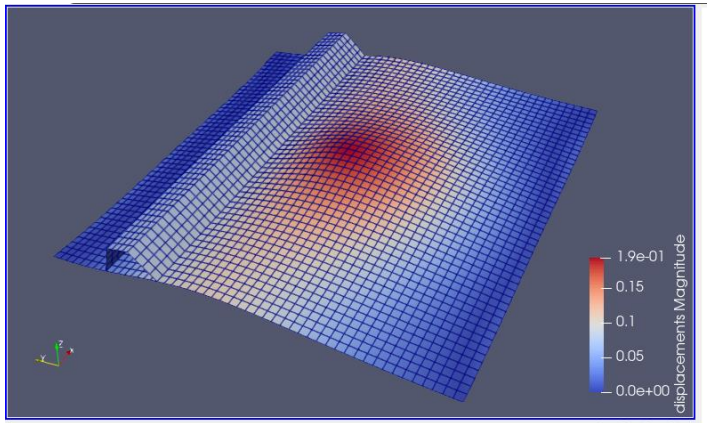
- XDMF export
- Useable by
  - ParaView
  - VisIt







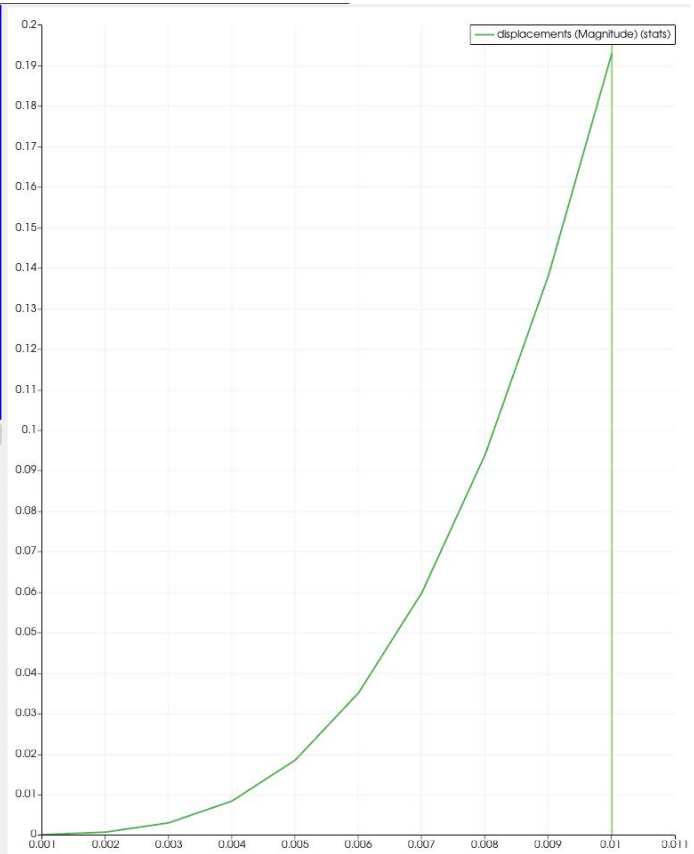
- Automated postprocessing enabled, e.g. by pvpython



SpreadSheetView2

Showing PlotSelectionOverTime1 Attribute: Row Data Precision: 6

	Block Name	Row ID	N	Time	avg(X)	avg(Y)	avg(Z)	avg(displacements (0))	avg(displacements (1))	avg(displacements (2))	avg(d
0	stats	0	1	0.001	250	240	0.0185639	8.32518e-08	-1.42928e-06	7.42557e-05	7.4269
1	stats	1	1	0.002	250	239.997	0.177728	5.07389e-07	-1.09788e-05	0.000710913	0.0007
2	stats	2	1	0.003	250	239.992	0.749688	-1.0978e-08	-3.38598e-05	0.00299875	0.0029
3	stats	3	1	0.004	249.998	239.983	2.09672	-8.9602e-06	-7.05869e-05	0.00838689	0.0083
4	stats	4	1	0.005	249.989	239.969	4.62704	-4.39711e-05	-0.00012289	0.0185082	0.0185
5	stats	5	1	0.006	249.969	239.954	8.77877	-0.000125061	-0.000186723	0.0351151	0.0351
6	stats	6	1	0.007	249.937	239.943	14.9661	-0.000254121	-0.000227641	0.0598646	0.0598
7	stats	7	1	0.008	249.899	239.957	23.4999	-0.000405452	-0.000171531	0.0939994	0.0940
8	stats	8	1	0.009	249.865	240.017	34.5641	-0.000539408	6.5739e-05	0.138256	0.1382
9	stats	9	1	0.01	249.847	240.138	48.2696	-0.000611975	0.000551454	0.193079	0.1930



Expand level 10

Enter text to search... Find Clear

	{} Object
model	{} Object
@type	String = ModelTreeBuilder ...
name	String = e15288f8-ba12-4c ...
assemblyData	{} Object
@type	String = TreeBuilderAssem ...
name	String = 17d2bef7-64a6-4f ...
part	[], size 1
[0]	{} Object
@type	String = TreeBuilderPart
name	String = d6250751-34db-4f ...
geometry	{} Object
@type	String = GeoMemoryTreeBui ...
r0Data	{} Object
@type	String = R0GeometriesTree ...
item	[], size 8
[0]	{} Object
@type	String = PointR3
name	String = f2d28093-e2a2-49 ...
topology	{} Object
@type	String = PointR3Topology
coordinate1	Float = 0
coordinate2	Float = 0
coordinate3	Float = 0
preferences	String = 5eb070f7-b2d8-4f ...
[1]	{} Object
@type	String = PointR3
name	String = fda44dc9-b546-46 ...
topology	{} Object
@type	String = PointR3Topology
coordinate1	Float = 1
coordinate2	Float = 0
coordinate3	Float = 0
preferences	String = 5eb070f7-b2d8-4f ...
[2]	{} Object
@type	String = PointR3
name	String = 3eb63f8f-0601-4d ...
topology	{} Object
@type	String = PointR3Topology
coordinate1	Float = 1
coordinate2	Float = 1
coordinate3	Float = 0
preferences	String = 5eb070f7-b2d8-4f ...
[3]	{} Object
@type	String = PointR3
name	String = add07b42-fe7f-49 ...
topology	{} Object
@type	String = PointR3Topology
coordinate1	Float = 0
coordinate2	Float = 1
coordinate3	Float = 0

JSON file

	testSerialize2JSON.json
File size	83.737 kB
Parse time	0:00.121 min
Display time	0:00.286 min
Node count	1510

Object	381	Flo	5
Array	130	String	751
Int	153	Bool	37

Properties of selected node

Misc	
ArrayLength	**0**
Type	**Object**
Value	

Array data:

CONCLUSION

Conclusion
Current

- Hierarchical information storage based on level of assumptions possible
  - Grind: E.g. mech.object.dto project → 2891 classes, interfaces & enums
- Used as input/output for several DLR tools
  - Analytical methods
  - Model generators
  - FEM data handling
- Use of common solver-agnostic internal data format with solver-dependent extensions to convert from & to solver-specific formats possible
  - Proof of concept in io plugins
- State:
  - By no means feature-complete
  - Hierarchy converging, yet not free of refactoring

Conclusion
Next steps

- On our way to open source
- Goal:
  - Hierarchy development & standardization
  - jMeS backend directly linked to hierarchy creation, but is just one reference implementation
  - Further hierarchy development possible independent of backend, e.g. similar to CPACS
- Looking forward to feedback

THANK YOU FOR YOUR ATTENTION

Contact

Martin Rädel

German Aerospace Center
Institute of Composite Structures and Adaptive Systems
Department of Structural Mechanics

Lilienthalplatz 7
38108 Braunschweig

☎ +49 (0)531 295-2048
🖷 +49 (0)531 295-2232
✉ martin.raedel@dlr.de
💻 www.dlr.de/fa

Virtual Product House

Cornelius-Edzard-Straße 15
28199 Bremen

💻 www.dlr.de/vph

Impressum

Thema:	Complete Model Storage – A solver-agnostic hierarchical approach
Datum:	19.09.2022
Autor:	Martin Rädel
Institut:	DLR-FA-STM & DLR VPH
Bildcredits:	-