LESSONS LEARNED FROM DEVELOPING A DIGITAL PROTOTYPE WITHIN THE ARENA2036 ENVIRONMENT AND IMPROVEMENTS WITH THE NEW VMAP STANDARD

Christian Liebold, Dr.-Ing. André Haufe
(DYNAmore GmbH, Germany)

Mathieu Vinot, Martin Holzapfel
(DLR Stuttgart – German Aerospace Center, Germany)

Jörg Dittmann, Patrick Böhler
(IFB – University of Stuttgart, Institute of Aircraft Design, Germany)

Florian Fritz, Hermann Finckh
(DITF – Deutsche Institute für Textil und Faserforschung, Germany)

Abstract

In 2013, a new research campus called *ARENA2036*, which is the abbreviation for *Active Research Environment for the Next Generation of Automobiles*, has been established at the University of Stuttgart, Germany. The year 2036 accounts for the 150th anniversary of the patented version of Carl Benz's first three-wheeled vehicle with a combustion engine which led to the first serial production of motorized vehicles as we know them today. The main focus of the research campus is to investigate the rising importance of *Industry 4.0* applications and in a first phase, three leading technical projects worked on topics such as lightweight design and system/sensor integration (LeiFu), new and adaptive production lines in industrial environments (ForschFab), and enhancing simulation capabilities with the definition of a *Digital Prototype* (DigitPro) [Dittmann et. al. 2015]. The latter aimed to improve the daily life of simulation engineers by developing a digital prototype which defines a simulation data exchange platform based on a HDF5 data storage container and by improving mapping schemes to structural analysis meshes for selected process simulations such as braiding, resin infiltration, open-reed weaving (ORW), and draping of continuous fiber reinforced plastic (CFRP) materials. This should help to close the simulation process chain within the project and is now used as a basis for a subsequent 5-years governmental funded project called *Digital Fingerprint* [Böhler et. al. 2018].
To develop accurate and efficient mapping and data exchange interfaces, physical relevant parameters for the different processes have to be identified in a first step. In a second step, a proper data exchange format has to be defined,

since the different partners use various software solutions which output results in different formats and therefore cannot be interpreted by other software tools.

In this work, it will be shown why the HDF5® format seems to be a valid storage platform for a large amount of data which accumulates when following the full simulation process chain for all different kinds of materials, components, and processes. Furthermore, a first step has been made to define a common standard for simulation result data exchange and it will be discussed, why the ITEA-VMAP project [Wolf et. al 2018] seems to consist of the better consortium to target topics such as data semantics for resulting analysis data and their interpretation depending on the physical meaning in material modeling and what steps have already been taken to define a common data exchange format. In addition it will be discussed, how the resulting VMAP standard might influence the subsequent project *Digital Fingerprint* which aims for the inclusion of all relevant data being collected along a components life-time from first concepts and computer aided design via the engineering process and its virtual process simulation chain until its final usage in real applications and probable failure. Finally, an overview regarding the implementation status of the VMAP standard into the used mapping software envyo® shall be given, showing the capabilities to cover the use cases defined within the VMAP project. These examples will include a use case coupling thermal and structural analysis and a use case showing the simulation process chain for continuous fiber reinforced plastics. Furthermore, enhancements made for mapping processes within the ARENA2036 project will be discussed.

1. *Definition of the Digital Prototype within ARENA2036 - DigitPro*

The sub-project *Digital Prototype*, or short *DigitPro* of the ARENA2036 research environment consisted of one OEM, three research partners, namely the Institute of Aircraft Design (IFB), University of Stuttgart, GER, the Deutsche Institute für Textil und Faserforschung (DITF), Denkendorf, GER, the German Aerospace Center (DLR), Stuttgart, GER and one small- and medium sized enterprise (SME) – the DYNAmore GmbH, also based in Stuttgart, GER. Given the natural interest of each of the above mentioned institutions, they were looking onto the aspects of numerical simulation of continuous fiber reinforce plastics (CFRP) with different targets and eventually on different length scales. The manufacturing processes chosen within this research project were the braiding process, the resin transfer molding (RTM) process, the open-reed weaving (ORW) process, and a preliminary or subsequent draping process of a preform of any of the above mentioned preprocessing steps. Besides investigating and improving the simulation processes with different simulation tools, one of the targets was also to allow the consideration of simulation results gained along the simulation process chain in the following analysis step. For evaluation purposes, a small component was chosen for each of the starting processes (braiding and ORW) and by the end of the five years project, two main targets should be fulfilled:

the overall development time of a car floor should be reduced by 50 % and the
weight saving factor should be around 10 %. The latter also explains why
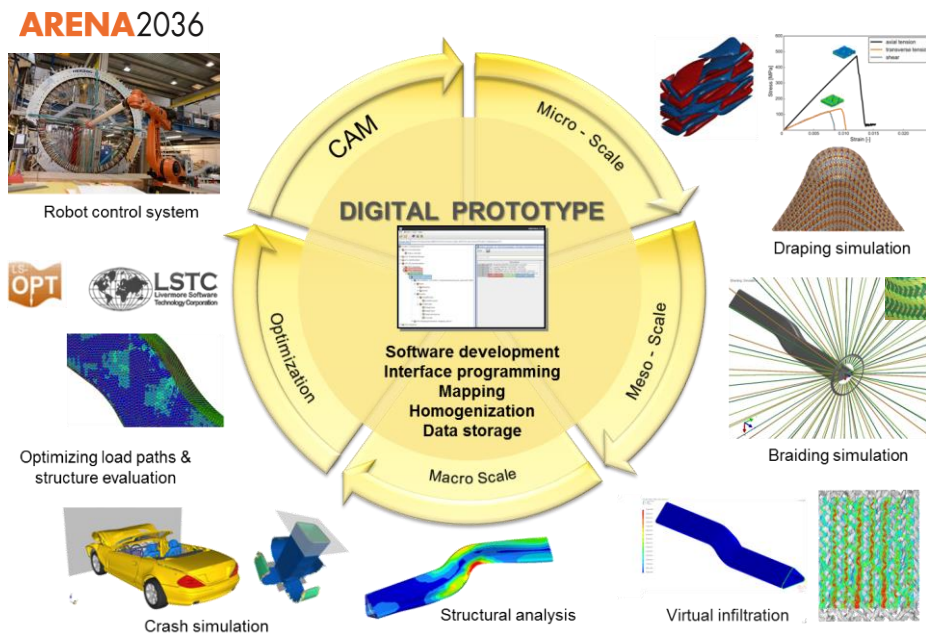CFRP materials and components were chosen as demonstrator parts.

*Figure 1:   Illustration of the various process simulations and length scales along
the simulation process chain [Böhler et. al. 2018].*

Simulations were performed with different length scales and discretizations,
taking into account the need for accuracy of the results in each of the
simulation steps along the simulation process chain. As shown in fig. 1,
different simulation disciplines are used along the process chain of a CFRP
component – in this example a braided structure. To the upper right, material
data is being generated for unknown materials, e.g. a biaxial or triaxial braid.
Therefore, representative volume elements (RVEs) are used to calculate stress-
strain responses into different loading directions and to get an estimation of the
failure properties – maximum stresses and maximum strain. These simulations
are performed on a micro- or mesoscopic level, depending on the resolution of
the single fibers or fiber bundles. Process simulations such as braiding, draping
or even the infiltration process are usually performed on a meso- or
macroscopic scale. Thereby, smaller element sizes can be chosen to model a
draping process compared to the standard element sizes being used in a
structural analysis which usually uses element sizes of 3 – 5 mm. To increase
accuracy, research institutions tend to use even beam or shell elements to
model the fiber bundles in the braiding, weaving, or draping processes. Resin
transfer molding (RTM) analysis can also be performed on different length
scales. Depending on that, rather complex solution methods such as
computational fluid dynamics (CFD) are used to calculate the locally varying

porosities and to consider them on a macroscopic RTM analysis [Dittmann et al. 2017]. On the other side, crash or structural analysis tend to use rather coarse meshes due to restrictions on the simulation time, e.g. a full car crash analysis should run over night – so no longer than 14 to 16 hours. In a following step, an optimization step should be possible to improve the component design, also considering production process specific needs. Finally, a link to a machine, in this case a weaving or braiding machine should be possible, allowing the translation of simulation input data into machine code such that a full computer aided manufacturing (CAM) process is possible.

Table 1:   Overview of the various software tools being used throughout the project for the different processing steps as well as resulting data.

Process stage	Software tool(s)	Discretization	Resulting data	Data type
Preliminary design	OptiStruct®	Shell	fiber orientations	vectorial
			thickness distributions	scalar
NVH	MSC-Nastran®	Shell	requirements fulfilled	boolean
Material characterization (RVE)	LS-DYNA®	Solid	stiffness	scalar, but direction dependent
			strength	scalar, but direction dependent
Braiding simulation	LS-DYNA®	Beam	fiber orientation	vectorial
			stresses	tensorial
	Pam-Crash®	Beam Shell	fiber orientation	vectorial
			stresses	tensorial
			thickness	scalar
ORW simulation	LS-DYNA®	Beam	fiber orientation	vectorial
RTM analysis	Pam-RTM®	Membrane	filling degree	scalar
	Open-FOAM®	Shell Beam	porosity	tensorial
Draping analysis	LS-DYNA®	Beam Shell	fiber orientation	vector
			stresses	tensorial
			thickness	scalar
Structural analysis	LS-DYNA®	Shell	Requirements fulfilled	boolean

As shown in the center of fig. 1, all the data being generated shall be stored in a readable, easy accessible data storage container, allowing to track the development status of a component from a first material characterization until the translation into machine code for an optimized component manufacturing process. Various challenges regarding that data storage were faced: the

simulation result data comes from different sources, since different software tools were used, the resulting data may be scalar, vector, or tensor data – the latter being direction dependent. Additionally, the expected amount of data can be very large, especially in the case when several optimization loops are performed. Table 1 gives an overview of the various simulation tools being used for the different processing steps within the project consortium, the discretization being used, and the resulting data that shall be stored and transferred to other processing steps. Furthermore, the information about the type of data such as direction dependent data (vectors and tensors) as well as scalar values is provided. The list is not complete, but depicts the main characteristics worth considering in following processing.

Referring to table 1, at least six different software codes have been used in DigitPro for the different simulation process steps, which results in six different result output formats, both ascii or binary based. From a discretization point of view, 1D, 2D and 3D elements have been used, depending on the chosen degree of discretization. Main result data that is being transferred are fiber orientations, thicknesses and stresses. These data sets include scalar, vectorial and tensorial data types, the latter needing special handling since they are direction dependent and therefore need to be stored with reference to a coordinate system.

From a practical point of view, the following requirements have been identified regarding the data storage container: it should be able to handle a large amount of data, the storage format should be independent of a proprietary simulation software and should be easily adaptable. Furthermore, a user interface which allows easy access to the stored data for evaluation purposes would be preferable. Due to these requirements, the available open source HDF5® format developed and maintained by the HDF-Group has been chosen. It allows a binary data storage and can handle a large amount of data. The storage container is independent of other software codes and allows the user to create a data storage format depending on his needs, such as a *COMPOUND* data format. An existing Java-application called HDF® View allows for visual inspection of the file's content.

In the following, an overview on the developed HDF® storage container will be given.

2. *Resulting Interface Definition within DigitPro*

In a first approach, the consortium tried to define a common data format for various output data such as nodes, elements, connectivity's, fiber orientations and so on. Therefore, HDF5®'s capability to store *COMPOUND* data types has been used extensively. As shown in fig. 2, a *COUMPOUND* data type to store nodes for finite element analysis has been created, allowing to have an integer value at the beginning, followed by three floating point values describing the

node's position in a 3-dimensional space, and two further integer values allowing to restrict translational and rotational degrees of freedom. This format is highly aligned with the ascii – input being typically used within LS-DYNA®

	NID	x	y	z	tc	rc
0	4	61.3541717529297	-1.5959130525589	395.844604492188	0	0
1	211	60.294605255127	1.67197620868683	396.337982177734	0	0
2	418	61.7202301025391	5.09902572631836	396.141662597656	0	0
3	625	61.1797904968262	8.86372089385986	396.460601806641	0	0
4	832	59.8460655212402	12.1788463592529	397.294982910156	0	0
5	1039	57.2126197814941	14.9763622283936	397.51123046875	0	0

Figure 2:   Nodal representation in the HDF5® data container created in DigitPro.

input definitions.

This also leads to one of the problems that the consortium was facing. Even though several software tools were used, only one software vendor was working within the group, so that the data storage creation was very similar to the already existing LS-DYNA® ascii – input format. This can also be seen in the compound data structure defined for fiber orientation storage on element level, fig. 3. Therefore, a compound data structure with six integer values at the beginning of the list was chosen, followed by several columns of floating point values, depending on the number of fiber orientations assigned to one of the elements. The first integer value represents the element ID, the second integer value receives information about the part ID, and then, this is followed by the node IDs. Depending on the element type, more than four nodes can be defined. The integer values are followed by floating point values, describing the material direction within the element. This can also be more than just two values, depending on the laminate layup being used in the simulation.

Nevertheless, this example also illustrates why only one software vendor is not enough to define such a standard. It can be seen that in this case, triangular elements are defined, using equivalent IDs on node 3 (n3) and node 4 (n4) input. This is also typical for LS-DYNA®, when using triangular shell elements in a mesh definition. Furthermore, the angle input following the element connectivity definition has to refer to a specific coordinate system. In this case, the element coordinate system defined by the direction from node 1 (n1) to

	eid	pid	n1	n2	n3	n4	angle #1	angle #2
0	1	3	35	10921	10925	10925	-161.78	150.026
1	2	3	4	10926	10930	10930	-142.273	-16.4673
2	3	3	5	10928	10934	10934	37.0293	-26.101
3	4	3	8	10935	10939	10939	-162.679	147.932
4	5	3	10	10940	10944	10944	-159.063	-19.5961

Figure 3:   Element with orientation definition as represented in the HDF5® data container created in DigitPro.

node 2 (n2) and the element normal is chosen as reference coordinate system. This has the advantage that the mesh can be transformed within the coordinate

space without having to change these values. Nevertheless, some 3rd party software might not be aware of this specific element coordinate system definition and therefore may not make use of these values within a mapping process. An alternative would be to define the orientation as a vector in a global coordinate system for shell elements. Then its meaning would be rather obvious to anybody, but it would have to be recalculated with any mesh transformation and the number of floating point values would increase by a factor of three.

Having both, integer and floating point values within one column can be advantageous when it comes to the readability of such a data storage container, e.g. when inspecting the contained mesh files using HDF®View. When looking at storage size and in- and output speed of such *COMPOUND* data types, it might be beneficial to have just a listing of any of this information in separate lists, taking into account the loss of readability. For this case, a clear documentation of the storage container is needed such that any 3rd party software can retrieve the information it is containing.

In order to cover all these needs and unknowns, a project consortium dealing only with the specific data format issues would be beneficial. Therefore, it was decided to focus on the development of a clear component development tracking scheme that allows all users working with the *Digital Prototype* to easily assign data to the data container and to follow the development status that has already been reached for a specific component. In the following, all data not referring to elements or nodes were pushed into the data container as an array of characters which can be easily read using the listed format and is obvious to anyone who is aware of the specific wording being used within the specific software code.

The HDF5® data container as it is proposed by *DigitPro* can be initialized using an ascii file containing the preferred file structure. This file structure can be changed during the product development, which might be necessary in case that new processes have to be considered. The initialized file structure is shown in fig. 4, lower right. Subfolders are initialized, using indentions in the ascii file defining this initial file structure. This structure is also available for inspection in the initialized *000_PROCESS_STATUS* subfolder. This is the only folder name which is reserved and its name may not be changed or overwritten. This folder also contains a list which is named *status_overview* which is displayed to the upper right in fig. 4. It contains information regarding the input order of files being stored within the *Digital Prototype*. The first input is the HDF5® file definition. The information is stored together with a given time stamp indicating date and time of data assignment to the container. This is followed by information regarding the folder where the data is being stored in. This should also give information about the process status. Furthermore, information about the file format which is being stored is provided following the name of the subfolder containing the data. In the example case given in fig.

4, CAD data given in *step* format has been written into the data container after project initialization. Then, an optimization was performed, the result file was provided in a *NASTRAN* format. Afterwards, a process simulation has been performed, using the Pam-Crash® solver. The resulting data are then mapped onto a mesh for crash simulation and a mesh for RTM analysis. Furthermore, a subfolder for a 30 degree biaxial braided structure, using 24k fiber bundles and a fiber volume fraction of 62 % is created that contains analysis results from a virtual testing simulation in axial tension and compression, transverse tension and compression and shearing. These original data are analyzed and the resulting engineering constants such as the Young's and shear moduli, failure strength and the corresponding strain as well as remaining stress levels after failure are stored and can be used directly with one of the mapping routines developed within the project. It allows clustering material data depending on
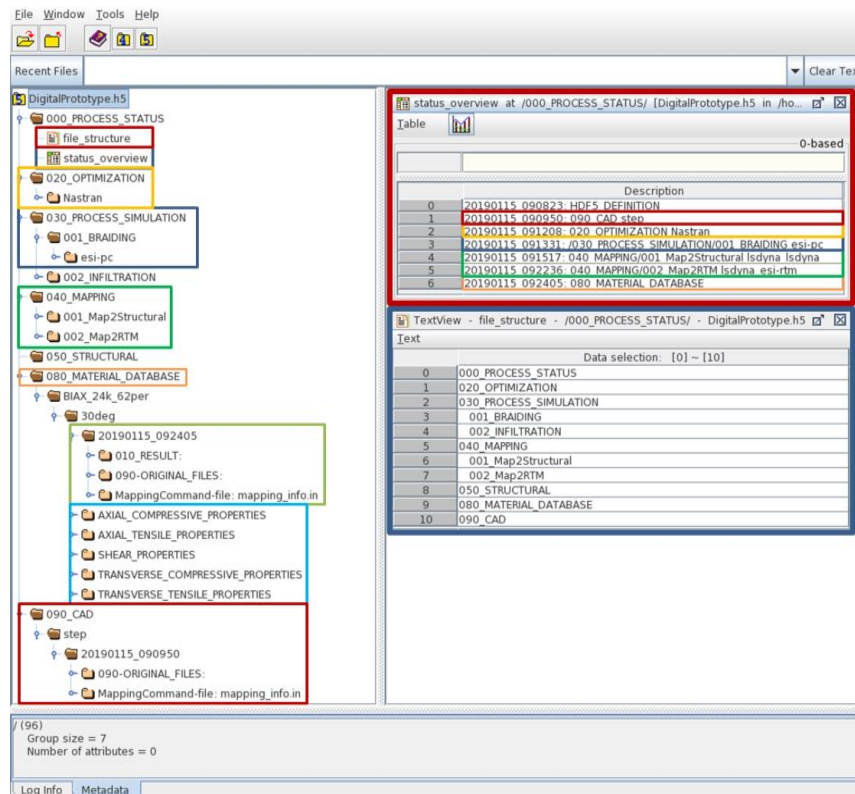

Figure 4: A snapshot of the developed digital prototype, showing the stored data (left), the development of the project (upper right) and the initialized file structure (lower right).

the local properties of the RVE which can be determined using process simulations.

Input and output routines for the Digital Prototype have been realized within envyo® which is developed by DYNAmore. The input into the HDF5®

container can be controlled by an ascii command file which is read by the program and the commands will be executed.

3. *Enhancements through the VMAP Standard*

As mentioned in the previous section, it was not possible for the consortium of *DigitPro* to define a standard exchange format based on a *COMPOUND* data type which allows to easily export and read result data from the different solvers being used (see tab. 1). Instead, the group of researchers focused on a clear data storage structure within the *Digital Prototype* and a closed simulation process chain for ORW and braiding processes. Therefore, necessary mapping routines have been developed which close the gap between processing and structural analysis. These routines are described in Section 5.

Within the ITEA-VMAP project, which runs from September 2017 until September 2020, a new data structure and result output standard shall be defined which allows for an easy exchange of simulation result data and also facilitates the reading and interpretation process for people that are not specifically an expert for one of the software codes. Therefore, it will be easier for other software developers to create data mapping interfaces between the different process and structural simulation codes [Wolf et. al 2018]. The consortium, which consists of about 30 different partners, covers a large variety of software tools which are either distributed by the partners or even developed by them directly. An overview on the software codes used within the project and the respective vendors or developing companies is given in tab. 2. This table also indicates what the software can be used for – either for pre-, post-processing, or mapping purposes and the type of analysis for which the software can be used for. This can be either process simulation or structural analysis. This list is not complete; other software solutions are being used within the project as well but are not supported by software vendors or developing companies during the format definition.

From the list of software developing and selling companies within the ITEA-VMAP project, one can see that the group is very different to the one working within the *DigitPro* research project. Having a group of software developing companies allows to discuss the format specifications and data storage container questions in more detail then it could be done with a group consisting of software users. Nevertheless, it is also important to consider their needs as well. Therefore, seven use cases have been defined which shall show the benefits of a closed simulation process chain and demonstrate, how easy it can be to switch between the different solvers, once a standardized simulation result output format is defined. These use cases shall cover the following topics: blow molding of packaging and large containers, composite design (continuous fiber reinforcements) for lightweight vehicle structures and in aerospace industries, injection molding with structural and fatigue analysis, hybrid modeling, such as over-molding or multi-step processes, and additive

manufacturing. These use cases shall evaluate the defined VMAP standard with exchangeable software solutions for structural and process analysis and allow transferring the data properly between the different software tools. By the end of the project, various results and locally varying material properties shall be stored on an HDF5® based data container which can be written and read by the different software codes and therefore, there is no format issue for the user when switching between the different software suits. Generic demo examples shall be available to users to test the VMAP format and to evaluate their process chain requirements. The format shall be enhanced and developed over the years following the end of the project by a NAFEMS working group and shall incorporate standards which are developed by other groups such as the EMMC (European Material Modeling Council), the CCMRD (Canadian Composites Manufacturing Materials Management Consortium), the German VDA FAT (Research association of Automotive Technology) [Wolf et. al. 2018], or the ISO STEP AP209 standard [Boy 2018] which is based on standardization of CAD data given in step format and now works on enhancements for CAE result and material data.

Table 2: *Overview of the different software tools for FE analysis and their distributors/developers participating in the VMAP project.*

Software	Vendor/developer	Simulation stage
4a Fibermap	4a engineering	Material modeling Mapping solution
ANSA	beta-CAE	Pre- and Postprocessing Mapping solution
Cadmould	Simcon	Process simulation
Digimat	e-Xstream	Material modeling Mapping solution
envyo	DYNAmore GmbH	Mapping solution
LS-DYNA	DYNAmore GmbH	Process simulation Structural analysis
MpCCI	Fraunhofer SCAI	Mapping solution
MSC.Marc	MSC	Structural analysis
Open-FOAM	Open source ESI	Process simulation
PAM-Crash	ESI	Structural analysis
PAM-Form	ESI	Process simulation

4. Overview on the VMAP Use-Cases

For this section, three use cases were selected that shall demonstrate the advantages of having an interchangeable result file format: the continuous fiber reinforced plastics (CFRP) use case, the short fiber reinforced plastics (SFRP use case and the bending beam use case. It should be mentioned, that the way that the simulations shall be performed in the different use cases is not the standard approach for every simulation tool. Nevertheless, it allows having comparable results based on similar problems and therefor, demonstrating the applicability of the format to other users.

4.1 Bending Beam Use Case

The simplest use case defined within the VMAP project is the bending beam use case. Figure 5 illustrates the simulations which shall be performed. In the first case, a temperature difference of 40 K is assigned to the boundaries compared to the surrounding temperature and the temperature of the beam. It is assumed that the beam will change in length and width due to the thermal loading. The other test performed on the bending beam applies a temperature difference of 40 K at the bottom of the beam, such that it will bend upwards. This is of course dependent of the material properties which are nevertheless irrelevant to show the data exchange capabilities of the defined VMAP standard. In a first step, the thermal analysis shall be performed and the result data stored in the VMAP data container. After mapping, the structural analysis can be performed using the input from the thermal analysis, maybe on different discretizations or with another software code. Analysis will be performed using both 2D and 3D elements. For this use case, first simulations with LS-DYNA[®]
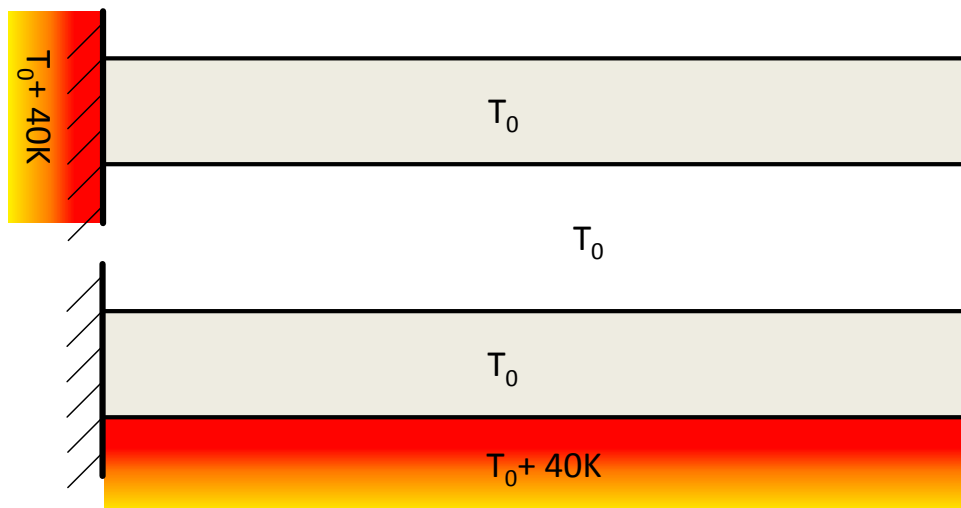

Figure 5: *The bending beam use cases temperature increment on the left (upper picture) and temperature increment on the bottom (lower picture).*

have been done, but the internal coupling capabilities of the code have been used for the following analysis. Therefore in the near future, an output to the VMAP standard has to be generated for the thermal results which can then be mapped onto the structural mesh.

4.2 SFRP Use Case

The second use case with LS-DYNA® being involved is the SFRP use case. Therefore, injection molding simulations will be performed with different software codes such as Cadmould 3D-F®, Moldex3D®, and Moldflow® and the resulting fiber orientation tensors shall be used together with respective homogenization techniques or micro mechanically motivated material models to simulate the structural response after a specific loading condition. In this use case, it is even necessary to switch the software code since the above mentioned analysis tools only consider the processing step and have limited modeling capabilities for structural analysis. To make the process analysis results easily accessible to the structural FE codes, it is quite handy if the data is provided in a standardized format – in this case the VMAP standard and therefore can be interpreted easily since any mapping tool or software code will only need one parser to read the specific data. For now, the molding analysis has been performed, the mapping tool envyo® has two parsers which allow for the consideration of Moldflow® and Moldex3D® analysis results with LS-DYNA® material models *MAT_ANISOTROPIC_ELASTIC_PLASTIC (*MAT_157) and *MAT_4A_MICROMEC (*MAT_215) [Reithofer et. al.]. For the Cadmould 3D-F® result, either a third parser has to be implemented into envyo® to allow for the mapping to LS-DYNA®, or the provided VMAP format can be used to initialize the required material properties within the structural analysis. The latter would simplify the analysis with other software solutions and mapping processes significantly.

4.3 CFRP Use Case

The CFRP use case shall demonstrate the benefits of the VMAP standard for the process chain of such materials. The following processing and analysis steps should be performed: a draping process, an injection molding process, a curing process, and a structural simulation. For each stage, different meshes are chosen, as it is shown in fig. 6. For the draping analysis, triangular shell elements shall be used; for the infiltration stage, one hexahedral solid element across the thickness of the component is chosen. The curing analysis uses four solid elements over the thickness to get better spring-back results due to bending of the component, and the structural analysis shall use standard quadrilateral shell elements. Therefore, not only different process parameters shall be transferred between the different process stages, the resulting data are stored on different integration point positions as well, and elements use different shape functions so that information regarding element types and discretization has to be stored based on a neutral format. In general, two

different mapping scenarios are conceivable: one follows all the steps of the simulation process chain, the other one makes a direct transfer of the draping results onto the structural mesh. With the used mapping software envyo®, the latter is already realized and the process chain is closed assuming that both,
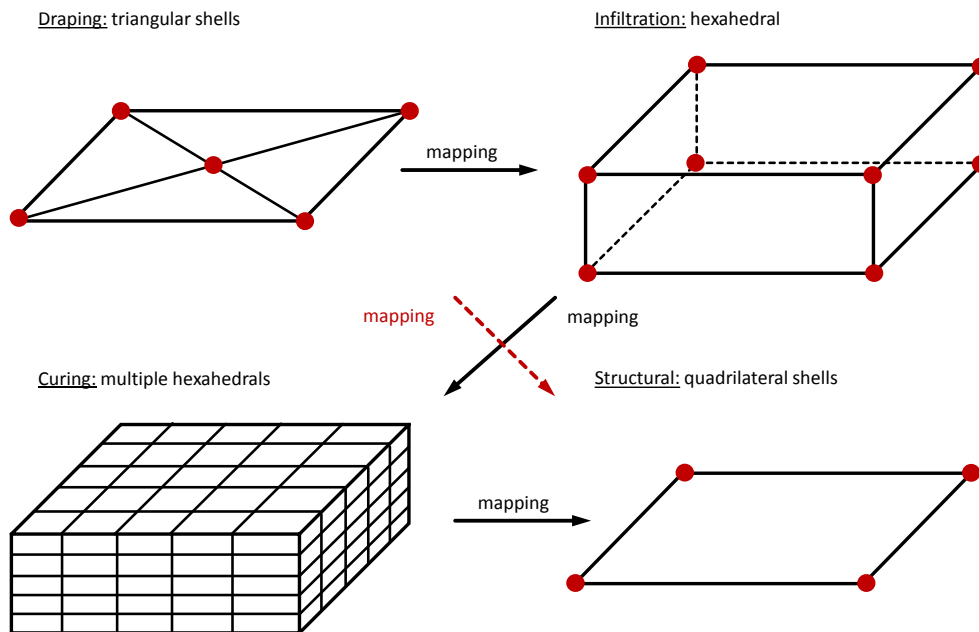

Figure 6:   The CFRP use case and the different discretizations used for draping (u.l.), infiltration (u.r.), curing (l. l.), and structural (l.r.) analysis.

draping and structural analysis are performed with LS-DYNA®. Nevertheless, future work will investigate mapping and analysis capabilities for infiltration and curing. For infiltration, CFD (computational fluid dynamics) methods and ALE (Arbitrary Lagrangian Eulerian) methods are available in the software code. For curing, the additional CURING option in *MAT_ORTHOTROPIC_THERMAL might be beneficial when closing this third step of the simulation process chain [LS-DYNA I & II 2019].

5. *Mapping Routines Implemented for Mesoscopic CFRP - FE Models within ARENA2036*

Even though there was no ongoing standardization process for simulation result data within the project Digital Prototype, several mapping options have been realized in envyo® to transfer simulation result data from mesoscopic shell and beam models onto shell meshes for structural simulation. Therefore, parsers for reading ESI – software result formats and input data (Pam-Crash®, Pam-RTM®) have been developed (see also tab. 1). Main information that can be transferred from process to structural analysis is the fiber orientation, the

thickness and the identification of resin-rich areas. For the fiber orientations, the direction was assumed to be aligned with the numbering of the shell element's nodes, such that the connection from node#1 to node#2 was assumed to be equivalent to the fiber direction. This assumption had to be made since the provided result data did not contain orientation information. This would be overcome by a standardized output as it is created within the ITEA-VMAP project. For beam elements, the orientation was assumed to be aligned with the axis of the beam elements.

5.1 Closest Point Approach

The first and simplest mapping routine being implemented searches for each element in the target mesh the nearest fiber direction of each part of the source mesh being defined in the mapping command file. It also considers the ordering of the parts as they are given in this command file and assigns the direction to the integration points following this definition. Furthermore, it does not consider any thickness offsets between the fibers in the source mesh – and therefore, the thickness is a user input variable which is then split by the number of source parts which should be searched for and therefore, the integration point position will be equally spaced. This method is also referred to as the closest point approach in the context of this mesoscale-specific mapping routine.

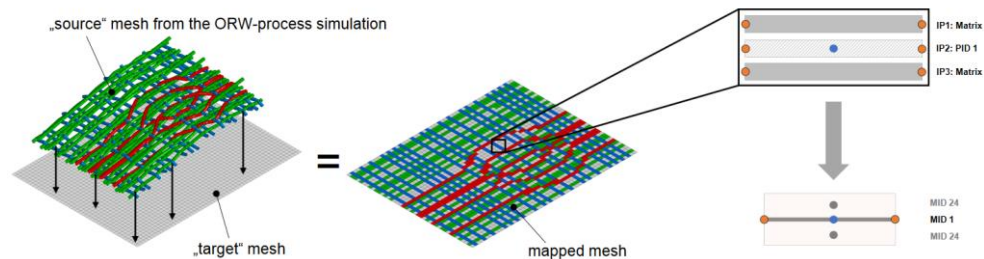## 5.2 Consideration of a Search Radius

Figure 7:   Mapping of a mesoscopic ORW simulation result onto a structural mesh considering a search radius and equally spaced thicknesses [Holzapfel et.al 2016].

The second routine which was realized is an enhancement of the first routine, considering a search radius. In case that not all parts of the source mesh are found within a user defined search radius starting from the target mesh element center, a matrix material ID, which is also defined by the user will be assigned to the respective integration point. With this approach, not all integration points through the thickness are assigned a reinforcing material property, but also isotropic and weaker elastic-plastic material properties as they are observed for

resinous materials can be considered. As in the first approach, the thickness is distributed equally through the thickness, regardless of the fiber or matrix material assigned to the integration point. Figure 7 illustrates this method of found and not found fiber orientations for an open-reed weaving example.

5.3 Consideration of Undulation

The third method that has been realized now also considers the undulation of the fibers, which means that the assigned orientation on the integration points on the target mesh are not sorted anymore with respect to the input order of the source part IDs in the mapping command file. Instead, the orientation which is closest to the element center of the target mesh in normal direction of the element is assigned to the respective integration point. Furthermore, this method allows considering the locally varying thicknesses by calculating the offsets in normal direction between the fibers and taking into account the thickness being defined in the input file. In a first step, the thickness $t_{transfer}$ that should be mapped from source to target mesh is calculated using the maximum distance between fibers found in the source mesh and by adding half of the thickness $t_{min/max}$ of these fiber bundles:

$$t_{transfer} = \overrightarrow{|p_{min}p_{max}|} + 0.5 * (t_{min} + t_{max}) \tag{1}$$

$p_{min/max}$ are the center positions of the found elements on source side. In a next step, the resulting thickness due to the found source part IDs is calculated, taking into account that due to a search radius, not all integration points get assigned with an orientation dependent fiber material ID. Therefore, the resulting thickness of transferred fiber directions is given as:

$$t_{th} = \sum_{i=0}^{n_{Fibers}} t_i \tag{2}$$

where n_{Fibers} is the number of found fibers in the source mesh, t_i the corresponding thickness of each of the fibers and t_{th} the resulting theoretical thickness, not considering matrix material. Then, the thickness assigned to the matrix integration points is calculated by subtracting the transferred thickness from the thickness given by the found fibers and dividing it by the number of integration points assigned with a matrix material ID. In a next step, the mapping routine checks if the transferred thickness is greater or equal the computed fiber thickness. If that is the case, there seems to be no compaction and therefore, the thickness from the input is assigned to the integration points, using a fiber material, and the thickness of the matrix integration point is increased equivalently for each matrix integration point to consider additional matrix enriched zones. For the case that the transferred thickness is smaller

than the resulting total thickness which is calculated using the fiber thickness, it is assumed that there is a compaction of the fibers. Therefore, all integration point thicknesses t_{IP} are scaled to fit into the transferred thickness by the following equation:

$$t_{IP} = \frac{t_{IP,sec}}{t_{th}} * t_{transfer} \tag{3}$$

Furthermore, the user may define a maximum thickness which overwrites the computed thickness $t_{transfer}$ and the methods mentioned above are applied to that user defined thickness. A result of this third method is shown in fig. 8.
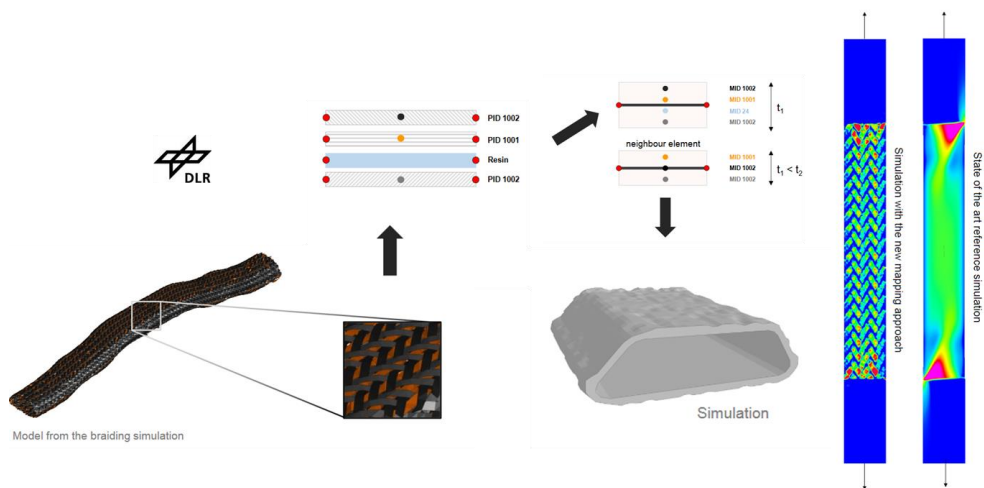
Figure 8:   *Mapping of a mesoscopic braiding simulation result onto a structural mesh considering a search radius, undulations and varying thicknesses [Vinot et.al 2018].*

With this approach, it is possible to get a more realistic stress distribution compared to a standard approach that assumes equally distributed orientations and thicknesses throughout a component (see fig. 8).

5.4 Detection of Representative Volume Elements (RVEs)

The final mapping routine that has been realized allows detecting representative volume elements (RVEs) within a result mesh of a mesoscopic process simulation. The method is also based on a user defined search radius and assigns a set of material parameters being derived from virtual testing to the specific elements. Therefore, the mapping routine detects the number of fibers pointing into similar directions and therefore can decide if the found fibers belong to a local biaxial or triaxial braid. Given the information of the part ID of the reinforcing fibers, the corresponding reference angle between the

fibers can be calculated. With that information, the program can look up
material properties given in a database stored in the above described HDF5$^{\circledR}$
data container (the Digital Prototype) and select the one which fits the most to
the found orientation characteristics. If only one fiber direction is found,
properties of a uni-directional (UD) material can be assigned. The idea of the
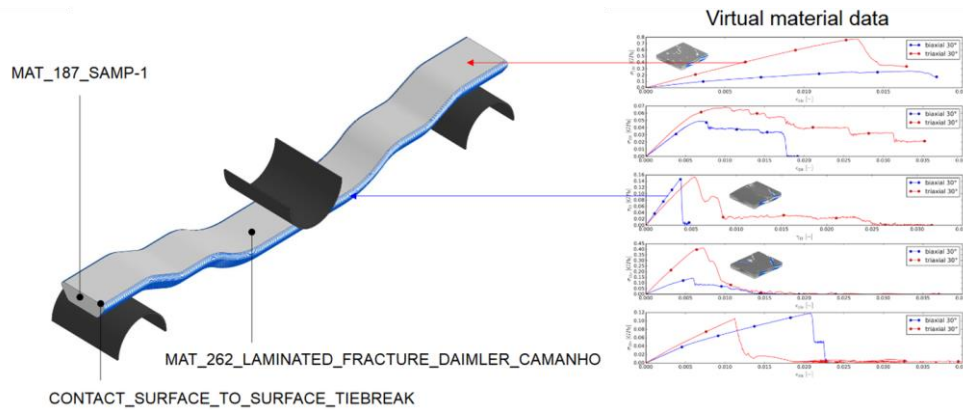method is illustrated in fig. 9.

*Figure 9:   Mapping of virtual material data onto a structural analysis [Vinot et.al
2018].*

6.   Conclusion & Outlook towards the Development of the Digital Fingerprint

Within the research project *DigitPro*, first steps towards standardization of
simulation result data have been made. Due to the group of researchers and
supporting companies from the industry, it was not possible to define a new
standard for simulation result data output. Instead, the consortium concentrated
on defining a data storage container which tracks the simulation and
development history and with developed mapping routines bridges the gap
between microscopic or mesoscopic process simulation and structural analysis.
Therefore, different parsers had to be implemented into the mapping tool
envyo$^{\circledR}$. With the new VMAP standard introduced in this paper, this work will
not be necessary any more for the various software codes. Instead, only one
parser needs to be written that can understand the VMAP result data as well as
integration rule and element definitions being stored in that format. VMAP
therefore simplifies the work of mapping software developers significantly.
Furthermore, users of specific software codes can switch rather easy between
the different software tools depending on the simulation tasks needed along the
simulation process chain. Within the VMAP project, the applicability of the
defined standard will be evaluated using the described use cases for a standard

bending beam, a SFRP component and a CFRP component simulation process chain.

Within the just started research project *Digital Fingerprint* which is also part of the ARENA2036 research campus [Böhler et. al. 2018], the already created link between the different simulation tools for the various process stages and the created digital prototype shall be stored together with sensorial measurement results which are collected during the production of a real component and during a component's life-time. Therefore, it is thought to use the recent status of the VMAP standard to easily store and transfer the generated simulation data. Therefore, its applicability can be tested within a semi-industrial environment already. Furthermore, simulations shall be improved taking into account these sensor measurements and may help to predict remaining strengths of a component after unexpected events measured during its usage or already at manufacturing stage.

7. Acknowledgements

8. References

J. Dittmann, P. Böhler, D. Michaelis, M. Vinot, C. Liebold, F. Fritz, H. Finckh, P. Middendorf: "*DigitPro – Digital Prototype Build-up Using the Example of a Braided Structure*", IMTC, Chemnitz, GER, Oct. 2015.

P. Böhler, J. Dittmann, H. Finckh, F. Fritz, A. Haufe, C. Liebold, M. Holzapfel, M. Vinot: "*ARENA2036 & the Digital Prototype: Introduction and Overview*", German LS-DYNA User's Forum, Bamberg, GER, 2018.

K. Wolf, G. Duffet, A. Flyod: "*ITEA VMAP – A new Interface Standard for Integrated Virtual Material Modelling in Manufacturing Industry*", NAFEMS DACH Conference, Bamberg, GER, 2018.

J. Boy: "*Simulation Data Exchange based on standards – STEP AP209 (not only) for long-term archiving*", NAFEMS DACH Conference, Bamberg, GER, 2018.

Lessons Learned from Developing a Digital Prototype within the ARENA2036 Environment and Improvements with the new VMAP Standard

P. Reithofer, A. Fertschej, B. Jilka: "*MAT_4A_MICROMEC – Generating Material Card and Considering Fiber Orientation*", German LS-DYNA User's Forum, Bamberg, GER, 2018.

M. Holzapfel, M. Vinot, C. Liebold: "*A Multiscale Approach for the Mechanical Investigation of Textile-Based Composite Structures within a Closed Process Chain*", German LS-DYNA User's Forum, Bamberg, GER, 2016.

M. Vinot, M. Holzapfel, C. Liebold: "*A Multiscale Strategy for the Simulation of Braided Composites with ENVYO*", German LS-DYNA User's Forum, Bamberg, GER, 2018.

"*LS-DYNA$^®$ Keyword User's Manual - Vol. I*", Livermore Software Technology Corporation (LSTC), 2018.

"*LS-DYNA$^®$ Keyword User's Manual - Vol. II – Material Models*", Livermore Software Technology Corporation (LSTC), 2018.

Dittmann, J., Hügle, S., Seif, P., Kauffmann L. and Middendorf, P., "*Permeability Prediction Using Porous Yarns in a Dual-Scale Simulation with Openfoam*", ICCM21 - 21st International Conference on Composite Materials, Xi'an, China, Aug 2017