Chapter 4
Virtual Product Creation (VPC) Explained

Executive Summary

This chapter deals with the following topics:

- Understanding of the term and development element *virtual product*,
- Explanation the activities within and the capabilities of *Virtual Product Creation* and
- The difference and relationship between *Virtual Product Creation* and the overall concept *PLM (Product Lifecycle Management).*

Quick Reader Orientation and Motivation

The intention of this chapter is:

- to provide an overall understanding of the discipline *Virtual Product Creation (VPC)* as part of the product lifecycle concept and to map the VPC activities throughout the lifecycle engineering spectrum;
- to explain the principle elements of a virtual product;
- to introduce the variety of digital models as part of the virtual product elements;
- to provide an understanding how *Virtual Product Creation* differs from other digital enterprise capabilities;
- to introduce the major capabilities of *Virtual Product Creation* in the context of technical system development, digital manufacturing and service development.

4.1 The New Engineering Discipline *Virtual Product Creation*

Virtual Product Creation is a discipline for digital engineering in Product Development (PD) as well as in Manufacturing/Production System Development and Process

R. Stark, *Virtual Product Creation in Industry*,
https://doi.org/10.1007/978-3-662-64301-3_4

Planning. During the last ten years, it became evident that the ongoing digital activities of Maintenance Repair and Overhaul (MRO) and ongoing product updates have also become part of Virtual Product Creation activities. Especially functional product updates via the delivery of software updates as well as algorithmic self-learning aspects as part of autonomous operations will drive regular and intense Virtual Product Creation processes. Virtual Product Creation, therefore, represents a major part of the overall digital enterprise capabilities.

Virtual Product Creation encompasses many engineering activities, which in contrast to the traditional engineering approach no longer uses physical but virtual elements as a "model for progression".

The term *virtual* implies that those elements do not (yet) exist in the traditional physical world and therefore only exist in a specific transient or seeming format, i.e. they are not yet real. In order to avoid that such virtual models only exist in the minds of individual human beings in their specific roles such as engineers, planers, managers, professors, students, workers etc., the power of digitization and standardization helps to make them "existent, visible and executable" on computers. Hence, the Webster dictionary explains the term virtual as, amongst others, "being on or simulated on a computer or computer network".

The term *product* means traditionally "a thing that is produced or created by labor". Nowadays, however, a product can also be a service or a software. A product is offered on a market and has a lifecycle. A product in general can be used in many different real-world environments but typically, the field of use must already be anticipated early on during its ideation. Typical product examples are:

- a specific cutting machine for usage in a factory,
- a sedan automobile for passenger transportation vehicle as part of traffic system in certain global regions,
- a coffee machine for private use in a household or as a vendor machine in a cafeteria, etc.

Since the beginning of the millennium, the concept of a product-service system evolves as a further evolution of traditional products: product service systems treat products and their offered service as an integrated offering to the market.

The term *virtual product* was formally introduced in a comprehensive manner by the researchers Krause and Spur in 1997 (Das virtuelle Produkt, the virtual product, compare [1], p. 307):

- The *virtual product* is the central information carrier for a complete computer-aided product development. The interaction with the *virtual product* is only possible via the help of modeling and verification functionalities [of a (PLM-type) software].
- The *virtual product* can be defined as a realistic computer-aided representation of a product, with all functions of the product for all life cycle phases of the product. Specifically, the activities of product planning, design/styling, design engineering, manufacturing engineering, production operation, recycling and service are addressed.

- The *virtual product*, therefore, is a computing model, which describes the product with the help of information sets and product functions as realistic as possible.

Figure 4.1 shows the definition and illustrates the basic understanding of a *virtual product*: the virtual product today is represented by a series of computer-aided models and information sets which requires the existence of corresponding modeling and simulation algorithms usually represented by different sets of IT-applications (such as CAD, CAE, CAM etc.).

The cascade of different elements associated with a virtual product is shown in Fig. 4.2: first, it is necessary to provide a *designation* or sometimes even an explanation of the virtual product in mind. If a new virtual product—or a sub-system or component of a virtual product—is "born" in the mind of a developer, oftentimes just a "nickname" or a general term is being used before assigning an official part or product designation to it. Strict naming and numbering data base regulations as part of the overall PDM or PLM environment also kick-in and require such specific designations in order to guarantee precise and reliable collaboration with other engineers and departments.

It is important that all members of the virtual product creation community associate the proper *imagination* with the virtual representation of the product. For such goal, it is essential to use symbols for recognition (e.g. icons) and simple illustrations together with short explanation of the main function, the architecture and the context of a virtual product.

In addition, it is necessary that the virtual product is represented in a certain way with the help of product model representations. As outlined in Fig. 4.2 the most common product model representations are the following ones:

- Design models (e.g. CAD, i.e. Computer Aided Drafting/Design): partial models, which describe the geometric shape and topology of a product, meanwhile often enriched by additional semantic data such as technological parameters (tolerances, reference system), functional information and manufacturing and quality inspection data.
- Behavioral models (3D CAE models) of the virtual product such as FEA (Finite Element Analysis), MBS (Multi Body Simulation) or other 3D CAE model types.
- Physics models, also called 1D CAE models: they represent the functional and/or behavioral models in a specific mathematic way within out explicit 3D geometry.
- Full selection of a high number of different virtual product models in specific light weight visualization representation, also known as DMU (Digital Mock-Up) model or even in 3D stereoscopic format known as Virtual Reality scene model.
- Various specific models to represent GD&T (Geometric Dimension and Tolerancing), or CAM (Computer Aided Manufacturing) or CAPP (Computer Aided Process Planning) for manufacturing process and tool path information capturing.

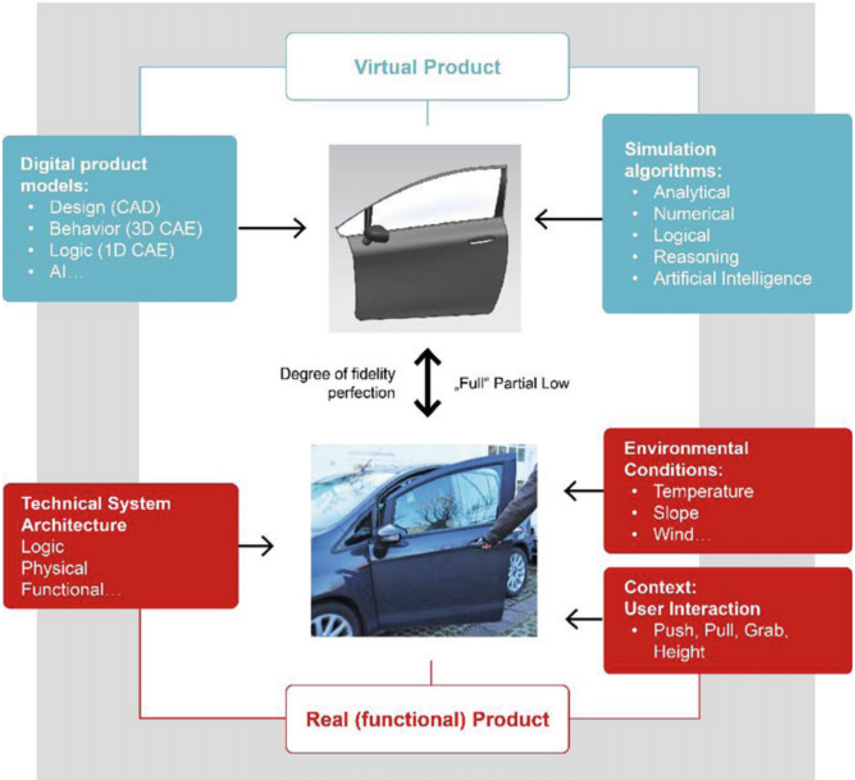
The term *product creation* describes all business, planning and engineering activities of product planning, product development or engineering and manufacturing/production system development in order to guarantee form fit and function of

What is a Virtual Product?

Definition:

A virtual product represents all - or at least major - capabilities of a real functional product by integrating various sets of digital product models with corresponding simulation algorithms. Although it exists only virtually it can be configured, used and tested as part of different physical, functional and interactive conditions and environments.

Understanding:



→ **Consequences:**

The degree of fidelity perfection match between the real product and the virtual product depends on:

A) avilable digital modeling capabilities
B) acceptable digital modeling efforts
C) needs and priorities to virtually validate, test and use real (functional) product behaviours

Fig. 4.1 What is a virtual product? Definition, understanding and consequences

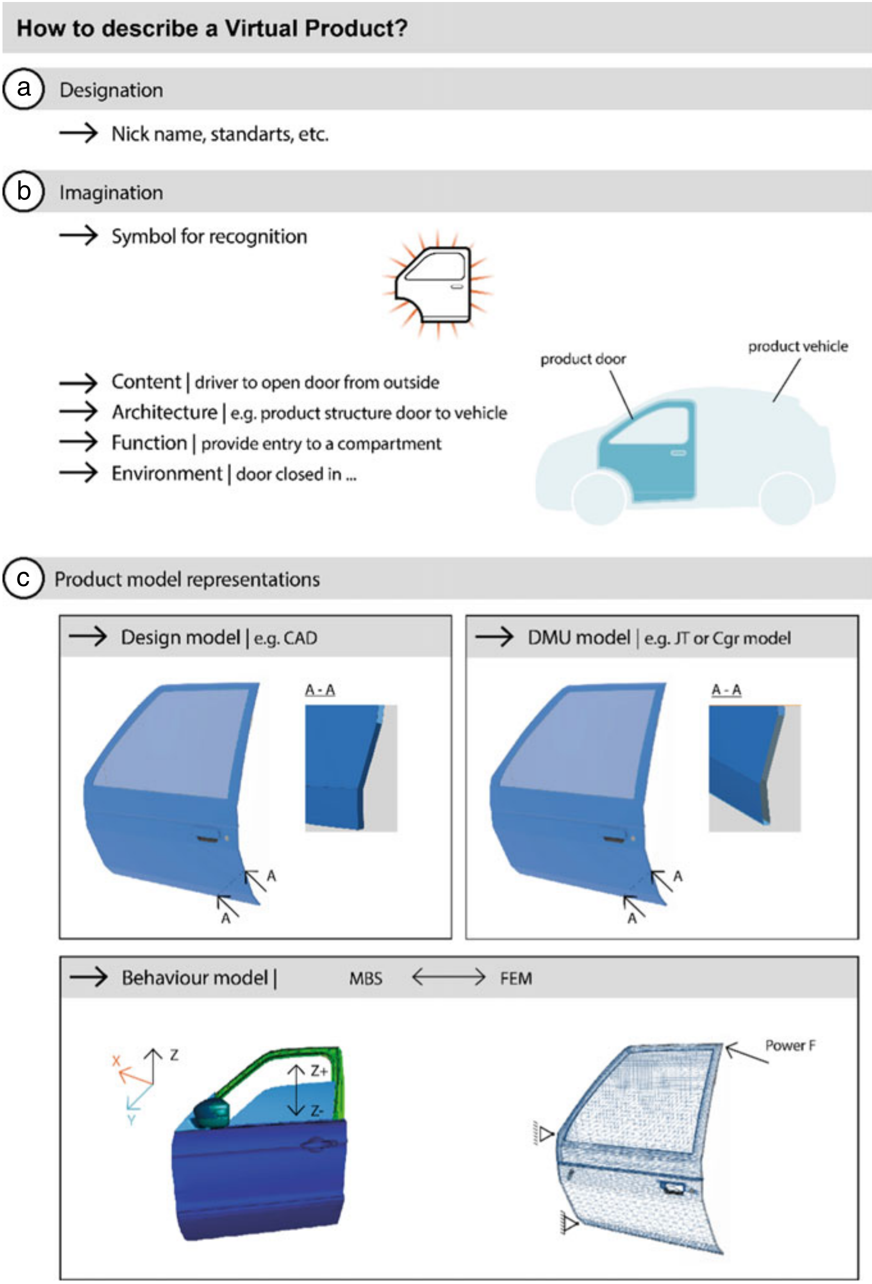


Fig. 4.2 How to describe a Virtual Product? Designation, imagination, product model representation and meta-information

the product itself and its production and operational readiness (compare [2]). In the English-speaking world there exists a synonym to the term *product creation* that is called *product realization.* However, the term product realization implies oftentimes also the notion of "putting something into reality" and, therefore, is oftentimes understood as the tail end *of product creation.* Therefore, the author recommends using the term product creation, which also nicely fits into the German term *"Entstehung"*: here, it carries a notion of "following an implicit process of growing something to its final existence".

Definition of *Virtual Product Creation*:
Virtual Product Creation constitutes of all process steps and engineering activities (and their iterations) that use digital applications, IT tool functions, software algorithms, working methods and assessment and decision capabilities to create, modify, simulate, analyze, test, validate, verify, sign-off, release and exchange *virtual products* and their derivations.

Therefore, *Virtual Product Creation* is determined by the explicit tasks and activities of engineers and the specific capabilities of computer algorithms and software, in order to define, change and use CAx models and other digital models (such as algebraic or function models) in the context of other digital data and information.

Virtual Product Creation therefore is pivotal to determine shape, function and characteristics as well as logical and physical behavior of real products under "*adjustable virtual conditions"*.

4.2 Virtual Product Creation Capabilities and Activities

The activities of *Virtual Product Creation* must be understood in the context of the overall lifecycle of products and related services. Figure 4.3 shows typical *Virtual Product Creation activities* along the product life cycle. In the beginning, when a new product idea and plan is born, the full set of system engineering and/or product development activities is started and followed up very stringently, including all associated project management and gateway execution schemas. In general, there exist different styles of development processes:

- from traditional concept design, embodiment design, detail design and physical prototype engineering and testing;
- via V-shape illustrated and phased mechatronic and systems engineering approaches including formal development phases for requirements engineering, functional modeling and networks, system architecture and behavior simulation,

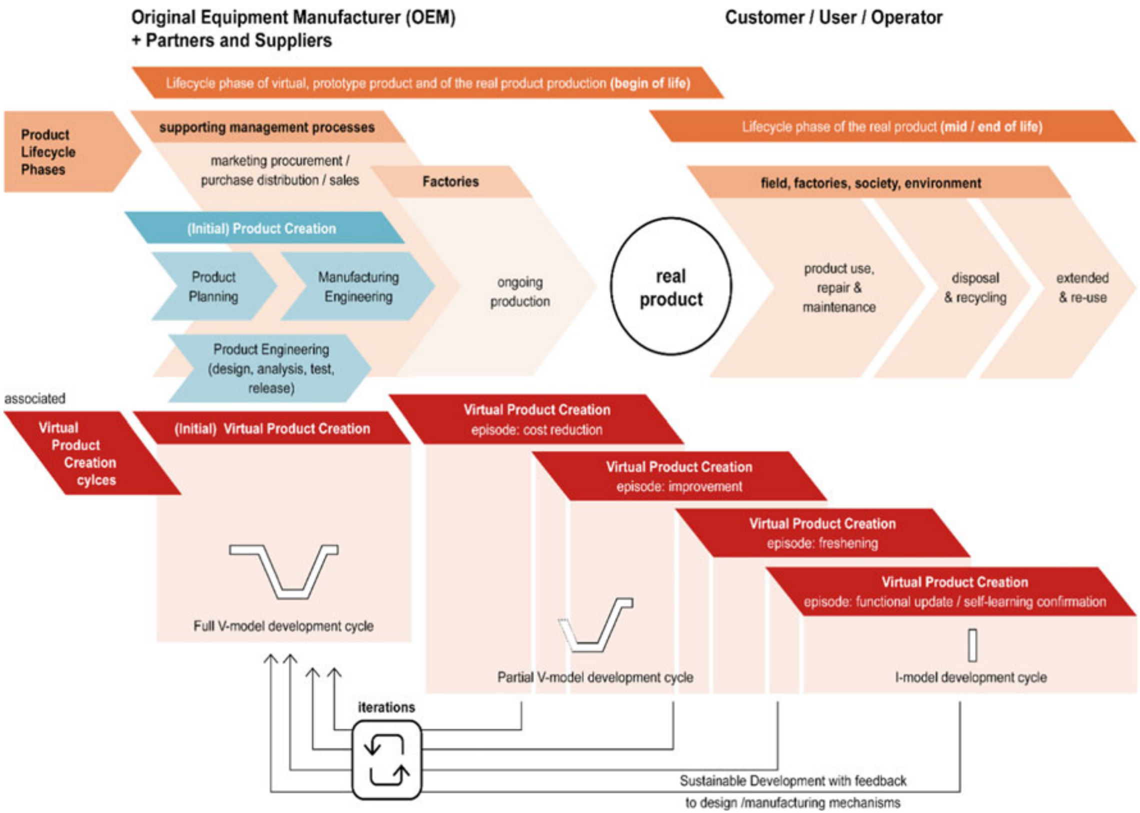


Fig. 4.3 Initial Virtual Product Creation (VPC) and VPC episodes later in the life cycle

disciplinary detail design (E/E, mechanics and software) up to different kind of prototyping, sign-off and release management;
- up to agile development activities based on incidents and/or small development task arrangements with stakeholders and costumers.

Meanwhile, all of those development and engineering styles are deeply relying on *Virtual Product Creation activities* and related capabilities such as *methods and tools*. The degree of *Virtual Product Creation execution* is a matter of *affordability* (cost and resource wise), *necessity* (need for upfront and ongoing simulation and confirmation to avoid risk and gain of clarity during the development process and its stage gates) and *complexity* (part, assembly, sub-system, full product or system and interface to other systems). Unfortunately, standards for the degrees of *Virtual Product Creation capabilities* in industrial companies and industry branches don´t yet exist. Consequently, there is a high degree of uncertainty regarding the questions of which technologies and methodologies need to be deployed in order to become and stay competitive.

Hence each company has the burden to define their own virtual product creation destiny and capability roadmap that is difficult due to the limited knowledge ordinary management holds in the subject of *Virtual Product Creation*. By the end of the 90s software vendors in Europe and in the US started to (mis-) use the term product lifecycle: originally, the new engineering discipline *Virtual Product Creation* encouraged designers and analysts to substantially create models for design and behavior simulations. Over the years, however, those activities started to create challenges and problems due to missing capabilities to store and manage computer-aided models and files consistently and safely. Consequently, throughout the 90s of last century, software vendors started to develop IT-systems for engineering data management (EDM), team data management (TDM) and finally product data management (PDM, the next level up to EDM and TDM), which provided all tool sets to name, number, store and manage computer-aided models and their associated files in the context of projects and product structures.

Around the millennium, the term PLM (Product Lifecycle Management[1]) was originally introduced by IT vendors in order to gain more traction in the mindset of business processes and the strong push of re-engineering efforts strongly supported by Senior Management and the new business sector of Business Consulting. PLM in its neutral meaning carries the idea of describing the full life cycle of a product and its surroundings such as the factory, its production systems and cells, other resources and the usage in the field. The lifecycle of a product, therefore, is divided into three phases Begin of Life (BoL), i.e. from the first idea of a product until its readiness to be manufactured, Mid of Life (MoL), i.e. the production of the product as well as its usage and maintenance in the field, and End of Life (EoL). Nevertheless, the core driver for all of the PLM embedded digital engineering and simulation activities and associated business and IT solutions are represented by the appropriate *Virtual Product Creation tasks*.

[1] Please compare detail descriptions and explanations of the term and the disciplines of PLM in [3–5] and follow the major technologies of PLM in Chap. 11.

In addition, the new influence of IoT (Internet of Things) driven data streams from already existing live products will need to get connected to the upfront models of the BoL (Begin of Life) phase of the entire product lifecycle. Such connections are established with the help of the new generation of virtual product models, called *Digital Twins*.

As depicted in Fig. 4.3 the style and intensity of virtual product creation activities might differ along the needs in the engineering and development cycles. During the last years, it became clear that more and more development episodes and specific aims of development need to be supported by virtual product creation activities. If indeed, the degree of self-learning and autonomous products and technical systems will substantially and steadily grow—as the strong efforts in the mobility and industry sector indicate—then the virtual product creation skill set needs to be transformed in order to "on the fly" simulate the consequences of AI (Artificial Intelligence) driven software control on technical products and systems in the context of the environment.

The core competence *Virtual Product Creation* provides a rich and complex mix of capabilities that consists of digital processes, methods, tools and model/information/data objects. Figure 4.4 depicts the situation for technical system development, which includes product development. *Virtual Product Creation* is executed within a *domain layer*, the *system development layer* and the *application layer* with specific digital modes that correspond to specific development tools. In

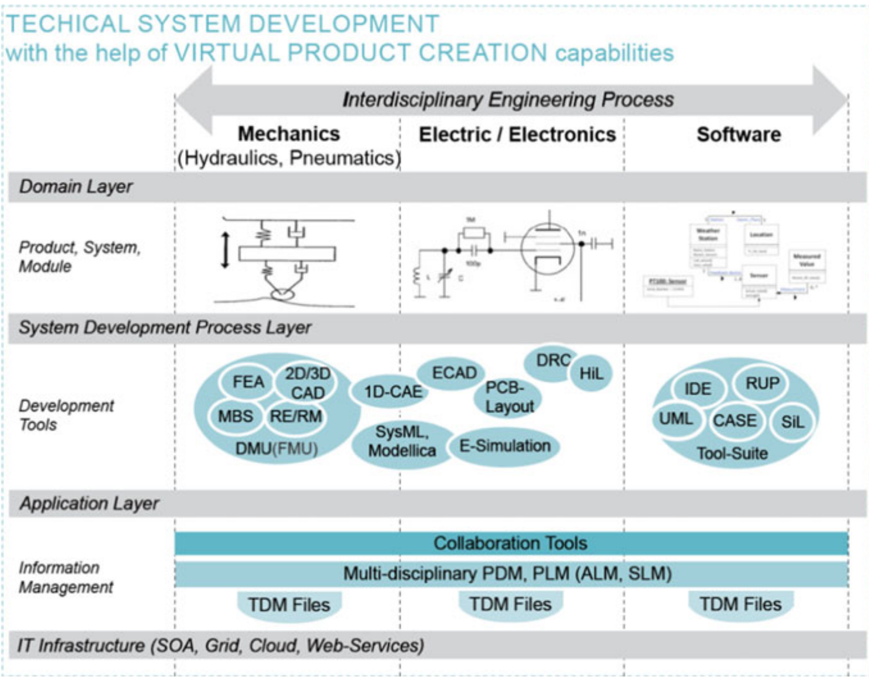


Fig. 4.4 Layer view of Virtual Product Creation capabilities for technical systems development

order to maintain, manage and share all of those models and information it is essential to maintain and organize a substantial information management environment based on a robust *information technology (IT) infrastructure layer*. Information and data management environments do offer a wide range of handling of documents, models meta data and administrative data within the context of user-oriented functions and underlying data bases: starting from *Team Data Management (TDM)*, mainly architected for file-based data repositories of teams of 20–30 individuals. Those environments have been extended to *PDM (Product Data Management)* and *PLM (Product Lifecycle Management)* environments, which offer globally available full configuration and structure rich repositories and exchange platforms. Nevertheless, still today, there are different environments for the "physical" model world of hardware and for line of code environments for software.

As the *domain layer* at the top of Fig. 4.4 shows, it is essential to differentiate between the three major technical domain areas mechanics (hydraulics, pneumatics), electric/electronics and (embedded) software. For each one of those major domains different sets of modeling and development environments and tools have been developed.

The mechanics VPC portfolio offers development tools such as *Computer Aided Design* (CAD; design of components and assemblies), *(3D) Computer Aided Engineering* (CAE; algorithms to analyze shapes and behaviors with the help of discrete volumetric or surface elements and components) tools such as *FEA* (Finite Element Analysis) and *MBS* (Multi Body Simulation) but also simplified mathematical equation based *1D CAE* approaches. From 2010 onwards, official tools for requirements management (RM) and requirements engineering (RE) have been also added to the mechanics-oriented development world (before it was mainly used in software development). *DMU (Digital Mock-Up)* provides a digital environment which can visualize and analyze dynamically complex products in real time manner.

The next level up from DMU is called FMU (Functional Mock-up). There currently exist two basic understandings of the FMU: the first one concentrates on the upfront representation of all pure (neutral) functions and their interplay, whereas the second one does encompass all logical and behavioral characteristics, i.e. not just the mechanical (hydraulic/pneumatic) ones. The later one obviously requires the descriptions and behavior models of the electrical/electronic domain and of the software control domain in addition to the mechanical (hydraulic, pneumatic) one.

The entire mechanic VPC portfolio has its foundation on the mechanical physics models based on masses, forces, inertia, momentums and torques, displacements, velocities and accelerations, both in static and dynamic circumstances.

The electric/electronics VPC portfolio offers development tools and environment such as E-CAD and PCB (Printed Circuit Boards) layouts and simulations (e.g. schematics of electric circuits and networks). In addition, there exist Design Rule Checking (DRC) tools for determining the best parameter set-up for semiconductor manufacturing and complex HIL (Hardware in the Loop) test environments in order to provide test beds for embedded software testing against a full set of the functional behavior of the overall technical system under development. The physical parameters and models of current, voltage, magnetic flux and associated electronic functions and networks represent the basis of this VPC portfolio.

The relatively young discipline of Software Engineering as part of Computer Science made its way into the Virtual Product Creation portfolio from the 2000s onwards as part of the sharply increased embedded software control of modern mechatronic products. Here the tools suite embraces a range of different capabilities in order to develop robustly and test software code and to verify its function within the bigger context of the technical product/system. *Integrated Development Environments* (IDE) as well as *CASE* (Computer Aided Software Engineering) provide connected engineering tools for code editing, code testing, software compilation and linkage and build automation as well as error reporting and explanation. CASE also provides orientation for software development approaches such as waterfall or scrum development. It also provides options for requirements management for information system and software development as well use case specifications, e.g. in the context of *UML* (Unified Modeling Language). Some software companies, such as IBM, have established specific environments such as *RUP* (Rational Unified Process) in order to provide adaptable process frameworks of the entire software development innovation and development operation and testing. Finally, there exists a range of *SIL* (Software in the Loop) engineering and test environments, describing a test methodology where executable code such as algorithms (or even an entire controller strategy), usually written for a particular mechatronic system, can be tested within a modeling environment that can help prove or test the software.

Figure 4.5 illustrates the Virtual Product Creation capabilities within the context of digital manufacturing. Principally, there exists a similar technology world as in

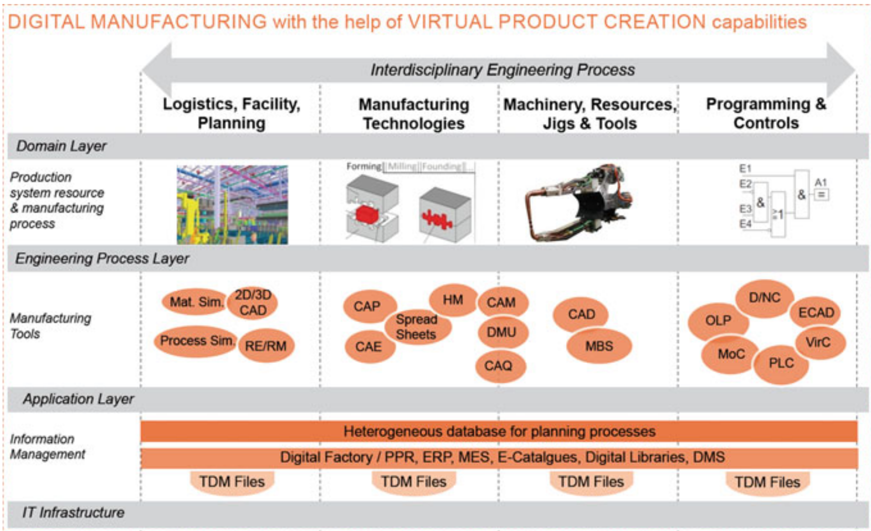


Fig. 4.5 Layer view of Virtual Product Creation capabilities for digital manufacturing

the technical system development arena; however, the viewpoint in manufacturing is different. Typically, the following four areas of expertise prevail with respect to Virtual Product Creation for digital manufacturing:

- logistics, facilities, overall "digital factory and process" planning;
- simulation of specific manufacturing technologies and physics;
- modeling, simulation and management of machinery, resources, jigs and fixtures;
- programming of robots and machines, control engineering.

In all these VPC disciplines, a robust data management regime needs to be orchestrated through the application of specific data management solutions like TDM (Team Data Manager) as well as within overall PDM (Product Data Management) environments. Different kinds of underlying data bases (files based, client server based) exist in order to safely store and provide data and information of the final product itself, the manufacturing process steps and related technical details as well as the wide variety of resources (tools, fixtures, jigs, machines, workers etc.).

As depicted in Fig. 4.5, the *Virtual Product Creation* capabilities *for logistics, facilities and overall production systems and lines* provide a range of mathematical and digital process planning and simulation techniques. They are necessary in order to describe, simulate and analyze the interaction of the manufacturing objects within the *Digital Manufacturing (Factory)* set-up, i.e. the digital representation of the real production environment in a factory.

The physical behavior prediction of material cutting, flow and forming as part of specific manufacturing technologies such as milling, late work, casting, forming, deep drawing, drilling, welding, brazing, coating etc. does require computer aided engineering tools (FEA, MBS etc.) and methods. Interactions with the machines themselves do incorporate more and more *HMI (Human Machine Interface)* simulations. Overall configurations are oftentimes stored in type of spreadsheets linking all critical machine process parameters.

Automation of discrete manufacturing cells with robots and jigs require a solid simulation of dynamic forces, momentum and torques as well as efficient motion control, which is achieved by traditional modeling and simulation packages such as CAD (Computer Aided Design) and MBS (Multi Body Simulation). Automation within an overall manufacturing cell or production line, however, needs software and hardware control. Hence, a wide range of *Virtual Product Creation solutions* is offered for such tasks: *DNC (Distributed Numerical Control), PLC (Programmable Logic Control)* and *OLP (Offline Programming)* make it possible to create, simulate and analyze/test upfront—as part of ViC (Virtual Commissioning) within the *Digital Factory* environment—and to finally provide control programs for all numerically controlled machines and robots within the real factory.

As part of manufacturing engineering, the VPC capability *MoC (Management of Change)* is key in order to manage all engineering changes during the development of the product, the manufacturing process and all related manufacturing resources. Later on during real production, the *MOC (Manufacturing Operations Center)* is critical in order to control ongoing job batches as part of *MES (Manufacturing Execution Systems).*

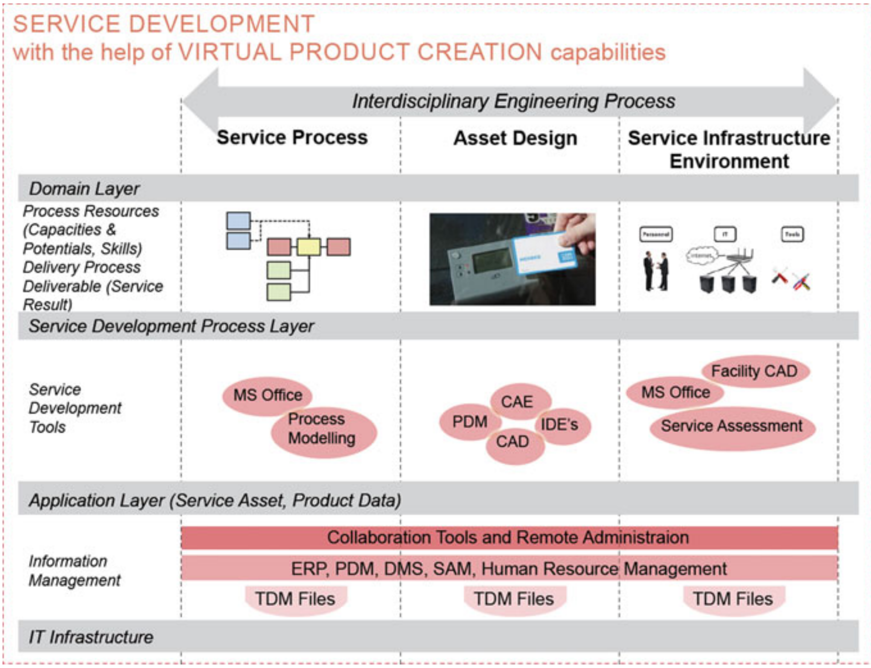


Fig. 4.6 Layer view of Virtual Product Creation capabilities for service development

Finally, Virtual Product Creation capabilities should be explained in the context of *service development* (see Fig. 4.6). From 2010 onwards as part of the increased importance of maintenance and customer service as well as of product-service systems (PSS) *Virtual Product Creation capabilities* are increasingly requested within the context of the *service process design* itself and in close conjunction with the related *asset design* (i.e. products, machines, gadgets etc.) and the service infrastructure environment (compare Fig. 4.6).

One recognizes the use of similar core VPC capabilities also here in the layer of service development tools but within a different context. The usage of modeling paradigms and tools such as CAD (and in some advanced research approaches even already VR) is still in the beginning since service process and its interaction with products, infrastructure elements and processes still miss a substantial library of relevant objects, their notations and the common engineering semantic.

However, it is worth noting that service development is highly dependent on the availability of information from both, PDM and ERP systems, as well as from specialized environments such as *SAM (Software Asset Management)* and Human Resource Management.

All of the above introduced Virtual Product Creation activities and capabilities will be explained in detail in Chaps. 7 through 16 (all major technologies).

Virtual Product Creation of today and its linkage to the different phases of Product Lifecycle Management keep a clear distinction, however, to other digital enterprise capabilities. Examples of other digital enterprise capabilities are represented by the event-based order to delivery management world in the factory, where the following IT solutions are widely used:

- ERP (Enterprise Resource Management), with its integrated Product Planning System (PPS) capabilities and
- MES (Manufacturing Execution Systems) together with other shop floor IT systems.

These technologies are not subject of this book but should be mentioned as an important boundary condition for *Virtual Product Creation* concentrating on the operational side of the factory business.

The concept and the solution sets of the Digital Twin, however, might close the separation of lifecycle and event based digital business in the future! (See more details in Chaps. 20 and 21).

References

1. Spur G, Krause FL (1997) Das virtuelle Produkt: management der CAD-Technik (The virtual product: management of the CAD Technology). Fachbuchverlag Leipzig. ISBN-13: 978-3446191761
2. Anderl R, Eigner R, Sendler U, Stark R, Smart Engineering (2012) Interdisziplinäre Produktentstehung (Interdisciplinary product creation). Acatech, Springer Vieweg. ISBN 978-3-642-20371-9
3. Saaksvuori A, Immonen A (2008) Product lifecycle management. Springer, Berlin. ISBN 978-3-540-78173-8
4. Stark J (2005) Product lifecycle management—21st century paradigm for product realization. Springer, Berlin. ISBN 1-85233-810-5
5. Eigner M, Stelzer R (2008) Product lifecycle management—Ein Leitfaden für product development und life cycle management (A guideline for product development and lifecycle management). Springer, Berlin. ISBN 978-3-540-44373-5