Functional modeling: Representation of dynamic aspects in function structures

o. Prof. Dr.-Ing. Dr. h.c. H. Grabowski
Dr.-Ing. Ralf Lossack
Dipl.-Inform. Harald Kunze

University of Karlsruhe, Germany
Institute for Applied Computer Science in Mechanical Engineering (RPK)

Abstract

The following article discusses dynamic aspects in function structures as a result of the simulated product behavior in time. In this case product behavior is separated into different product states. We show how the the product behavior can be *simulated* within a single state and how views on function structures are changing by considering multiple states. In addition to that, solution finding processes of effective solutions as part of conceptual design can be supported for innovative and creative design.

Introduction

Modeling of product functions is an essential part of purposeful and methodological design. In this stage of the design process the function structure of a future product is determined at an abstract level. Sub functions specified during the functional modeling stage form the suppositions for the following design stages such as principle modeling and shape design (Grabowski, Lossack, & C. 1995).

To model the functional behavior of a product it is not sufficient to regard only its *static* structure, i.e. the function structure of a single state. Rather the state dependent views on a function structure have to be examined and developed. The aim is that on one hand the product behavior can be simulated, analyzed and documented on a very early logical layer. On the other hand the *specification of temporal restrictions* during functional modeling enables a computer based selection of physical principles with much more precision. Both methods increase the effectiveness and precision of the specification and development of new products or product components.

The concepts presented in this paper contribute to the development of functional design and by doing so providing a good basis for the subsequent design stages especially the stage of physical principle modeling.

We organized the paper into the following sections, first describing the concept and realization of a typical static simulation of function structures, which allows to simulate product behavior and product analysis within a single state. Afterwards we discuss concepts of physical principle modeling and present an extension to adapt it to our needs. Finally our findings of state dependent views on function structures and the support for analysis of effect chains as part of the stage of physical principle modeling is illustrated.

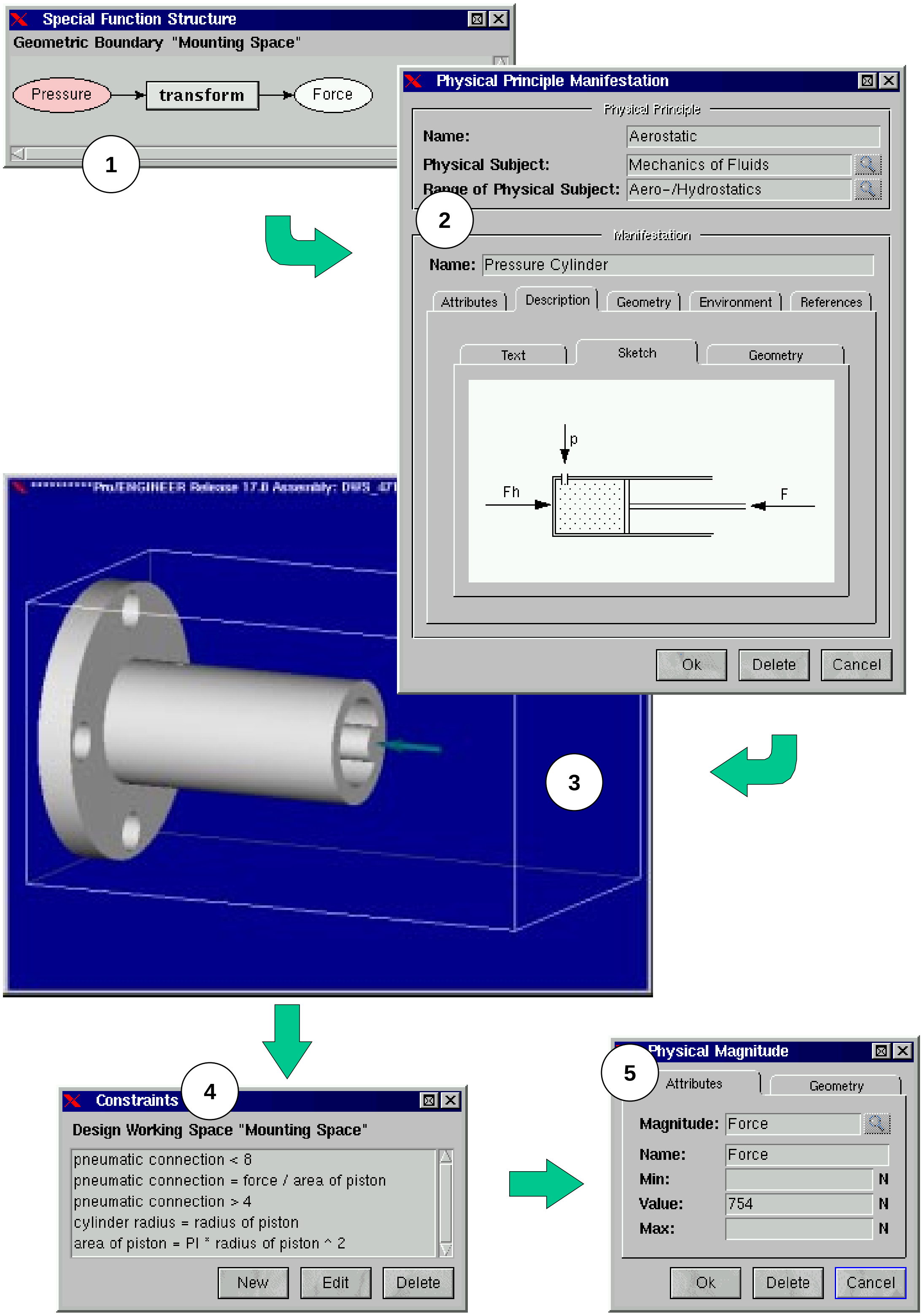
Static simulation on the basis of function structures

A function is known as a general and wanted connection between input and output quantities of a system with the aim to accomplish a certain task (Pahl & Beitz 1988). In a step of concretization functions are attributed to physical principles (Roth 1994),(Koller 1994). With the help of this information and the already known customer requirements the product has to meet, a *static simulation* can be performed.

During static simulation a single state of a system is considered. Therefore the input and output quantities of function structures are specified by discrete values. The mentioned physical principles (or effects) as part of the subsequent design stage are connected to the functions by a mathematical relation between inputs and outputs of an effect. In the stage of physical principle modeling this mathematical relation is called a physical law and can be computed or verified. If the values of the input as well as the output quantities are known, the validity of these values can be *verified*. If inconsistencies occur, i.e. the mathematical equation known from the physical law is not solvable with the specified values of input and output quantities, the user is notified and asked to intervene. If otherwise only the value of one input or output quantity is known, the corresponding unknown value can be *computed*. The computation or verification of values is done step by step with each single function of the function structure. By this means the values of the input quantities can be *propagated* through so called effect chains to the output quantities. An effect chain is a network of physical principles (effects) connected with each other by physical laws. The performing program module which accomplishes the task of computation and verification is called a *constraint solver* (Grabowski *et al.* 1998). In addition to that the constraint solver indicates inconsistencies to the user.

In general the physical laws, which are connected to the physical effects, specify the mathematical connection between input and output quantities through further *parameters*. The parameters mostly result from the geometric construction of the physical effect, the physical principle. E.g. the physical law of the “pneumatic effect” $p = F/A$ (pressure is force per area) contains the geometric parameter area

From: Proceedings of the Twelfth International FLAIRS Conference. Copyright ' 1999, AAAI (www.aaai.org). All rights reserved.



Geometric Boundary "Mounting Space"

Name: Aerostatic
Physical Subject: Mechanics of Fluids
Range of Physical Subject: Aero-/Hydrostatics

Name: Pressure Cylinder

Design Working Space "Mounting Space"

pneumatic connection < 8
pneumatic connection = force / area of piston
pneumatic connection > 4
cylinder radius = radius of piston
area of piston = PI * radius of piston ^ 2

Magnitude: Force
Name: Force
Min: N
Value: 754 N
Max: N

Figure 1: Example for the support of design steps through static simulation

A. With the principle solution "pressure cylinder" this area corresponds to the area of a piston. This indicates that the functionality of the constraint solver must be distinguished. The constraint solver is only able to verify the function with its connected effect, if the input and output quantities and all further parameters included in the physical law of the effect are specified by a value. If a single value is unknown, this value can be computed by the constraint solver. If more values are unknown, this results in a *solution space* of valid values.

With the occurance of geometric parameters in physical laws a static simulation is only feasible with a strong coupling of early design stages from requirement modeling to embodiment. Thereby the constraint solver can "recognize" whether a parameter is already specified by a requirement, respectively the value of geometric parameters can be read from the shape model. If the constraint solver computes the value of a geometric parameter, this value is written to the shape model again. In that way an "automatic" shape design can be obtained for known values of input and output quantities.

An example will illustrate these facts. A designer is asked to develop a pneumatic robot gripper. From a requirement list the interface conditions *pneumatic connection* $4bar \leq p \leq 8bar$ and *holding force* $1152N \leq F \leq 2000N$ are known. In the following design step the designer defines a special function[1] which transforms the input pressure into a force (Figure 1(1)). A solution finding process is performed on this function with the result of the physical principle *pressure cylinder* (Figure 1(2)). To support the following embodiment of a physical principle so called *effective geometries* are described in the solution base. In general the effective geometry describes *where* a physical phenomenon takes place and is defined by its *effective surfaces*. When the pressure cylinder is inserted in the current solution space, its corresponding effective geometry is also instantiated in the product model. With this we also have the advantage of representing the corresponding parameters of the effective geometry elements *cylinder* and *piston*, the explicit constraints *radius of cylinder* $=$ *radius of piston* and *area of piston* $= \pi \cdot$ (*radius of piston*)^2 in the product model. After that the effective geometry has to be adapted interactively to the environmental conditions by reading the parameter *radius of piston* from the geometry. Hereby the the value $r = 20mm$ is automatically fixed in the model and the constraint solver computes the *area of piston* to $1256.6mm^2$ and determines the outgoing force to $F = 754N$ by the physical law $p = F/A$ of the pressure cylinder (Figure 1(3-5)).

Reversion and inversion of physical principles

During the solution finding process special functions become concretized to physical principles. To achieve this mapping physical principles in the solution base are described by their function and physical law[2]. With the help of this process physical principles can be found which might be used as a specialization of the function. The problem is that physical principles can accomplish various functions, the mapping process is not a one-to-one mapping. Depending on the view these functions can be interpreted as useful or disturbing functions.

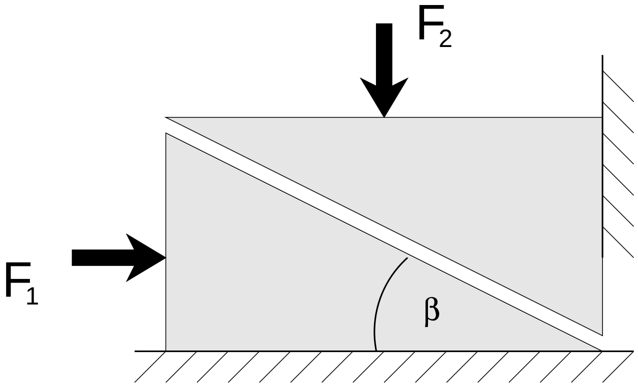


Figure 2: Principle sketch of the wedge effect

The functions which are accomplished by a principle can be understood as "directed" because they transform an input quantity with the functional connection (the physical law) to the output quantity. The question arises whether the according principle accomplishes the "reversed" function, i.e. the transformation of the output quantity to the input quantity. Three cases can be distinguished: *not reversible*, *completely reversible* and *conditionally reversible*. A completely reversible effect is for example the lever effect which increases respectively decreases an input quantity *force* to an output quantity *force* and vice versa. This is not possible with all effects. For example we have a look at the wedge effect in more detail. The wedge effect accomplishes the function "scale force into force" with the according physical law $F_2 = F_1 \cdot tan(\beta)$, the input quantity *force* F_1 and the output quantity *force* F_2 (cf. Figure 2). However whether this principle can be used reversely depends on the gradient angle β. If this angle was designed too flat (approximately $< 20^o$) the principle can not be reversed. We call this, the wedge effect is *conditionally reversible*.

Another usage of physical principles arises by their *inverse* usage. In this case the direction of the special function is the same, i.e. the input quantity becomes transformed to the output quantity. The difference is that the value of the input quantity is negated. For example the given input quantity F_1 of the wedge effect (Figure 2) is not a pressing force but a tractive force. It can be seen easily that the wedge effect *cannot* be used inverse, because a tractive force at the input does not result in a tractive force at the output.

With the knowledge whether a principle can be used re-

---

[1] A special function is a function which describes the product behavior by a verb and a set of inputs and outputs (Lossack, Umeda, & Tomiyama 1998).

[2] We call the inherent connection between a supposition and its solution a solution pattern. In this case a pattern for a physical solution is established by a function as supposition and a physical law as solution (Grabowski, Lossack, & C. 1995).

versely and/or inversely an analysis of effect chains is possible. But first we have a closer look at state dependent views on function structures.

State dependent views on function structures

All functions of a product can be established by a systematic investigation of the product life cycle stages[3] (Figure 3). Therefore during the design it has to be considered that for multiple usage of a product both deployment functions for the actual purpose of the product and reset function have to be established. Production and marketing functions are further developed during the design process together with later stages, such as scheduling and tooling. Recycling functions are established during the design process (Grabowski, Rude, & Langlotz 1994),(Grabowski, Rude, & Langlotz 1996).

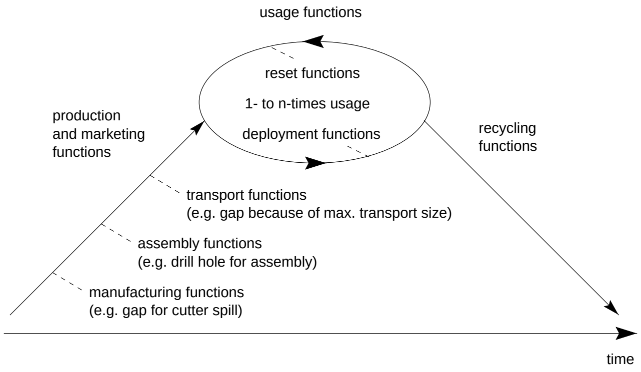


Figure 3: Product life cycle approach with using function concepts

Usage functions specify the behavior and in this sense states and state transitions of a product. On the functional level the state of a product is determined by its output quantities. That means that the change of output quantities in time can be assigned to states and state transitions. In Figure 4 this relation can be seen with the help of an example, a robot gripper. While investigating the product functions of the robot gripper the states `released` and `hold` can be derived immediately. The state transition from the state `released` to the state `hold` is described as `grip` (deployment function). The opposite state transition is `release` (reset function). Having modeled states and state transitions the ideal change in time of the output quantity *holding force* can be specified.

The output quantity is determined by a mathematical function processing the input into the output quantity. Therefore having knowledge about the output quantity (i.e. change of behavior in time) properties and features of the input can be derived. Within the robot gripper example statements about the pneumatic interfaces can be made, such as: the state `released` can be reached with pressure and the state `hold` without pressure. This is also true for the reverse order whereas the function has to be adapted appropriately.

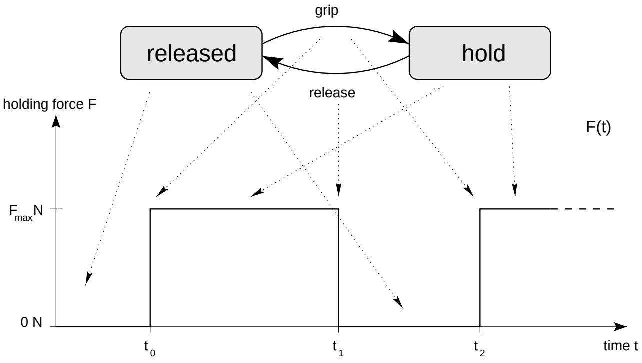


Figure 4: Relation between states, state transitions and the graph of the output quantity

The functions (function structures) established to meet a desired product behavior are always related to a certain state respectively state transition. The upper part of Figure 5 shows the function structure of the robot gripper example in the state transition grip respectively the state hold. If now the designer models the state transition *release*, then he/she has a different *view* of the function structure *grip*. In case of the function *grip* especially the generation and transformation of a force is of importance. On the other side in the state transition *release* there is no force at all because there is no pressure either. Therefore the reset function *release* does not have to counteract this force.

On the other side the jaws cover a certain *distance* which has to be reseted by the function *release*. Therefore in the function structure *grip* from the viewpoint of the function *release* the generation and transformation of a *distance* is of special interest. And that's the reason why the function structure *grip* has a *distance* as output in the view of the function *release*.

The function *release* will reverse the *distance* the jaws have covered. From this it follows that the output quantity of the function *release* is a *distance*. To generate a distance energy is necessary. The state transition *release* however is performed by no pressure and therefore no energy. It appears from this that energy has to be stored so that a distance can be generated by the function *release*. In the next steps designers can search for physical principles storing energy for the function *transform energy into distance* and adapt this principle in the current solution space. E.g. a mechanical spring between the jaws can be used to open the jaws when there is no pressure.

Analysis of effect chains

According to the different function flows of the special function structure which result from the different states of a prod-

---

[3] This notion of a function is different from the concept we have used so far. This definition of a function is often refered to as the verb-noun approach, because the product behavior is described by a noun and its transforming verb. We call this type a *product function (Lossack, Umeda, & Tomiyama 1998)*, whereas the function flow described by its inputs and outputs is defined by a *special function* (Figure 1(1))

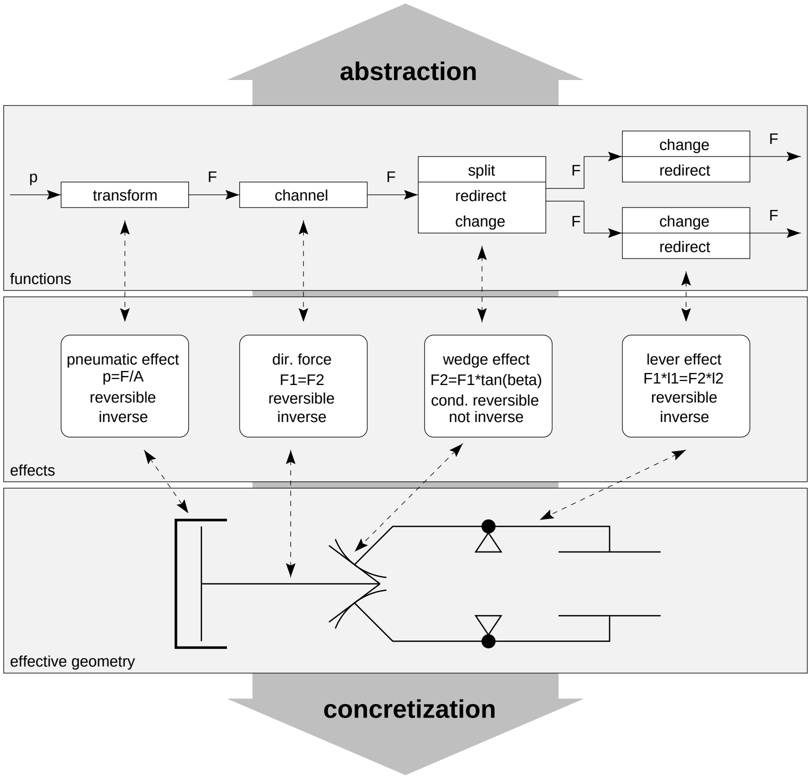


Figure 5: Function structure of a pneumatic robot gripper with denoted physical principles and effective geometry

uct the effect chain might be traversed in different directions. With the knowledge whether a principle might be used reversed and/or inversed an *analysis of effect chains* is possible. This will be illustrated with the function structure and related physical principles of the pneumatic robot gripper.

Figure 5 shows the special function structure of the pneumatic robot gripper in the state `hold`, the concretization of the functions to effects and the concretization of the effects to effective geometry. With a static simulation the resulting force at the output can be computed from the specified value for the pressure at the input. In this case the effects are used along the function flow of the state `hold`. An analysis of the effect chain can be done by regarding the reversed function flow as it occurs in the state transition *release* when releasing the gripper jaws. In this state the compression spring between the gripper jaws puts a force on the gripper jaws. The lever effect, which represents the effect of the gripper jaws, is reversible. So there is no problem with the lever effect when reversing the function flow. In the following step of the reversed effect chain the lever puts a force on the wedge. But the wedge effect is *conditionally reversible* only. Depending on the layout of the wedge and its gradient angle problems may arise (cf. Figure 2). If the gradient angle is too flat the wedge effect is not reversible. In this case the designer is asked by the system to increase the gradient angle of the wedge enough that the wedge is reversible again. He also might add an additional function to reset the wedge. This can be done with an additional spring at the pressure cylinder which resets the piston.

Conclusion

With the key concepts of state and state transitions we could model the behavior of a product in terms of function modeling with more precision and by doing so supporting the solution finding process of subsequent stages of the conceptual design stage. Especially when solving problems in the stage of modeling physical principles we found advantages by using these concepts.

Another important point is that we can contribute to the integration of two main concepts of function modeling; the verb-noun approach and the function flow approach[4]. In this article the product states represent the verb-noun approach and from this the function flow approach can be derived very easily. The main benefit of this approach can be seen in supporting effectively solution finding processes.

References

Chakrabarti, A., and Blessing, L. 1996. Representing functionality in design. In *AIEDAM, Artificial Intelligence for Engineering Design, Analysis and Manufacuring*, volume 10(4). Cambridge, England: Cambridge University Press.

---

[4] You can find interesting discussions on that topic in (Chakrabarti & Blessing 1996)

Grabowski, H.; Rude, S.; Maier, M.; and Michelis, A. 1998. Integrated Product and Process Modelling as a Basis for the Realization of modern PDT-Concepts. In *ProSTEP Science Days*. Darmstadt (Germany): ProSTEP Verein zur Förderung internationaler Produktnormen e.V.

Grabowski, H.; Lossack, R.-S.; and C., W. 1995. Supporting the Design Process by an Integrated Knowledge Based System. In *Formal Design Methods for CAD*. IFIP WG 5.2.

Grabowski, H.; Rude, S.; and Langlotz, G. 1994. Informationsspeicher zur Unterstützung der Lösungsfindung im Konstruktionsprozeß. In *VDI-Bericht Nr. 1148, Tagungsband zur Systec'94*.

Grabowski, H.; Rude, S.; and Langlotz, G. 1996. Die Funktionsmodellierung auf dem Weg zur Wissensverarbeitung in CAD-Systemen. In *CAD'96 Verteilte und Intelligente Systeme*.

Koller, R. 1994. *Konstruktionslehre für den Maschinenbau, Grundlagen zur Neu- und Weiterentwicklung technischer Produkte*. Berlin: Springer-Verlag, 3. edition.

Lossack, R.; Umeda, Y.; and Tomiyama, T. 1998. Requirement, Function and Physical Principle Modelling as a Basis for a Model of Synthesis. In *Computer Aided Conceptual Design*. Lancaster University Engineering Design Centre.

Pahl, G., and Beitz, W. 1988. *Engineering Design: A Systematic Approach*. Berlin: Springer-Verlag.

Roth, K. 1994. *Konstruktionslehre*, volume 1 of *Konstruieren mit Konstruktionskatalogen*. Berlin: Springer-Verlag, 2. edition.