

7

Putting the Framework to Operation: Enabling Organizational Development Through Learning

This chapter picks up the framework developed in Chap. 6 and shows how it can be put to operation using instruments that have been successfully deployed in practice (cf. Chap. 8). These instruments enable articulation and alignment of work process knowledge, allow its representation and transfer within organizations, and facilitate acting on these representations for validation and implementation in diverse organizational settings. We here adopt the organizational learning perspective, already proposed in Chap. 1, and situate the presented instruments along a multi-perspective learning chain informed by the components of the framework presented in Chap. 6. This allows us to offer an integrated view in Sect. 7.5, which shows how the framework can be instantiated in organizational practice.

Organizational learning comprises learning on two layers:

1. Individual level
2. Collective level within an organization (e.g., CoP) or beyond (e.g., company network)

According to the Knowledge Lifecycle (KLC) (Firestone and McElroy 2005), it also can occur within running business operation and beyond. However, learning outputs need to become concrete and visible, in particular when implemented in the business operation. Single loops target towards an envisioned optimum of operational procedures, for example, zero-tolerance with respect to quality of products. Learning is thus related to business processes and their management. Learning is less concrete when questioning assumptions and background of business operations. Double loops allow claiming and evaluating change proposals that could affect business operation. Deutero-learning reflects double-loop learning. It structures reflection on operation and learning procedures (knowledge processing), keeping multiple levels of development activities apart with dedicated point of coupling, triggering redesign of processes (Table 7.1).

With respect to handling knowledge according to an organizational learning architecture (see earlier in the chapter), reflection, referring to as-the-organization-is, and some prospective organization of work need to be supported and may occur intertwined. Hence, stakeholders need to be able to

- *Express* themselves in terms what they know, in order to document starting points of change
- *Reflect* on articulated knowledge, either alone, with peers, or other groups
- *Represent* and *manipulate* codified knowledge, forming baselines for further steps
- *Store* to avoid loss of information and process know-how ('again-invented-here')
- *Process* knowledge to evaluate or establish adjunct or resulting operational procedures
- *Share* knowledge by distributing content to put it to operation

Summing up, the development of support (chains of) technologies or enablers is a multi-dimensional endeavor, as it needs to address

- individual/group learning
- single/double/deutero-loop learning

Table 7.1 Learning/design dimensions, activities, and tools

Levels, loops, and enablers		Learning loops: single/double/deutero		Enablers: cognitive/social/emotional	
Learning chain elements	Individual layer	Collective layer	Constituents (what it is about)	Learning loops: single/double/deutero	Enablers: cognitive/social/emotional
	Features	Features			
Articulation and elicitation	Express (model), prepare for sharing, Reflect on shared knowledge	Active sharing (express and negotiate)	Knowledge about operation (processes), content/assumptions, claims, change proposals/reflected learning processes	(m = method, t = tool) m also includes tool-specific feature set Concept mapping (m), VNA (m), S-BPM modeling (m), structure elaboration (m), (rural) Comprehend (t), Faraw (ARCOR) (m)/social media (forum, chat, blog, Facebook, Twitter, LinkedIn) (m + t)/social media (m + t)	
Representation	Navigate, search, store, trace, Versioning	Navigate, search, store, trace, Versioning	Business process models (business processing)/Knowledge (change claims (knowledge processing)	Filter (m), topic map (m), annotations (m), Nymphaea-content management, social media support and intertwining (t), Metasonic Build (t)/annotations (m), Nymphaea view management (t)/annotations (m), Nymphaea view management (t)	
Manipulation	Search, retrieve, develop perspective Edit content, prepare sharing perspective Explore	Search, retrieve, develop group perspective, edit content, perspective management,	Perspective on constituents	Tagging (m), tracing (history) (m), editor (t), Comprehend(t)/like, follow, exchange annotations (m) social media (t), Comprehand (t)/like, media (t)	
Processing		Explore impact on organization level	Evaluated knowledge claim	Validation and processing of BP (m) BP-execution engine (t), Simulating BP (m) Simulation engine (t)	

- cognitive (content, domain)/social processes
- knowledge elicitation, representation, visualization, presentation and communication, sharing, processing

Support technologies require respective features, may stem from different kinds of applications:

- articulation, elicitation, elaboration/modeling tools
- content management systems
- social media
- business process management suites
- CSCW systems

Support technologies are of different kinds, with respect to mobility and chaining:

- Stationary/mobile
- Articulation—learning—processing

Support for explication, exploration, and distribution should capture both, scenarios for individual and collaborate development and deployment:

- In-situ scaffolding during operative work processes (what can I do now/next)
 - based on previous own task implementations
 - based on others' task implementations
- Learning from other actors who previously took the same role in a task bundle or are experts in the area during operative work processes or in knowledge integration (whom can I ask)
- Reflection on previous own and others' task implementations as a part of problem claim identification and knowledge production (what have we done)

- individually
- as a group
- Aligning with other actors who take other roles in a task bundles during knowledge integration (how can we work)

These requirements can be methodologically and technically approached using instruments for self-directed learning. In the following, we revisit the set of tools described throughout the former chapters and put them into mutual context using the framework developed above.

7.1 Sample Actor-Centric Tool Support for Articulation and Elicitation

The aim of the articulation and elicitation phase is to methodologically and technically allow for modeling of organizational phenomena whenever the need arises, especially of that directly situated in the actual work context. The approach exemplified here follows a physical card placement paradigm, for example, as proposed for structure elaboration techniques in the field of mental model externalization and alignment (Dann 1992) and, for example, adopted in the field of BPM by Luebbe and Weske (2011). The semantics of the used cards as well as their spatial arrangement is predetermined to be interpretable as actor-centric, communication-oriented business processes.

A modeling approach, which serves the goal of capturing a comprehensive representation of the overall business process, needs to take into account all individual contributions and facilitate identifying and making visible different mental models of how the collaborative aspects are performed (Fischer and Mandl 2005). In order to be able to identify different perceptions of how collaborative work is carried out, the individual mental models of the collaborating contributors need to be made accessible for alignment (Engelmann and Hesse 2010). Consequently, an approach for collaborative modeling of work should profit from a stage during which the participants individually externalize their mental model of the business process in the form of a conceptual model.

The results of these individual modeling activities provide a foundation for argumentative co-construction of a shared understanding about the business process. Argumentative co-construction can again be facilitated by conceptual models that serve as a shared artifact (Fischer and Mandl 2005). A conceptual modeling approach supporting this process should allow expressing individual claims and collaboratively putting them in the context of other claims for referral in the argumentative chain.

7.1.1 Comprehend Cards

Comprehend Cards (Oppl 2017; Oppl et al. 2017) enable creating models of work without the need for any dedicated technical infrastructure. The aim of this component is to allow for collaborative modeling whenever the need arises, especially of that which is directly situated in the actual work context. The Comprehend Cards approach (cf. Fig. 7.1) follows a physical card placement paradigm, for example, as proposed for structure elaboration techniques in the field of mental model externalization and alignment (Dann 1992). The semantics of the different card types is

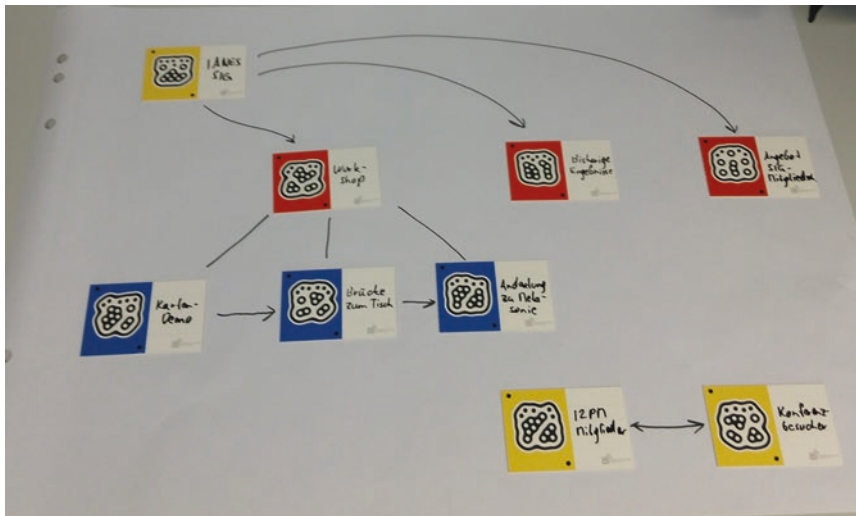


Fig. 7.1 Sample model created with modeling cards

determined by the use case they are deployed in—the system can be used for semantically open modeling (e.g., for concept mapping) as well as procedural modeling adhering fixed semantics as described in Sect. 5.1. What sets apart Comprehand Cards from other physical card modeling approaches (e.g., Decker and Weske 2009) is an additional software component, which allows extracting the conceptual model from any picture taken of the card-based model. The model extraction algorithms recognize concepts (i.e., cards), concept types (i.e., card types), and relationships between concepts (i.e., connections drawn between cards). Labels written on cards or besides connections are also extracted and provided as scaled and rectified images.

The recognition engine is designed to be used with pictures taken by smartphone cameras without any strict constraints on image angles and lighting conditions (Oppl et al. 2017). Pictures of a model are uploaded to an online platform that acts as a front-end for model extraction. If a model is too large to be depicted on one image in sufficient detail, multiple pictures covering distinct but overlapping areas of the model can be uploaded, which are then automatically processed by the system to improve recognition quality. For recognition of the cards, an adapted version of ReacTIVision (Kaltenbrunner and Bencina 2007) is used. All cards thus bear optical markers that make them uniquely identifiable by the system. Connections are traced using image recognition algorithms adapted from the study by Jiang et al. (2011). The extracted model information is represented in a configurable XML-based format that allows for further processing of the model in any compatible tool (cf. next sections).

7.1.2 Comprehand Table

The *Comprehand modeling table* (Oppl 2006; Oppl and Stary 2009; Oppl and Rothschädl 2014) is an interactive collaborative modeling environment that used graspable modeling elements to support externalization activities and to allow for equal access to the model for multiple modelers at the same time. The Comprehand Table aims at supporting in-depth, potentially controversial, modeling situations, where the need for flexibly

altering the model conflicts with the constraints of card-based models and their hand-drawn, hardly changeable connections. The table (cf. Fig. 7.2) thus focuses on features supporting the modeling process, such as easy connection creation and deletion, altering the model layout without changing its conceptual structure, and tracking the modeling history, thus allowing to revert the modeling process to earlier stages. A detailed description of the features and their mode of operation is provided in Oppl and Stary (2014).

Technologically, the modeling table is an implementation of a tangible tabletop interface (Ishii and Ullmer 1997) that uses back-projection on the table surface to blend the physical modeling elements with digital information. Tangible interfaces (Ishii and Ullmer 1997) are an approach

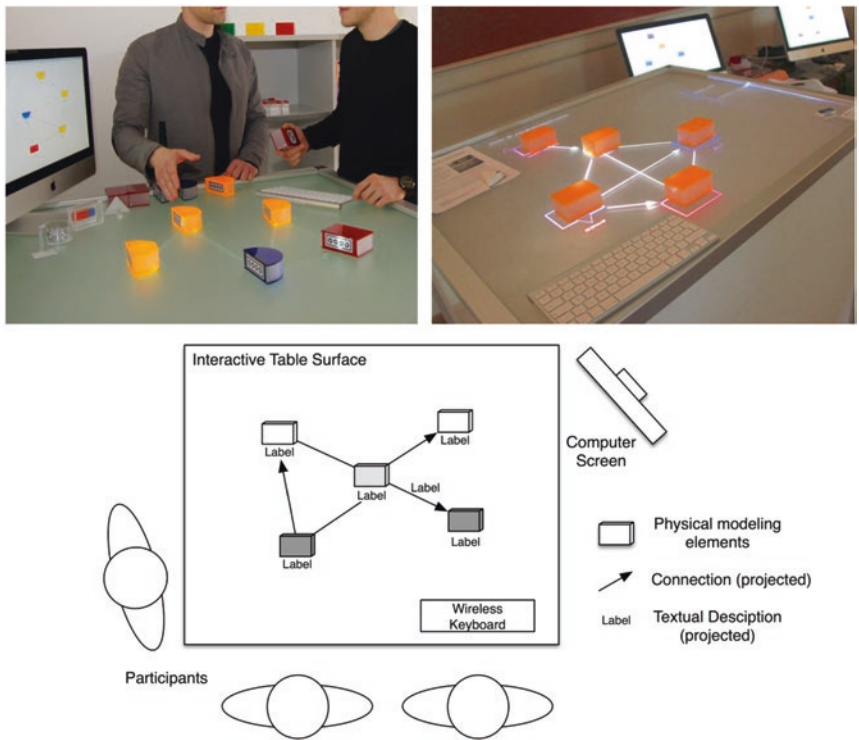


Fig. 7.2 Comprehend Table overview (top-left: interaction on table surface; top-right: modeling tokens with projected connections; bottom: schematic bird's eye view of tabletop)

to HCI that aims at bridging the gap between artifacts in the physical world and digital information conceptually belonging to those artifacts. Information augments the physical artifacts, is accessible through them, and can also be manipulated directly using the artifacts. These properties make tangible interfaces a candidate for integration of the superficially opposing requirements of physical immediacy and computer support during externalization of mental models.

Tangible interfaces are not only favorable from an externalization point of view. Previous research has shown that tangible interfaces also facilitate cooperation (Hornecker 2001) and learning (Resnick et al. 1998; Zuckerman et al. 2005). Hornecker (2001) examines the effects of tangible interfaces on human cooperation (with a focus on tabletop interface, which are built upon a common physical surface used for interaction with the system). Based on a review of existing literature and validated empirically, she identifies four social effects of tangible interfaces that facilitate cooperation: they act as enabler for (1) intuitive and simultaneous manipulation, (2) focusing, (3) awareness of gestures and performatives of actions, and (4) are facilitator for externalization and role as boundary object. Regarding effect 1, Hornecker (2004) claims that tangible interfaces lower the barrier for initial usage and allow for performing the actual task of coordination instead of investing effort in handling the tool. Tangible interfaces are also claimed to facilitate focusing on the topic of coordination for all involved individuals (effect 2) by making the physical representation a spatially focal point of interaction and in this way, creating a transactional space. The co-located setting around the interface also facilitates communication of non-verbal signals and performatives of actions of other individuals (effect 3). The physical representation also enriches communication beyond verbal expression, for example, by allowing gestural referencing of aspects of the shared information. Finally, the physically shared representation acts as a boundary object, providing an anchor for the development of shared understanding among the involved individuals (effect 4).

Comprehand has been implemented using an interactive table and uses physical tokens for cooperative structure elaboration. The semi-transparent table surface is back-projected from below to display additional information like connections or captions (cf. Fig. 7.2, bottom

image). Labels of modeling elements as well as connections between them are part of the projected information, which allows for easy rearrangement of models. The recognition of modeling elements and manipulation tools (such as a connection removal tool) is again based upon the ReacTIVision system (Kaltenbrunner and Bencina 2007), which makes the Comprehand Table compatible to the Comprehand Cards system in terms of recognizable model elements. While the cards could be used on the table, the standard configuration uses 3D modeling blocks, which are graspable more easily and additionally can be opened physically to embed smaller modeling elements that can be bound to additional contextual work information, such as documents, forms, or other, already existing models. This approach allows mutually linking models, and more explicitly situates them in their context of work.

Interaction with the system has been designed based upon real-world activity metaphors (Fishkin 2004), which improve learnability of the interaction with the system (Oppl and Stary 2011). The interaction features are described in more detail in the following:

User-defined representational semantics. Models of individual perceptions of work have to allow arbitrary model element types to avoid misrepresentation (Oppl 2018) or loss of information due to lacking support of what people want to express (Goguen 1993; Sarini and Simone 2002). Typically, the semantics of a representation also evolves in the course of identifying/putting nodes on the tabletop and creating links. As more than one person may be part of a mapping session, essential nodes and relations can be shared and stored as meaningful information for groups, including the generation of variants with respect to a certain issue (Rentsch et al. 2010). Typical variants of course designs are subject-specific lectures for different curricula, for example, computer science and business information systems, involving educators with different intentions and learners with heterogeneous backgrounds.

The tool provides various types of tokens for modeling, and their respective meaning has to be assigned by the user(s) through labeling. An arbitrary number of token types is supported, and the shapes available as hardware tokens can be configured dynamically in the software system. Various categories of tokens enable the flexible specification of concept classes and the flexible assignment of meaning in the course of structure and/or behavior modeling.

At the same time, semantics can also be constrained to the pre-specified elements of existing modeling languages such as Subject-oriented Business Process Management (S-BPM). In this case, the interactive modeling support still provides the features listed in the following but is used analogous to the card-based modeling approach described in Sect. 3.2.

Labeling and associating. Work process modeling relies on the ability to assign names to model elements, to define associations between them, and—in case of clarification—to attach annotations to objects. In traditional, software-based modeling tools, these interactions are performed using mouse and keyboard. Using traditional input devices in a physical modeling environment requires switching between media. It might distract users from their original modeling task.

Accordingly, the interface has been designed to avoid input devices like mouse or keyboard. Several tools can be used to manipulate the model directly on the surface: Tokens are associated by putting them into close proximity and then placing them back in their original position (like linking them with a rubber band). Directed connections can be created using an arrow-tip-shaped tool that is put onto the connection near the intended endpoint. A rubber-shaped token enables users to delete connections. In case of multiple connections between two tokens, these connections are dynamically spread to avoid overlapping. Token types (e.g., red, blue, and yellow elements) are not semantically predefined. Users can assign meaning to them in the course of using *Comprehend*, as described earlier.

Labeling (i.e., assigning designators to concepts, cf. Fig. 7.3) is performed by using the keyboard for naming tokens or connections. The input text is assigned to the most recently added element (token or association), thus avoiding explicit selection of the target. A pen-shaped selection token enables the explicit selection of an element to rename it.

Abstraction support. Features like zooming or the selective display of concepts allow reducing the complexity of visualizations. They are, however, restricted to the computer-based desktop or multi-touch tabletops without any tangible elements. The tokens act as containers in order to overcome this limitation and to reduce complexity in physical models, too (cf. Fig. 7.4). They represent either an arbitrary digital resource (file), or a model state captured previously. The latter information type enables users generating parts of a work representation separately and connecting these parts on a higher level of abstraction. In this way, the common

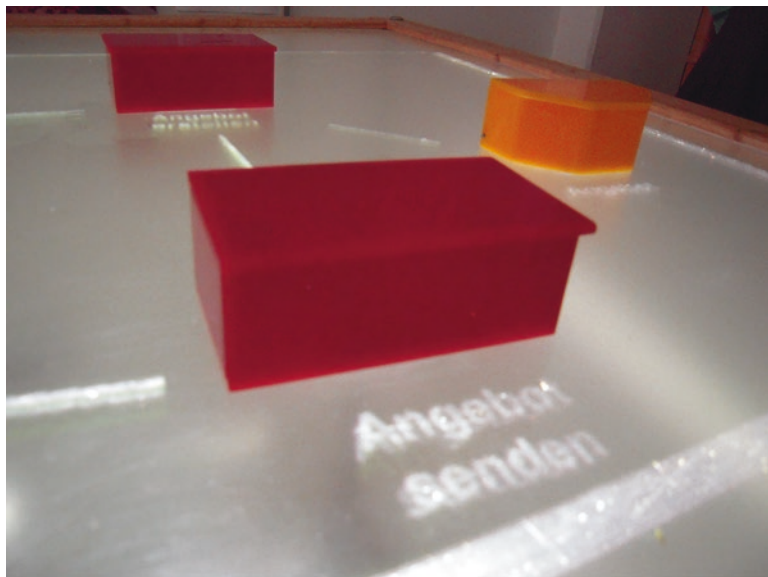


Fig. 7.3 Labeling and associating

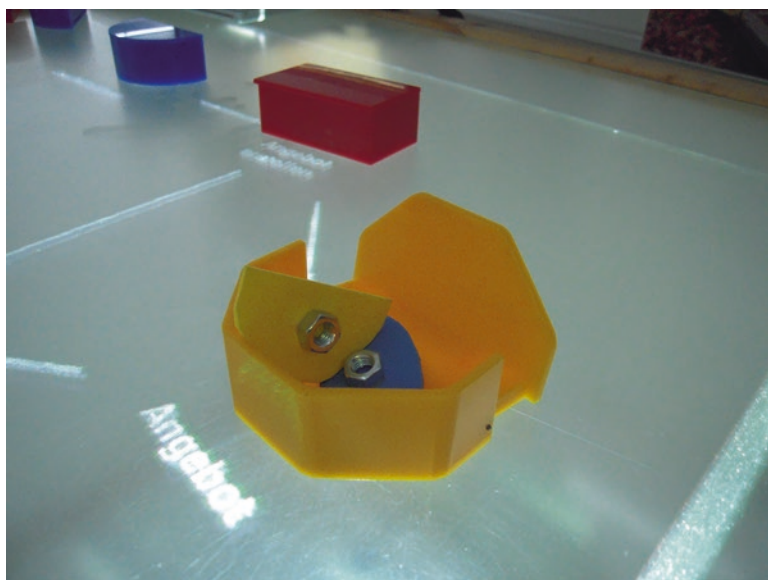


Fig. 7.4 Users can open a token and put additional information into it. Additional information is bound to smaller tokens

modeling concept of abstraction through subsuming or detailing representations is mapped to the physical world. In the case of S-BPM modeling, the container metaphor is used to link individual behavior models in subjects, which are then used for interaction modeling.

The ratio between token size and the size of the table surface allows about 10-15 tokens to be placed on the physical surface simultaneously. For complex modeling tasks, this number of elements might be too small. The container feature is designed to overcome this deficiency according to its purpose of nesting and embodying.

History and reconstruction. The traceability of the modeling process is ensured in Comprehand through capturing the modeling history. This feature also facilitates the understanding of a representation (Klemmer et al. 2002), in particular for cooperative endeavors. The modeling history enables participants to recapitulate and reflect the modeling steps made so far, even when they join a session later on, or in case they have to continue working on a model generated by different individuals.

The tool captures the modeling history by taking snapshots automatically. Whenever the model has not changed for a couple of seconds, the system takes a snapshot of the current state. In addition, a dedicated camera-shaped token enables users to take snapshots on demand. It allows explicit capturing and storing a certain model state using the back-end system for later retrieval. The users can navigate back and forth in the modeling process using the stored information.

The history mode (i.e., recalling former model states) can be activated using a clock-shaped token. It can be rotated counterclockwise or clockwise to go back and forth in time, respectively. When the users switch to the history mode, the computer screen displays a graphical visualization of the currently selected model state along with a status bar, indicating the point in time when the state has been captured.

Additionally, the modeling history enables support for rolling back changes of the model. This is necessary when encouraging the exploration of potential model elements (concepts) and associations. Experimental changes need to be reversible. Such a requirement can be implemented in a straightforward way for desktop applications, but hard to accomplish in a physical modeling environment. The reconstruction feature built upon the history navigation mode supports the physical reconstruction of a previously selected model state. When triggered, Comprehand guides the

users step by step and indicates visually which physical tokens need to be (re)moved and/or added according to the differences between current and requested model state, in order to complete a reconstruction of the model state on the table surface. When reconstruction is triggered, a new sequence of modeling states is forked from the original strain of model development, as proposed by (Klemmer et al. 2002). In this way, not only the temporal evolution of the model but also its conceptual development and alternative ways of model representation become accessible. These complex model histories, however, are not accessible via the tabletop but only via the desktop system in order to keep interaction during modeling simple.

Figure 7.5 shows the set of tabletop elements and the toolset for:

- Selecting elements (node or link) of a tabletop map going to be manipulated
- Marking a link as directed relationship, for example, indicating a procedure (chain)

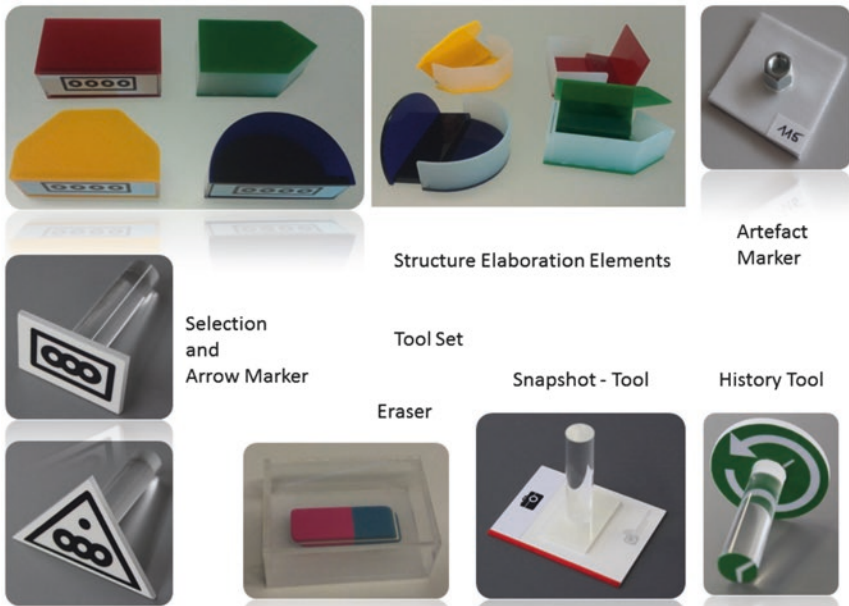


Fig. 7.5 Elements and tools for tabletop concept mapping

- Removing a link or text label of a node or link (eraser tool)
- Storing the current state of the map as a snapshot in a repository for later use (snapshot tool)
- Step back in time showing previous snapshots (history tool)

7.1.3 Collaborative Model Articulation and Exploration

Comprehend is generally used by individuals for reflection and articulation purposes, although it can be applied in multi-user settings as well (Furtmüller and Oppl 2007; Wachholder and Oppl 2012; Oppl 2013). In a multi-user setting, the participants gather around the table. From a methodological point of view, they need to agree on their modeling task in the form of a focus question (Novak and Canas 2006) or a topic of interest (Dann 1992) that targets at the cooperative work process or contingency at hand. The remainder to the instrument's application is a self-moderated process, with interventions only happening in case technical issues arise.

The tabletop allows for simultaneous access to the modeling surface, allowing all participants to freely place and associate physical tokens. They start by placing a token on the surface and assign a designator describing the meaning of the concept represented by the token. Tokens are available in three different shapes and colors. They are, however, semantically not predefined. When using a certain kind of tokens, the participants cooperatively specify their meaning and thus a class of concepts relevant to the topic to be modeled. This meta-information is also captured by Comprehend, which provides means of textual specification of the concept type whenever a new kind of token is used.

As the model evolves, more tokens are placed on the table surface. They are again labeled and can be put into explicit mutual relationship by briefly moving them into close proximity. Projected lines between the associated tokens visualize relationships. These associations also can be labeled, if considered necessary by the participants. All labeling processes are performed using a wireless keyboard, which is passed among the participants as required.

The model on the table surface eventually represents the agreed-upon view all involved participants have on the topic of modeling. During the modeling process, however, the individual views might be incomplete, complementary, or even controversial. Exploration of the modeling space (in terms of different concepts and associations among them) facilitates the articulation process among the involved participants and allows resolving open issues. The system keeps track of the model evolution and separates different strains of development in case of temporary experimental changes. This allows recapturing the modeling process at a later point in time but also enables active exploration of different model variants and their documentation, as model changes become traceable and can be undone. Comprehand supports reconstruction of prior model states on the tabletop surface, as described earlier.

When the participants agree that they have come to an end and have found a common model representing their views on the modeling topic, they are able to make persistent both, the final model and the modeling process. Using the semantically flexible form of representation via Topic Maps, as mentioned earlier, the model, its semantics and its history can be captured in a single, self-contained file using standardized means of representation. The modeling process can be reproduced along its time line and can be resumed on the table surface whenever requested. Comprehand Tables in addition can be linked with each other to allow for spatially (Oppl 2011) or conceptually distributed collaborative modeling (Oppl and Rothschädl 2014; Wachholder and Oppl 2012).

The Comprehand Table provides all these features in order to support conceptual modeling with a focus to facilitate explicit articulation and alignment of mental models through cooperative modeling rather than in a generic way. The tabletop acts as an enabler for communication between the involved persons in the process of performing Articulation Work, requiring the elaboration of models only in so far as they can serve as common point of reference. Model completeness and correctness with regards to formal semantics are not relevant in this case. The evaluation of the instrument consequently has focused on the usability of the toolset (i.e., not hampering users in their alignment activities), its ability to support concept mapping as a means to support Articulation Work, and the effects of its application on actual cooperative work processes.

Models created with both, the table and the card-based instrument, are represented on a conceptual level and—together with their creation and revision history—become part of the distributed organizational knowledge base (DOKB) as knowledge claims to be processed further in the course of double-loop learning processes, according to the KLC.

7.2 Sample Actor-Centric Tool Support for Representation

An OL platform needs to aim at putting people seeking for or being able to provide knowledge in control of the transfer process. It also allows situation-specific communication among the parties involved in the transfer process. In the following sections, we review the feature-set of an OL platform clustered along the different knowledge types described earlier.

7.2.1 Representing Role Knowledge and Descriptive Meta-knowledge

Role knowledge is represented in the OL platform using fine-grain content objects. A content object is a conceptual building block within the knowledge to be represented, such as a definition, an example, or an explanation. Instead of using a document-centric approach to provide information, content is split in its fundamental didactical elements. These elements can be flexibly arranged and reused to form representations of role knowledge.

Descriptive meta-knowledge is codified in the navigation structures of developed demonstrator Nymphaea (Neubauer et al. 2013; Weichhart et al. 2018) as well as directly anchored on content objects (Neubauer et al. 2009, 2011). The Nymphaea learning environment provides different “workspaces” for users, each one containing the relevant knowledge representations, necessary for a certain task. It provides meta-knowledge about the domain-specific scope of knowledge, its author, and creation date. Content within a certain workspace comprises of modules and hierarchically structured content objects. These content objects are enriched

with educational meta-knowledge such as ‘definition’, ‘motivation’, ‘background information’, ‘directive’, ‘example’, or ‘self-test’. This didactic meta-knowledge is displayed on the right side, on top of each content element (cf. Fig. 7.3), and can be used in the course of individualization when filtering content according to metadata.

Besides structuring content according to educational and domain-specific metadata, Nymphaea provides means to structure content according to level of details, allowing actors to retrieve content in the desired granularity.

7.2.2 Representing Conceptual Meta-knowledge

The OL platform provides an alternative navigation design focusing on domain-inherent structures that can be used complementary to hierarchical navigation. The navigation design represents and organizes conceptual meta-knowledge in a graphical concept map.

Concept maps (Novak 1995) are established means to organize and represent knowledge. They can be used to support the process of eliciting, structuring, and sharing knowledge and aim to enable meaningful learning (Chabeli 2010; Stoyanova and Kommers 2002; Steiner et al. 2007). Concept maps use concepts as entity to structure items of interest. Concepts might be central terms, expressions, or metaphors, as they represent a unit of information for the person(s) using it. Those items are put into mutual context, leading to a network of concepts. Persons express the items of interest and the relationships by means of language constructs. Per se, there are no restrictions in the naming of concepts or relationships.

Compared to the traditional navigation design, the concept map navigation enables domain-specific and cross-border relationships. Knowledge acquisition paths can considerably differ when using the concept map approach. Instead of implicit learning paths—via hierarchies of modules, content units, blocks, via internal/external links—learning paths using a concept map are oriented towards explicit structural relationships beyond hierarchies and domains.

Figure 7.6 depicts a part of a cross-disciplinary concept map for codified knowledge about ‘Enterprise Architecting’. It can be used for navigating content.

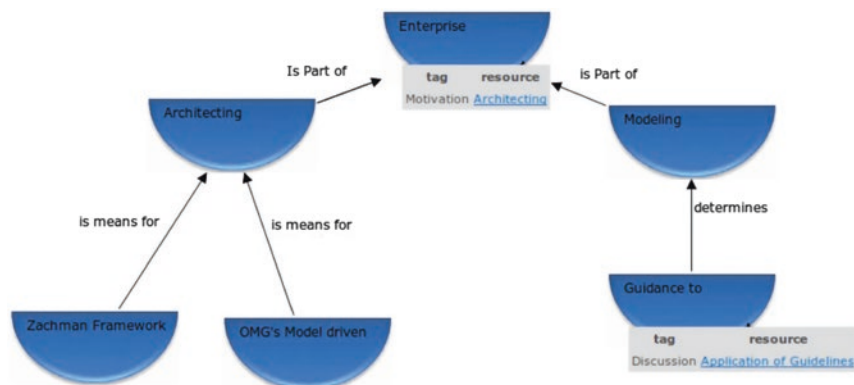


Fig. 7.6 Exemplifying CMap navigation and content links

Within the map, domain-specific associations are used for relating concepts. Furthermore, descriptive meta-knowledge (such as motivation and discussion—see Fig. 7.6) is used to semantically describe links between concepts and information resources. Hence, the associative navigation provides additional structural information that shapes learning paths and can guide the individual exploration of content.

7.2.3 Enabling the Assessment of Cognitive Meta-knowledge

Cognitive meta-knowledge (i.e., knowledge about one's own knowledge) and persuasive meta-knowledge (i.e., knowledge about the credibility and expertise of the knowledge source) are not directly represented in content structures in learning support systems, such as Nymphaea. However, they rather provide means to enable people seeking for knowledge to assess whether they need to access and acquire certain content. A specific instrument termed “Intelligibility Catcher” aids the acquisition of meta-knowledge (Chris Stary 2007). Intelligibility Catchers (ICs) are grounded in reformist-pedagogic and constructivist didactics in general, and the Dalton Plan approach in particular (Parkhurst 1922; Chris Stary 2007; Eichelberger et al. 2008).

The main objectives of the Dalton Plan are to provide freedom to learners and enable them to follow individual learning paths, involving group interaction and collaboration (Parkhurst 1922; Chris Stary 2007). The main vehicle guiding learners through larger and complex (group) learning tasks is the learning contract. It provides a specific structure which can be implemented through ICs. Based on the pedagogic foundation of Parkhurst's assignments, ICs are additionally tailored to be methodically and functionally integrated in online knowledge transfer environments. The IC's structure is as follows (clustered along transactive memory dimensions):

- Elements aiding the assessment of cognitive meta-knowledge
 - *Preface (Orientation section)*: The preface section provides the context motivating the knowledge acquisition tasks.
 - *Topic/objectives*: This section clearly states the central idea of the subject to be addressed. This helps learners to stay focused and reflect about their own work on/about this topic.
 - *Problems and tasks*: This section includes all tasks learners work within the frame of the current contract. It is advisable to state here which problems are to be solved individually by each learner and which problems are to be solved by a group of learners.
- Elements aiding the creation of instance and transactive knowledge
 - *Written work*: This section identifies the documentation to be provided by learners. When finished, the involved people discuss written work within a meeting/conference (see later in the chapter). ICs particularly should include references to functionality provided by a learning support system platform to effectively structure and support the learning process.
 - *Memory work*: In this part of the assignment, the intellectual and cognitive work is described. It comprises the intellectual effort to be spent when exploring content and apply it in a reflective way for problem solving.
 - *Conferences/meetings*: While learners are required to manage their own (learning) time, it is often advisable to schedule (online) meetings and check the intermediate progress.

- Elements containing descriptive meta-knowledge
 - *References*: All additional content for which it is advisable to be read, are referenced here.
 - *Equivalents*: The estimated effort (in hours of work) for the assignment is provided here.
 - *Bulletin Board*: A forum dedicated to discussions related to the assignment is provided.

Using this structure, guidance on how to use and interact based upon content for knowledge acquisition is provided. Following a self-regulated learning paradigm (Oppl et al. 2010), cognitive meta-knowledge is not directly captured and represented explicitly in a platform; however, users are enabled to self-assess their existing knowledge and learning requirements. Elements specifically targeting at didactically reasonable interaction among learners and knowledge providers facilitate building transactive knowledge and allow making use of instance knowledge within and external to the platform.

7.3 Sample Actor-Centric Tool Support for Intelligent Content Manipulation

Support for individualization of content can be provided in an OL platform with respect to capture instance knowledge (Fürlinger et al. 2004). Annotations enable individuals to (i) annotate or alter a specific content element, (ii) post questions, answers, or comments directly anchored on content, and (iii) additionally link the contribution to a discussion theme from the system's global discussion board. The latter link (being part of navigation) guides users to the adjacent discussion of the learning material. In case of real-time online connections, such as chats, the questions and answers can pop up immediately on the displays of all connected users (available in a buddy list). In addition, the content elements referred to can be displayed at the same time.

Annotation support for content is realized using a view concept. As soon as provided content is displayed, a view is generated like an over-

lay transparency. The view is kept for further access and reloaded when the content is accessed again. Within a certain view, users can (i) highlight, (ii) link, and (iii) add remarks to content elements. The features for view management (add view layer, delete view layer, share view layer, show available views) as well as those for annotations are located in the ribbon-bar at top, whereas the selection of a certain view is provided at the right-hand top of the content area (cf. “MyView” in Fig. 7.6).

While annotating content, users can add internal and external references to content items. Internal references are links between content and communication items, such as entries in the discussion forum or Infoboard, which support context-sensitive discussions. Furthermore, internal links might refer to other elements within the same or a different module. The corresponding features have been included into the annotation icon bar (cf. Fig. 7.6 ‘Link’). Editing internal links requires marking a position in the text that should represent the link. After evoking the respective function located in the ribbon bar at the top, a tree with the node of the currently addressed module is displayed. It allows users to select the target of the link (e.g., a forum entry or another content item).

Coupling content and communication is the core concept in the learning support instrument to foster sharing of instance knowledge. Features supporting sharing are integrated with the individualization features to comprise the possibility to contextualizing individual interactions by directly anchoring them on content elements. Sharing of individual views or creation of shared views, as suggested by (Shi-Kuo Chang et al. 1998), is enabled in the system.

The system allows linking content elements to forum and discussion entries, and vice versa. Sharing these links in a group enables the group to discuss the provided content in context. This feature is particularly useful when users are not only “passive” recipients of content but also actively provide or augment role or instance knowledge in the workspace. Having the discussion documented in the forum provides new users with justifications and background information that has led to previous revisions of content (Weichhart and Stary 2009).

7.4 Sample Actor-Centric Tool Support for Processing Work Models

Validation and refinement of work models by simulated enactment are supported by a workflow management system (WfMS) that has been adapted to allow for dynamic reconfiguration of processes during runtime. The WfMS is based on the S-BPM paradigm (Fleischmann et al. 2012) and integrates with a computer-based modeling environment via a shared model repository, which is part of the DOKB. One of the methodological requirements is to enable refinement of the process models during the simulated enactment whenever an issue is recognized. To avoid losing the context of the enacted work process (i.e., losing all entered data or information about already made decisions), these model changes must not require a restart of the simulated enactment. The need to restart workflow execution in the case of model changes is a technical constraint of most currently available workflow execution engines (Rothschädl 2012). The execution engine has been functionally extended in the course of adapting it to the needs of the methodology presented here and now supports deviations from a currently executed process model during runtime without the need for restarting the process and losing the execution context (Rothschädl 2012).

The overall architecture of the system is outlined in Fig. 7.7. A central model repository is used to store process models. The model importer

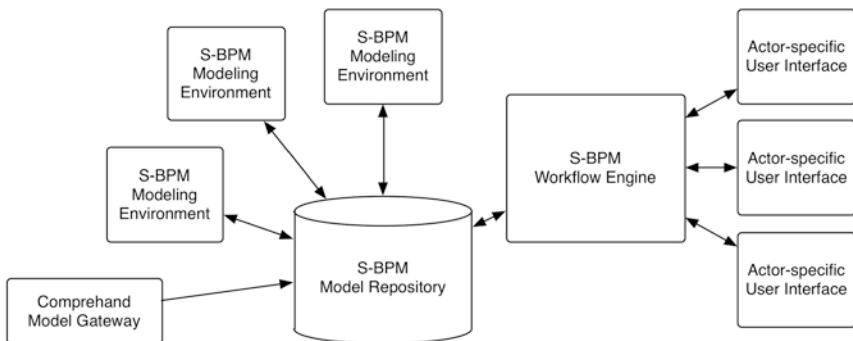


Fig. 7.7 Architecture of process enactment environment

provides the gateway to the articulation tools described earlier and uses the repository as a target for the resulting S-BPM models. The repository is used by the workflow engine to retrieve information about currently instantiated processes. Running process instances are made accessible via a web interface that offers individualized task lists for all actors. Process models stored in the repository can be altered at any time by accessing them via a modeling environment. The architecture visualized in Fig. 7.6 of the repository allows for synchronously altering models from different modeling environments.

7.5 Towards Seamless Tool Support—A Showcase

This section demonstrates the methodology and the use of the toolset based on a simple organizational setting. The aim of this section is to visualize the potential interplay between the distinct areas of support described in the former section. The same set of tools and methods as presented there, is used here for reasons of consistency.

The organizational setting used as a showcase here is not complex, either content-wise or in terms of required collaboration. Its simplicity, however, allows visualizing the intended effect of the proposed methods and tools in a straightforward way. More complex situations would require a more comprehensive description of the organizational setting, which would be beyond the scope of this paper.

The sample process is applying for a vacation. As a starting situation, we assume that the process of vacation application has been handled informally so far in our sample organization. The aim now is to establish a process that is agreed upon throughout the organization and support it by means of IT. In this context, informally established routines should be made visible, questioned, and examined for potential improvements (cf. double-loop learning in the KLC). Three organizational roles are originally involved in the process: an employee, the secretary, and the manager. For the initial articulation and elicitation activities, representatives of these roles are brought together for a workshop.

7.5.1 Articulation and Elicitation

The front-end for articulation and elicitation of knowledge about work processes is based upon the concept of structure elaboration techniques. In the initial workshop, card-based models are created to establish a shared understanding of the process across all involved roles and identify potential differences in how the necessary interaction is perceived.

In a first step, the individual views on one's own contributions to the work process are articulated by each involved person separately. In a second step, those individual views are consolidated in a common model. In the process of consolidation (cf. Oppl 2015), different views are made explicit and are required to be resolved. The card-based model here acts as a mediator between the involved people and always reflects the current state of shared understanding (cf. Fig. 7.8, left).

The interactive tabletop surface can also be used for modeling (cf. Fig. 7.8, right). In collaborative settings, it is used for concept mapping (Oppl and Stary 2009) to reconcile different understandings of fundamental elements of a work situation, such as artifacts, documents, or involved roles and their relationships. In the showcase, this would involve agreeing upon the relevant information to be submitted with a vacation application, identifying used artifacts such as a calendar and reflecting on which organizational roles are actually involved.

The interactive tabletop can also be used individually and collaboratively to reflect upon the procedural aspects of work. It here replaces the card-based system. While the tabletop is not as flexible with regards to

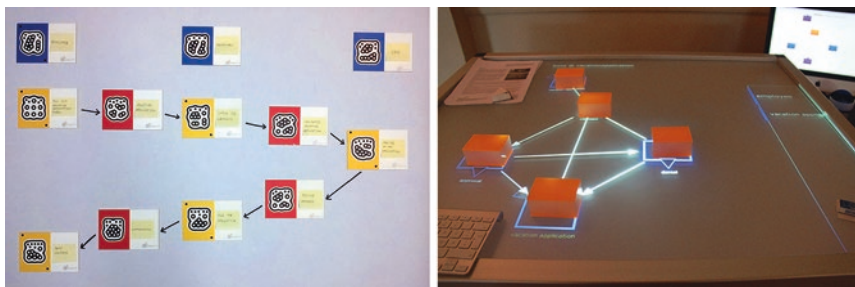


Fig. 7.8 Card-based model (left), interactive surface modeling (right)

spatial deployment, it allows for more sophisticated modeling support features such as version tracking and model reconstruction support (Oppl and Rothschädl 2014).

A combination of both tools is also possible from a methodological perspective. Card-based modeling does not require any dedicated infrastructure and is easy to use. It consequently is used for in-situ elicitation at the workplace. The interactive surface provides more comprehensive modeling support that is useful for model reflection and revision, which is an integral part of both, single- and double-loop-learning. Technologically, the tools are integrated via a shared representation, which is discussed in the following section.

7.5.2 Representation

All models, together with their creation history, become a part of the DOKB to be accessible during all stages of the knowledge processing and business processing environment in the KLC. Models must not only be stored as graphical representations but rather needed to be represented on a conceptual level to allow for fine-grain referencing and interlinking with other content.

The digital model versions created by interactive modeling surface can be stored in the DOKB without further transformation. The card-based models are only available as images and need to be processed further for deriving a conceptual model representation. A model recognition engine is used for that purpose (Oppl 2015). It offers web-based and mobile gateways to trigger model recognition and interactive revision support (cf. Fig. 7.9). The recognition engine outputs XML-based model representations using the semantics of the modeling language used during model creation. For the showcase, these could either be semantically open concept maps or role-distributed work process models.

The XML-based model representations are then imported in the learning platform, which serves as the DOKB. Here, each piece of content is augmented with metadata about authorship and its relationship to other content elements to allow for an implementation of a transactive memory system, as described earlier. The manipulation of content within the DOKB is discussed in the next section.

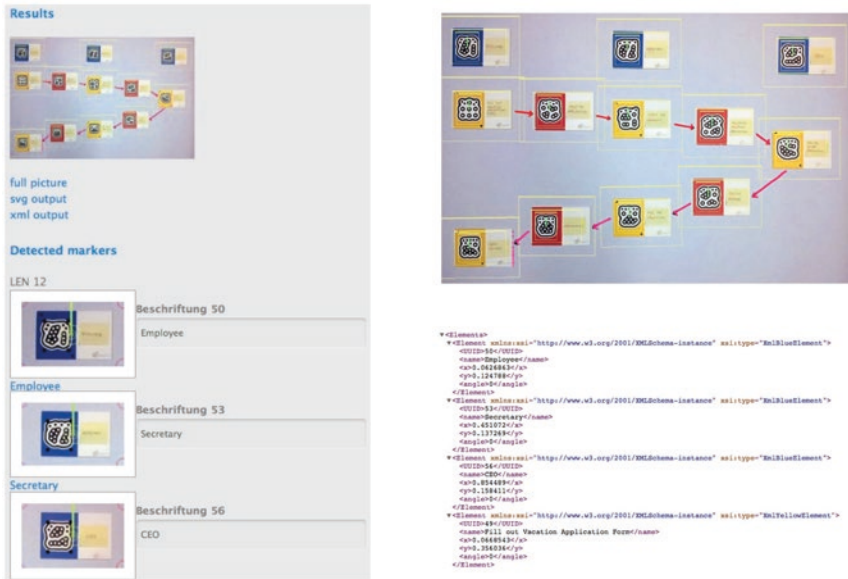


Fig. 7.9 Card-model recognition for conceptual representation: web-interface (left), recognition results (top right), XML-based model representation (bottom right) (released under a Creative Commons Attribution 4.0 International License (CC BY 4.0))

7.5.3 Manipulation

Manipulation of content here refers to embedding it into individually perceived or collectively agreed-upon organizational contexts. This involves searching for, retrieving, and linking relevant content to develop situation-specific perspectives in the DOKB. These perspectives can be shared with others and are available for annotation and discussion. The system implementing these features constitutes the main access path to the DOKB repository. It is built upon the Liferay portal software, which has been adapted to provide the features described earlier.

Figure 7.10 shows sample content from the showcase. The content area (marked “2”) shows the individually created process model of an employee’s activities in that vacation application process. It has been derived from a card-based model, as described earlier. The navigation area (marked “1”) allows accessing the different content types (models,

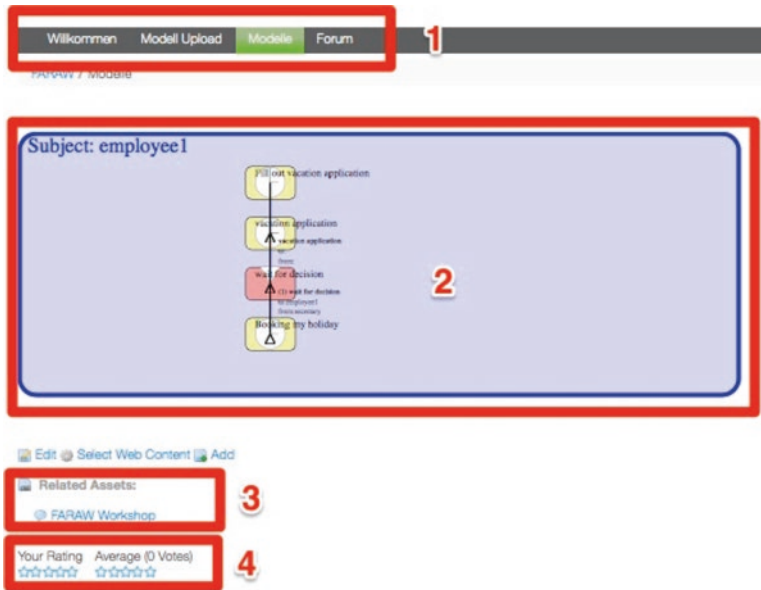


Fig. 7.10 Work process content in the learning environment (released under a Creative Commons Attribution 4.0 International License (CC BY 4.0))

forum-based communication) created for the respective situation. Content cannot only be accessed hierarchically but also contextualized by interlinking different content elements. This is visualized as an example in Fig. 7.10 by a link from the model content to a forum discussion (marked “3”). This link is bidirectional and can also be accessed from the forum, providing immediate context to the discussion. A prototype for a social interaction feature is shown at the bottom of Fig. 7.10 (marked “4”). It allows for rating the usefulness of the provided information.

The linking and rating areas also allow integrating external content providers (such as wikis or document management systems) or social service and communication providers (such as Facebook, Yammer, or Twitter). This allows for integration of platforms already established as means for collaboration and communication in an organization and does not enforce using “yet another” platform. To be able to process information from other applications contributing to the DOBK, an interopera-

bility platform is researched (Weichhart 2014). Technically based on a well-established and simple-to-use REST API approach, the platform provides an enhanced enterprise-service-bus-like environment. This enables not only technical support for the distribution aspect of the DOBK. This approach is in particular suitable for the KLC, as it also conceptualizes the organization as CAS and assumes changes in processes and the different applications and platforms to be the norm, not the exception. This interoperability service also allows for integration of tools for validation of content via simulation. This area of support is discussed in the next section.

7.5.4 Processing

Checking work process models for their feasibility in practical settings, reflecting on them as well as acquiring knowledge about them without prior practical experiences, can be supported by simulation tools. In the KLC, simulation supports the evaluation of knowledge claims and thus feeds back data in the DOBK. The simulation tool is used to execute work process models created with the modeling approaches described earlier. The tool chain described in this paper makes use of an execution engine that can directly process models that are focusing on the collaboration of roles involved in a work process and their activities.

In the showcase, the simulation step is used to assess the adequacy of the vacation application process model with regard to its usefulness in practical settings. The model is executed in a workshop setting, where the involved people check whether the simulated process matches their expectations and covers all potential process variants. Due to the nature of the used modeling approach, which focuses on single cases rather than generic processes, the latter cannot be observed in general. In the showcase, for example, the process steps to be taken in case of a rejection of an application could be missing. Whenever a mismatch or gap in the process model is identified, the model is altered or extended on the fly without restating the simulated process instance. This is technically realized via a workflow engine that allows dynamically changing process models during runtime (i.e., while instances of the process are being executed). In

the showcase, this would mean adding the respective activities to handle rejections to the behavior models of each involved role.

The simulation component retrieves its required model information from the repository that is also interfaced by the other components, such as the tabletop modeling tool used for elicitation or the learning platform (cf. Fig. 7.11). This allows to seamlessly switch between these tools even during a running simulation, annotating model variants in the learning

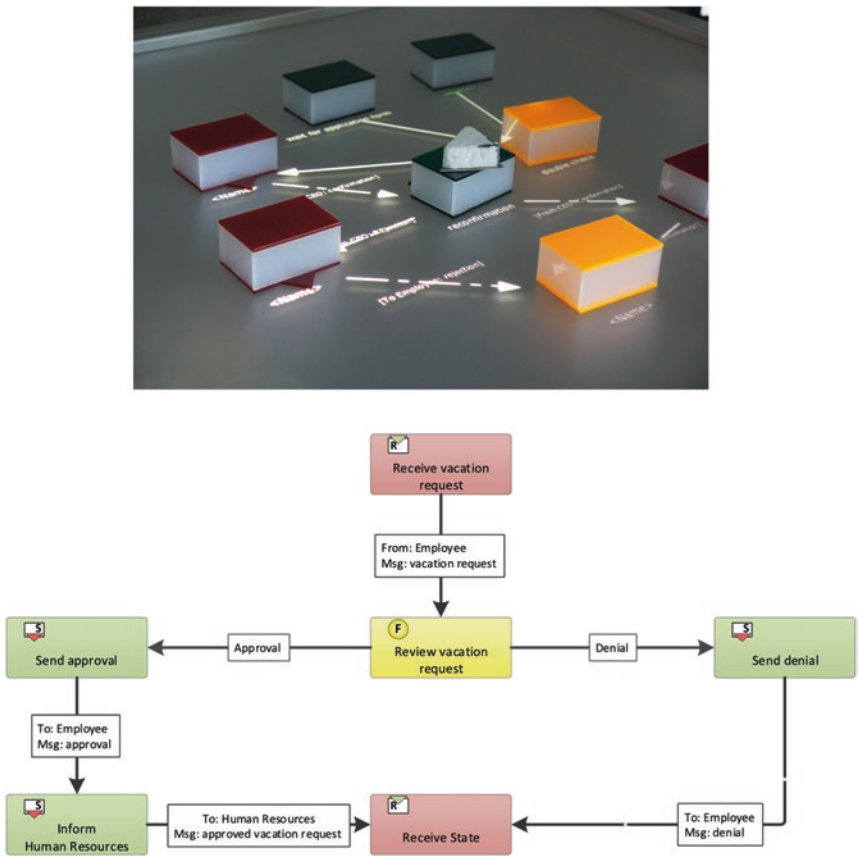


Fig. 7.11 Processing and simultaneous manipulation on an interactive modeling tabletop

platform or performing experimental model changes in the interactive tabletop surface. Consequently, the showcase process could be extended or altered by the original representatives of the roles, using the same tools they used during the initial articulation and elicitation step. Actors thus can perform all modeling steps without the need for expert modelers.

7.6 Conclusions

For effective digital work design, new approaches in involving stakeholders concerning process design are required. New methods and the intermediate usage of technologies could help agile organizations to structure their work on the fly while acknowledging their competencies and keeping their social identity. Concerned parties know best how their work can be improved, and their involvement in adaptation increases their capabilities to keep up with changing business requirements. A respective methodology and a tool chain need to be aligned with individual and collective learning facilities, both on the content and method level. It also requires a living memory to capture content and development processes in an actor-centered way.

In this chapter, we have backed the framework presented in the former chapter with actual technical tools. These tools offer support along the four elementary components of the framework: (i) articulation and elicitation of work knowledge, (ii) knowledge transfer and manipulation, (iii) multifaceted content representation, and (iv) business processes execution for validation and enactment. They thus can be matched to the methods and instruments introduced in the former chapters and thus form an overall coherent picture of how (digital) work design can be supported (by digital means), both methodologically and technically. We pick up this set of mutually interoperable and aligned instruments in the next chapter, where we present a set of case studies that show how they can be deployed (both selectively and in an integrated way, depending on the task at hand) in organizational practice.

References

- Chabeli, M. 2010. Concept-Mapping as a Teaching Method to Facilitate Critical Thinking in Nursing Education: A Review of the Literature. *Health SA Gesondheid* 15 (1) (OpenJournals Publishing).
- Dann, H.D. 1992. Variation Von Lege-Strukturen Zur Wissensrepräsentation. In *Struktur-Lege-Verfahren Als Dialog-Konsens-Methodik*, Arbeiten Zur Sozialwissenschaftlichen Psychologie, vol. 25, ed. Brigitte Scheele, 2–41. Münster: Aschendorff.
- Decker, G., and M. Weske. 2009. Toward Collaborative Business Process Modeling. *Cutter IT Journal* 22 (10): 29.
- Eichelberger, Harald, Christian Laner, Wolf Dieter Kohlberg, Edith Stary, and C. Stary. 2008. *Reformpädagogik Goes eLearning: Neue Wege Zur Selbstbestimmung Von Virtuellem Wissenstransfer Und Individualisiertem Wissenserwerb*. Munich: Oldenbourg Verlag.
- Engelmann, T., and F.W. Hesse. 2010. How Digital Concept Maps About the Collaborators' Knowledge and Information Influence Computer-Supported Collaborative Problem Solving. *International Journal of Computer-Supported Collaborative Learning* 5 (3): 299–319.
- Firestone, J.M., and M.W. McElroy. 2005. Doing Knowledge Management. *The Learning Organisation Journal* 12 (2): 189–212.
- Fischer, Frank, and H. Mandl. 2005. Knowledge Convergence in Computer-Supported Collaborative Learning: The Role of External Representation Tools. *The Journal of the Learning Sciences* 14 (3): 405–441. https://doi.org/10.1207/s15327809jls1403_3.
- Fishkin, Kenneth P. 2004. A Taxonomy for and Analysis of Tangible Interfaces. *Personal and Ubiquitous Computing* 8 (5): 347–358 (Springer).
- Fleischmann, Albert, Werner Schmidt, C. Stary, Stefan Obermeier, and Egon Börger. 2012. *Subject-Oriented Business Process Management*. New York: Springer.
- Furtmüller, F.G., and Stefan Oppl. 2007. A Tuple-Space Based Middleware for Collaborative Tangible User Interfaces. In *Enabling Technologies: Infrastructure ...*, 400–405. IEEE. <https://doi.org/10.1109/WETICE.2006.71>.
- Fürlinger, S., A. Auinger, and C. Stary. 2004. Interactive Annotations in Web-Based Learning Systems. In *Proceedings of IEEE International Conference on Advanced Learning Technologies, 2004*, 360–365. IEEE. <https://doi.org/10.1109/ICALT.2004.1357437>.

- Goguen, J. 1993. On Notation. Department of Computer Science and Engineering, University of California at San Diego.
- Hornecker, E. 2001. Graspable Interfaces as Tool for Cooperative Modelling. *Proceedings of IRIS 24*: 215–228.
- . 2004. Tangible User Interfaces Als Kooperationsunterstützendes Medium. Department of Computing, University of Bremen.
- Ishii, Hiroshi, and Brygg Ullmer. 1997. Tangible Bits: Towards Seamless Interfaces between People, Bits and Atoms. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI)*, 234–241. New York: ACM Press. <https://doi.org/10.1145/258549.258715>.
- Jiang, Yingying, Feng Tian, Xiaolong Luke Zhang, Guozhong Dai, and Hongan Wang. 2011. Understanding, Manipulating and Searching Hand-Drawn Concept Maps. *Transactions on Intelligent Systems and Technology (TIST)* 3 (1): 1–21 (ACM Request Permissions). <https://doi.org/10.1145/2036264.2036275>.
- Kaltenbrunner, Martin, and Ross Bencina. 2007. reacTIVision: A Computer-Vision Framework for Table-Based Tangible Interaction. In *TEI '07: Proceedings of the 1st International Conference on Tangible and Embedded Interaction*, 69–74. New York: ACM Press.
- Klemmer, S.R., M. Thomsen, E. Phelps-Goodman, R. Lee, and J.A. Landay. 2002. Where Do Web Sites Come From?: Capturing and Interacting with Design History. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 1–8. ACM.
- Luebke, Alexander, and Mathias Weske. 2011. Bringing Design Thinking to Business Process Modeling. In *Design Thinking*, 181–195. Berlin and Heidelberg: Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-13757-0_11.
- Neubauer, Matthias, Stefan Oppl, C. Sary, and Georg Weichhart. 2013. Facilitating Knowledge Transfer in IANES—A Transactive Memory Approach. In *Innovation Through Knowledge Transfer 2012*, Smart Innovation, Systems and Technologies, vol. 18, ed. R. Howlett, B. Gabrys, K. Musial-Gabrys, and J. Roach, 39–50. Berlin and Heidelberg: Springer. https://doi.org/10.1007/978-3-642-34219-6_5.
- Neubauer, Matthias, C. Sary, and Stefan Oppl. 2009. Towards Topic Map-Based E-Learning Environments. In *Proceedings of TMRA 2009*, November. Poster at the Fifth International Conference on Topic Maps Research and Applications.

- . 2011. Polymorph Navigation Utilizing Domain-Specific Metadata: Experienced Benefits for E-Learners. In *Proceedings of the 29th European Conference on Cognitive Ergonomics (ECCE 2011)*, 45–52. New York: ACM Press.
- Novak, Joseph D. 1995. Concept Mapping to Facilitate Teaching and Learning. *Prospects* 25 (1): 79–86 (Kluwer Academic Publishers). <https://doi.org/10.1007/BF02334286>.
- Novak, Joseph D., and A.J. Canas. 2006. The Theory Underlying Concept Maps and How to Construct Them. Florida Institute for Human and Machine Cognition.
- Oppl, Stefan. 2006. Towards Intuitive Work Modeling with a Tangible Collaboration Interface Approach. In *Proceedings of WETICE '06*, June, 400–405. IEEE Press. <https://doi.org/10.1109/WETICE.2006.71>.
- . 2011. Subject-Oriented Elicitation of Distributed Business-Process Knowledge. In *Proceedings of the 3rd Conference on Subject-Oriented Business Process Modelling (S-BPM ONE 2011)*, Communications in Computer and Information Science, vol. 213, 16–33. Berlin and Heidelberg: Springer. https://doi.org/10.1007/978-3-642-23471-2_2.
- . 2013. Towards Role-Distributed Collaborative Business Process Elicitation. In *Proceedings of the Workshop on Models and Their Role in Collaboration*, ed. Alexander Nolte, M. Prilla, Peter Rittgen, and Stefan Oppl, 33–40. CEUR-WS.
- . 2015. Articulation of Subject-Oriented Business Process Models. In *Proceedings of S-BPM ONE 2015*, 1–11. New York: ACM Press. <https://doi.org/10.1145/2723839.2723841>.
- . 2017. Supporting the Collaborative Construction of a Shared Understanding About Work with a Guided Conceptual Modeling Technique. *Group Decision and Negotiation* 26 (2): 247–283. <https://doi.org/10.1007/s10726-016-9485-7>.
- . 2018. Which Concepts Do Inexperienced Modelers Use to Model Work?—An Exploratory Study. In *Proceedings of MKWI 2018*.
- Oppl, Stefan, and Christian Stary. 2009. Tabletop Concept Mapping. In *Proceedings of TEI 2009*, 275–282. New York: ACM Press. <https://doi.org/10.1145/1517664.1517721>.
- Oppl, Stefan, and C. Stary. 2011. Towards Informed Metaphor Selection for TUIs. In *Proceedings of the 3rd ACM SIGCHI Symposium on Engineering Interactive Computing Systems (EICS 2011)*, Communications in Computer

- and Information Science, vol. 213, ed. F. Paternò, 16–33 (Chap. 2). Berlin and Heidelberg: ACM Press. https://doi.org/10.1007/978-3-642-23471-2_2.
- Oppl, Stefan, and Christian Stary. 2014. Facilitating Shared Understanding of Work Situations Using a Tangible Tabletop Interface. *Behaviour & Information Technology* 33 (6): 619–635. <https://doi.org/10.1080/0144929X.2013.833293>.
- Oppl, Stefan, and Thomas Rothschädl. 2014. Separation of Concerns in Model Elicitation—Role-Based Actor-Driven Business Process Modeling. In *S-BPM ONE—Setting the Stage for Subject-Oriented Business Process Management*, Communications in Computer and Information Science, vol. 422, ed. Hagen Buchwald, Albert Fleischmann, Detlef Seese, and Christian Stary, 3–20. Berlin and Heidelberg: Springer International Publishing. https://doi.org/10.1007/978-3-319-06191-7_1.
- Oppl, Stefan, C.M. Steiner, and D. Albert. 2010. Supporting Self-Regulated Learning with Tabletop Concept Mapping. In *Interdisciplinary Approaches to Technology Enhanced Learning*, 391–410. Münster: Waxmann.
- Oppl, Stefan, Christian Stary, and S. Vogl. 2017. Recognition of Paper-Based Conceptual Models Captured Under Uncontrolled Conditions. *IEEE Transactions on Human-Machine-Systems* 47 (2): 206–220. <https://doi.org/10.1109/THMS.2016.2611943>.
- Parkhurst, Helen. 1922. *Education on the Dalton Plan*. Read Books Ltd.
- Rentsch, Joan R., Abby L. Mello, and Lisa A. Delise. 2010. Collaboration and Meaning Analysis Process in Intense Problem Solving Teams. *Theoretical Issues in Ergonomics Science* 11 (4): 287–303 (Taylor & Francis).
- Resnick, M., F. Martin, R. Berg, R. Borovoy, V. Colella, K. Kramer, and B. Silverman. 1998. Digital Manipulatives: New Toys to Think with. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 281–287. New York: ACM Press.
- Rothschädl, Thomas. 2012. Ad-Hoc Adaption of Subject-Oriented Business Processes at Runtime to Support Organizational Learning. In *S-BPM ONE*, Lecture Notes in Business Information Processing, vol. 104, 22–32 (Chap. 2). Berlin and Heidelberg: Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-29133-3_2.
- Sarini, Marcello, and C. Simone. 2002. The Reconciler: Supporting Actors in Meaning Negotiation. In *Proceedings of the Workshop on Meaning Negotiation*.

- Shi-Kuo, Chang, E. Hassanein, and Chung-Yuan Hsieh. 1998. A Multimedia Micro-University. *IEEE Multimedia* 5 (3): 60–68. <https://doi.org/10.1109/93.713305>.
- Stary, C. 2007. Intelligibility Catchers for Self-Managed Knowledge Transfer. In *Seventh IEEE International Conference on Advanced Learning Technologies (ICALT 2007)*, 517–521. IEEE.
- Steiner, C.M., D. Albert, and J. Heller. 2007. Concept Mapping as a Means to Build E-Learning. In *Advanced Principles of Effective E-Learning*, ed. N.A. Buzzetto-More, 59–111. Santa Rosa, CA: Informing Science Press.
- Stoyanova, Neli, and P. Kommers. 2002. Concept Mapping as a Medium of Shared Cognition in Computer-Supported Collaborative Problem Solving. *Journal of Interactive Learning Research* 13 (1): 111–133.
- Wachholder, Dominik, and Stefan Oppl. 2012. Stakeholder-Driven Collaborative Modeling of Subject-Oriented Business Processes. In *S-BPM ONE—Scientific Research*, ed. C. Stary, 145–162. Berlin and Heidelberg: Springer. https://doi.org/10.1007/978-3-642-29133-3_10.
- Weichhart, Georg. 2014. Learning for Sustainable Organisational Interoperability. In *Preprints of the 19th IFAC World Congress*, August. International Federation of Automation and Control, 4280–4285.
- Weichhart, Georg, and Chris Stary. 2009. Collaborative Learning in Automotive Ecosystems. In *Proceedings of the 3rd IEEE International Conference on Digital Ecosystems and Technologies*, 235–240. IEEE.
- Weichhart, Georg, Christian Stary, and Markus Appel. 2018. The Digital Dalton Plan: Progressive Education as Integral Part of Web-Based Learning Environments. *Knowledge Management & E-Learning: An International Journal* 10 (1): 25–52 (Directory of Open Access Journals).
- Zuckerman, Oren, S. Arida, and M. Resnick. 2005. Extending Tangible Interfaces for Education: Digital Montessori-Inspired Manipulatives. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI)*, 859–868. New York: ACM Press.

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

