

Improving Collaboration Efficiency via Diverse Networked Mobile Devices

Franca A. Rupprecht · Georg Kasakow · Jan C. Aurich ·
Bernd Hamann · Achim Ebert

Received: date / Accepted: date

Abstract We introduce a framework for distributed or co-located teams to collaborate highly efficiently using diverse mobile devices for design and assessment of complex systems. Our framework enhances the efficiency of collaborations arising in design, simulation or data analysis, including visualization. First, we investigate which requirements on such a collaboration framework exist and which influences between task models have to be taken into account; afterwards we transfer those findings into a prototypical system. The devices provide three views of data to be processed collabora-

tively: (1) a simulation view; (2) a status report view; and (3) a status update view. These views serve the purpose of providing overview, detail and performance views. A smart watch view shows at-a-glance information, the environment or the process being inspected, possibly influenced by a user. Users can use their mobile devices as control interfaces. The framework is especially effective for combining the synergistic, complementary competencies of a team. We describe the design of our framework and discuss specific applications.

Keywords Computer supported collaborative work · Groupwork · Secondary displays · Overview and Detail view · Interdependence · Mobile devices · Smart devices · HCI

Acknowledgements This research was funded by the German research foundation (DFG) within the IRTG 2057 "Physical Modeling for Virtual Manufacturing Systems and Processes"

F. A. Rupprecht
Computergraphics and HCI
University of Kaiserslautern
Tel.: +49 631-2053265
E-mail: rupprecht@cs.uni-kl.de

G. Kasakow
Institute for Manufacturing Technology
and Production Systems
University of Kaiserslautern
Tel.: +49 631-2054210
E-mail: georg.kasakow@mv.uni-kl.de

J. C. Aurich
Institute for Manufacturing Technology
and Production Systems
University of Kaiserslautern
Tel.: +49 631-2052617
E-mail: jan.aurich@mv.uni-kl.de

B. Hamann
Department of Computer Science
University of California, Davis
Tel.: +1 530-754-9157
E-mail: hamann@cs.ucdavis.edu

A. Ebert
Computergraphics and HCI
University of Kaiserslautern
Tel.: +49 631-2053502
E-mail: ebert@cs.uni-kl.de

1 Introduction

Rapid advances in computer networks and visualization-based human-computer interaction technologies are promising to impact a large spectrum of graphics-based design-simulations. For instance, conceptual design and critical assessment of complex systems generally requires large teams of scientists, engineers and planners to work together. The conceptual design process is extremely time-consuming, typically involving several iterations of different options before a generally acceptable solution is obtained. The collaborative framework we present is aimed at pointing out the efficiency gained when bringing diverse areas of expertise together, i.e., teams of

experts from various disciplines, all necessary to come up with acceptable concepts.

Examples where next-generation network-enabled collaborative environments, connected by visual and mobile interaction devices, can have significant impact are: design and simulation of automobiles [25] and aircraft [52]; urban planning and simulation of urban infrastructure (e.g., transportation, electricity, water and communication grids)[39]; or design of complex and large buildings, including efficiency- and cost-optimized manufacturing buildings [50]. The conceptual design and simulation-based evaluation of a new aircraft requires a manufacturer to bring together experts from mechanical engineering, electrical engineering, computer engineering, ergonomics, material science, air quality, health, and even more fields. Team-members often have to switch between tasks to achieve successful collaboration [64]. When members make changes, they affect the entire system and individually performed tasks. It is important to understand this impact and adjust tasks and operations accordingly [35]. The desire for a common framework to support decision-making in this process was a main motivation for our effort.

Ideally, in a distributed and collaborative networked environment experts can work independently or jointly on sub-systems of an overall design [71]. Adoption of existing and available device and network technologies is still in its early stages, and the integration into a collaborative system as described here is not realized. We introduce a framework enabling a team working in a distributed setting to collaborate via computer networks using various mobile interfaces and visualization devices. The framework makes possible the effective and synergistic combination of team members' complementary competencies and expertise. We address relevant challenges in the design and realization of an efficient, effective, and satisfactory collaboration framework.

Our framework can be adapted to the specific requirements of an application to support collaborative design done simultaneously. Using mobile phones as secondary displays is commonly done to provide a private and detailed view of data. Different aspects of the data can be represented in task-driven views (second display). Impact on a system caused by changes applied to it by another user is visualized in the main view (first display). However, the impact on particular tasks is transparent to a single user. The system we present can substantially enhance efficiency of a distributed collaboration environment for design, simulation and analysis efforts, and this is our main contribution.

The devices we use provide three views of the data processed collaboratively: (1) a simulation view; (2) a status report view; and (3) a status update view. These

views provide overview, detail and performance views, see Figure 6. A large display device, acting as a public viewport, provides an overview of the data for all participants in a simulation view containing a virtual reality application. Therefore, a smart watch is used to redistribute the status update view, while the status report view is presented on the smart phone. Locating the status update on a smart watch is done analogously to using a usual watch, where users capture information, time, at a glance.

First, we state which requirements on such a collaboration framework exists and which influences have to be taken into account and afterwards we transfer those perceptions into a prototypical system. We initially present our framework as a general framework, from an application-independent perspective. Later, we demonstrate the specific adaptation and utilization of it for a mechanical engineering scenario, which documents well the various benefits offered by our framework.

2 Related Work

Marquardt et al. [47] demonstrated that information exchange between multiple users via mobile devices as input and output devices can be facilitated in support of collaborative work. With the use of a public screen and mobile devices, the transfer of artifacts between differently scaled devices is supported. Awareness of participants and accessible content are emphasized. However, the performed task in the discussed setup is the same for all participants.

Versatility and design space with cross-device interaction using hand-held devices was investigated by Marquardt et al. [48]. Based on micro-mobility and F-formations, natural conversation in collaborations is facilitated in addition to content exchange and cross-device interaction between hand-held devices. However, there clearly exists a need to consider additional aspects of collaborative settings.

Mendes et al. [51] introduced CEDAR, a design review tool supporting collaborative tasks, using a Cave Automatic Virtual Environment (CAVE) and hand-held devices, acting as independent clients applied to the same scene. The CAVE-system is controlled by gesture tracking (Microsoft Kinect). The iPad acts as independent application, and the device's display shows a first-person view of the scene that is synchronized with the large screen. While the presented tracking setup only supports one active user, the iPad configuration is scalable to a multi-user setting. However, executing tasks cooperatively with the CEDAR system is not supported. Simultaneous work performed by several users is not considered.

Hühn et al. [27] pointed out the lack of evaluation tools for pervasive applications. Their CAVE-Smartphone setup was used to evaluate the User Experience (UX) of a location-based advertising application, where a smart phone is used as an alert mechanism. Smart devices offer a broader range of interaction capabilities, which are not fully exploited in this framework.

Anslow et al. [1] developed SourceVis, a collaborative visualization system for co-located environments based on multi-touch tables. The table provides a horizontal display on which one viewport per user is created on opposite sides, used for interaction. Single tasks can be performed on the individual’s viewport and collaboration is possible due to the same-location setting. The number of active users is limited to the size of the table.

Similarly to the work of Borchers et al. [7], Finke et al. [22] extended an interactive large public display (LD) with small devices (SDs). User interfaces are distributed across the differently scaled devices, and one can take advantage of the input and output capabilities of both devices. Unfortunately, only single user interaction was considered.

Keefe et al. [38] combined a hand-held multi-touch device with six degrees of freedom with a large-scale visualization display. The interaction with a large display is improved, and group work tasks can be performed. Single-task performance and integration is not covered.

Seifert et al. [65] introduced MobiSurf, integrating interactive surface capabilities and information exchange for team members’ personal and mobile devices, supporting co-located collaborative tasks. Cooperative task execution is presently not possible. Work done simultaneously by several users involving multi-role perspectives is not considered.

The “visual information-seeking mantra” of Ben Shneiderman [68] states: “*Overview first, zoom and filter, then details on demand*”. In the spirit of this mantra, overview-and-detail-view techniques are widely used and supported today by mobile devices [60], [15], [17]. These techniques also imply challenges for system design.

An “O+D interface” (overview-plus-detail interface) is related to coordinated views, implemented with one small overview provided on top of a larger detail view. An implication pointed out by Burigat [16] is the fact that O+D interfaces on mobile devices do not provide advantages in terms of navigation performance, compared to traditional presentation techniques. Considering small displays, users perceive O+D interfaces as being detrimental. Chittaro [19] determined that O+D techniques tend to fail on mobile devices, as it becomes more difficult to relate two different views due to limited screen space. He suggested to use visual references

pointing to interesting parts outside the visualization area, or intuitive methods supporting the switching between parts of the visualization.

As an alternative to the O+D technique Pelurson and Nigay [58] introduced a “bifocal view” as a focus-and-context technique for mobile devices. Myers [53] introduced semantic snarfing, where a region of interest is tracked via pointing devices and copied to a secondary hand-held device. Baumgärtner et al. [4] presented a hybrid 2D+3D interface for visual data exploration that combines visual design techniques with mixed-mode interaction capabilities, demonstrated for document management.

Related research is also being performed in robotics, where monitoring processes using an overview level is crucial. One well-known example is supervisory control, to allocate tasks to machines and monitor execution performance [67], [35]. One solution providing the desired insight is done via an additional display monitoring the status of a process [13].

3 Background – Supporting the Collaborative Workflow Needs of Teams in a Networked Environment

3.1 Collaboration Process

Collaboration is the combination and exchange of different core competencies and expertise, with the goal of creating a joint outcome in agreement, considering ideas and objectives of all participants. Based on a thorough literature review, we identified the main task phases involved in collaborative processes, depicted in Figure 1:



Fig. 1 Working phases in generalized task model of collaborative working.

Every collaboration session starts with the assignment of tasks and roles to each actor, which is facilitated by a software tool or performed in a group meeting setup beforehand (phase 1). On the one hand, assignment of tasks and roles is necessary to coordinate the work and team member in order to manage dependencies between activities[46]. Members without a task or a role can simply not participate in the group work. Passive or even no participation of actors lowers the individual’s satisfaction that is linked with motivation [6], engagement [10], and self-perceptions [26]. On the other hand, task assignment might lead to workload equality, which increases individual’s satisfaction as stated in [14]. This individual satisfaction may influence the performance of the complete team, which has been investigated in [45] but also the individuals willingness to continue the cooperation, what could be observed in [70]. Therefore, the first requirement (R1) on a collaboration framework is that member can coordinate activities.

In the next phase, users create drafts of the desired goals and the approach for reaching these goals (phase 2). This task can be performed individually or in a joint session. In this phase the outline of the work is created and the assigned tasks refined. This phase is accompanied by continuous comments and feedback loops, which are leading into group discussions (phase 3). The outcome of this phase is a draft in which ideas and expertise from all participants are considered and integrated. Communication between members across phases is crucial in order to synchronize the approaches to guarantee the progress towards the agreed joint goal. The communication between members is another requirement (R2) to support collaboration.

Afterwards the working style changes from group task to single task performance where the actual execution of the assigned tasks is performed (phase 4). The actual execution of the assigned tasks requires the active performance of an activity (R3). This phase is closely linked with the iterative reviewing and revision task (phase 5) in which all ideas, comments, and suggestion are discussed and incorporated. The exchange between members is necessary to communicate individual ideas and the satisfaction with the performance state. The decision about accepting the output is the outcome of phase 5 that leads to task establishment (phase 6).

3.2 Collaboration Styles

As the progress description indicates, collaborative work entails different working styles and phases. Among working styles we find a rough distinction based on the attribute *division of labor*, in group task performance and individuals task performance. Group tasks require

the participation of all team members, while individuals tasks describe that team members individually taking on responsibilities for focusing their own task goals to establish the team goal. Individuals tasks are performed simultaneous and independent to other members but the integration and collection of the results need to meet the requirements of all participants and is performed in close-coupled cooperation. Members most likely switch between the different working styles (R4), which is necessary to make one’s contribution towards the individual’s progress but also to control and adjust the group task performance.

In regards to computer support of teamwork additional attributes like the participants’ location needs to be considered. Therefore, the task performance can be identified upon the attributes *location*; indicating the physical location of the participants and *division of labor*. Thus, we can distinguish among four types of working styles between which actors switch most likely during:

- co-located group task performance;
- distributed group task performance;
- co-located single task performance;
- distributed single task performance.

3.3 Task Dependencies

Changes made by the independently and simultaneously operating team-members have impact on the overall system and the tasks being performed. Johnson et al. defined interdependence in the context of joint activity as follows: “*Interdependence*” describes the set of complementary relationships that two or more parties rely on to manage required (hard) or opportunistic (soft) dependencies in joint activity [35]. Our view of interdependence generalizes their definition. The existence of one task is not necessarily dependent on the existence or completeness of another task; there is no necessary “relies-on” relationship, but there may be a “can-be-positively-or-negatively-influenced-by” relation. The performance of one task can influence another task. Tasks can be interdependent through dependencies, but not as a consequence of merely existing. Dependencies and influences in only one direction can exist, which is not equivalent to interdependence describing a bi-directional dependence. Therefore, we use the term *inferdependency*, which refers to the combination of influence and dependence between two elements, one- or bi-directional.

We explain our notion of inferdependency via a simple example: Consider the scenario of biocenosis, where organisms coexist in the same habitat and interactions

are evident in food or feeding relationships. In this scenario we have 4 actors: 1) a flower, 2) a butterfly, 3) a bee, and 4) a bear. A butterfly depends on a flower for nectar (food); the flower depends on the butterfly to pollinate and make seeds for reproduction. A direct interdependence between the butterfly and the flower exists. In addition to the butterfly, a bee coexists in the same habitat. The bee depends on the flower to produce honey; the flower depends on the bee to cross-pollinate. A direct interdependence between the bee and the flower exists. Bee and butterfly coexist without influencing each other. However, in reality both species influence each other by cross-pollinating flowers, thereby accelerating reproduction. The task performance of both species makes their jobs (pollinating flowers) easier and leads to an overall improved outcome (higher reproduction of flowers), which is also the precondition (food) for task performance. Consider a third participant, e.g., a bear eating honey (produced by the bees). We find a direct relation between bees and bears and indirect relation between butterflies and bears. Thus, influences and (indirect) dependencies, implying interdependencies, between all participants exist, see Figure 2.

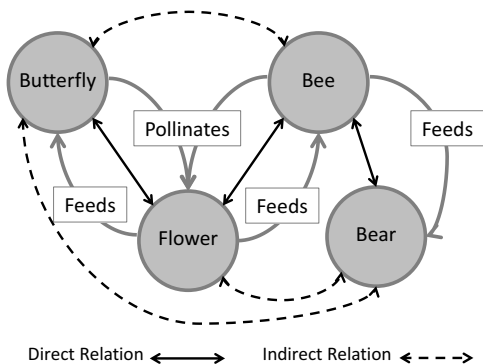


Fig. 2 Interdependencies showing direct and indirect relations between participants of a system.

Having interactive teamwork in mind, data and information exchange between experts has to be performed and interdependencies must be connected. Interdependent activities imply the presence of conflicting interests, which have to be coordinated to capture discrepancies before they become serious in order to achieve common goals with the help of common grounds [40]. Common ground is supported by continually informing others about changes that have occurred outside their views [40]. The determination of other's activities (R5) is therefore crucial. Johnson et al. stated that not all team members must be fully aware of the entire scope

of an activity; all must be aware of the interdependence in-between their activities [34]. Awareness of tasks and activities influences the coordination and task performance in a positive manner. Due to the establishment of shared knowledge and impact awareness (R6), team members can work together effectively and adjust their activities (R7) as necessary [56].

3.4 Requirements of Active Collaboration

In this work it is not our aim to deploy an all-embracing catalog for collaborative systems. Thus, to state necessary aspects that will be covered in this work. Our aim is to establish a work environment that allows one to perform individual tasks as well as group work task in a natural manner including the consideration of interdependencies. Based on the assumptions we state to support collaboration the requirements for collaborative work can be stated as follows:

- R1 Member can coordinate activities.
- R2 Member can communicate with each other.
- R3 Member can perform activities.
- R4 Member can switch between collaboration styles.
- R5 Member can determine others' activities.
- R6 Member can understand the impact of changes made.
- R7 Member can make adjustments based on impact.

4 Collaborative Framework Methodology

4.1 Collaboration Environment

To support a collaboration environment, we use a setup similar to the IN2CO (Intuitive and Interactive Collaboration) framework described by [61] that is enhanced in order to fulfill the requirements stated above. In analogy to Overview and detail views, a large display device presents an over-view of the complete data in form of a public viewport (see Figure 3).



Fig. 3 The IN2CO System: Large display device used as public viewing device, smart devices enabling private views of the collaborative task.

Supplementary, mobile devices are used to provide detailed information of task-driven aspects of the data

and they also act as input and control interface. In a user study performed for a typical, simplified factory-planning problem, it was demonstrated that team members could focus on the problem-solving task itself, instead of concentrating on interaction issues. By using the intuitive interaction capabilities provided by the smart devices, focusing on the actual task at hand was made possible. A virtual representation of the data on the shared viewport facilitated communication and decision making in a team-oriented manner.

Simultaneous work development is an important aspect to support efficient real time collaboration. Likewise different aspects and interpretations of the data matter of analysis are also important. Team-members have different interests, and, consequently, the data must be shown in multiple views. The IN2CO framework combines different task models together with visualizations for a shared public view on a large display devices and interaction and visualization capabilities for mobile devices. Elements of mobile devices are used to support particular tasks, while the public view visualization enhances the existing visualizations combined in one view. Impacts caused by changes of another autonomous team-member are considered in the public viewport, where changes of the entire system are visualized. Impacts on particular tasks are hidden in the complete system view and cannot be identified by a single team-member. Overview and Detail technique as well as Context and Focus technique are not sufficient to support collaborative work including consideration of interdependencies. Three different views are necessary to give insights in group performance, individuals' performance, and the visualization data. Our aim is to overcome the limitations by proposing a general framework holding three views: Simulation view, status update view, and status report view indicating overview, performance view, and detail view. The simulation view enact as Overview, holding a Virtual Reality (VR) application that generates realistic images and depictions of the processes. The status report view provides insights in individuals' aspect of the data. And the status update view indicates individuals' performance. The simulation view is located on a public large display device, thus all team-member are able to observe the same view and have the same base knowledge on which they can investigate. Both status views are private elements visualized on smart devices. In this way detailed information of single processes are removed from the public screen in order to not overwhelm the user with unnecessary information or even occlude more relevant information beneath.

According to van der Veer and Van Welie [73] it is necessary to include descriptions of many aspects

of the task world, not just the tasks alone in order to design groupware systems. Their framework structures task models. Task models for complex situations have three different aspects: agents, work, and situation. These aspects are further decomposed into five main foci, as described in [74]:

- Agents: Personified instances performing tasks
- Roles: Agents perform roles in role-based activities.
- Activities: Sub-tasks performed to reach a goal
- Objects: Artifacts shared among agents
- Events: Triggering relevant changes of task state

Accordingly, each focus describes the task world from a different viewpoint; tasks have specific relationships. For the design of the task-supporting tools, designers can read and design from different angles, thus assuring consistency and completeness. An overview of the aspects and relations involved is sketched in Figure 4:

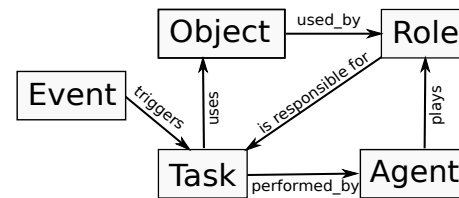


Fig. 4 Ontology of task world models [74].

As powerful communication technology has become increasingly pervasive, collaboration between people has moved in the direction of computer-supported cooperative work, where computers are now additional actors in collaborative processes. Roles can be exchanged easily between actors, and activities can be delegated to systems [43].

To cover the design of collaborative task models and a supporting system, activities to be performed must be clearly defined. Who is performing what activities? What objects are needed? How should one represent the information to user-groups? How can one enable interaction with the systems? What are the dependencies involved? These are the most important questions that one must answer.

It is necessary to have a clear understanding of the requirements for the intended system. Services, users, environment, and associated constraints, for example, must be defined and connected. Users in the system are actors performing tasks using the task world ontology. One must include dedicated task models into the system together with rules and rights of data access and functionalities. Participants must be able to choose profiles that are connected with tasks when they

register smart devices in the main system. Users can coordinate their activities and assign tasks (R1). The implemented visualization and interaction techniques are adequately realized for different scaled devices providing differing ranges of capabilities. These procure, that participants are enable to perform activities (R3). Setting up the co-located environment as depicted above, team-members share the same location and make use of a combined shared viewport on a large display device. Team members can determine others' activities (R5) and are self-evidently enabled to communicate with each other (R2) in a natural manner. However, the postulated requirements R4 (Member can switch between collaboration styles), R6 (Member can understand the impact of changes made), and R7 (Member can make adjustments based on impact) are not ensured and will be tackled with the distribution of the user interface across different scaled devices.

4.2 Distribution of User Interface Capabilities across Devices

Dividing the simulation view in order to enable Overview and detail techniques in one viewport is not sufficient. First, virtual reality is used for the simulation to generate realistic images and imagination of the process. Splitting the view in two parts would decrease the level of immersion and the perception of being physically present in a non-physical world [57]. Second, positioning a second view in the simulation view leads to a reduction of the visualization area.

Mobile devices enable the implementation of many interaction metaphors, leading to more natural and intuitive interaction. Tablet computers as well as smart phones offer input capabilities due to touch input mechanisms and other sensors, and we can use them as secondary displays. Tablet computers offer a big screen compared to smart phones, where the status update and status report views can be juxtaposed in a split-view. A split-view is not sufficient on a small display like those on smart phones. We use a smart watch to re-distribute the status update view, while the status report view is shown on the smart phone, as depicted in Figure 5. Locating the status update view on a smart watch is analogous to the usage of a usual watch, where users capture time information.

Simulation view. The simulation view provides an overview of the complete data and combines individual task model visualizations in one view. The simulation view holds a virtual reality application in which realistic images and depictions of the processes are generated. Virtual reality leads to a high level of immersion

and the perception of being physically present in a non-physical world [57]. The purpose of virtual reality is the facilitation of reception and understanding of complex data due to simulation and visualization of the data in a real-world perspective [11]. Users experience and observe the scene “from inside” and are able to concentrate their attention on the task exclusively [9].

Status update view. The status update view provides an overview of the task performance/progress at a glance. To overcome the limited display size of the smart watch, we use a glyph-based visualization. This visual design is a commonly used technique, where data is represented by a collection of glyphs. The data set is typically multivariate. Related work is for example performed by Steiger et al. [69], who described zoomable glyphs. Viewed from a distance, glyphs are recognizable in shape in color; zooming in brings out the information captured by each glyph at detail level. Relationships between variables and explanations of a glyph's appearance can be seen. A glyph-based visualization on mobile devices for the notional analysis in sport was successfully used to establish collaboration between different analysts on event-based visualization [44]. The major strength of glyph-based visualization is this: Patterns of multivariate data can be easily perceived in the context of a spatial relationship [8]. According to Borgo et al. [8] a glyph is defined as: “*a small independent visual object that depicts attributes of a data record*”. Characterization of those visual objects can be done as follows:

- Glyphs are discretely placed in a display space.
- Glyphs are a type of visual sign but differ in form.

Next to the number of dimensions that will be represented by the glyphs themselves, the placement of glyphs (positioning inside the display area; relationships between glyphs) on a display indicates significant information regarding the data values [76]. Taking into account the interdependencies of task models implying a multivariate nature of the data in the presence of only having limited screen size available, glyph-based visualization matches the requirements of a visualization technique for our setting and its conditions.

Status report view. Touch input applied to a glyph in the status update view on the smart watch opens a dedicated status report view on the smart phone. While the status update view is designed to provide a quick overview of the task performance, the status report view is designed to provide detailed information about the ongoing processes and the data visually presented via glyphs. Glyphs are repetitively visualized in the status report view to reflect the affilia-

tion of both views. Detailed information is presented as text together with graphical control elements. Interaction capability in form of adjustments to the underlying data is provided, which has a direct impact on the main application and the visual representation of the glyphs.

4.3 Prototype Implementation

Basic system. The IN2CO environment serves as foundation of the prototype’s implementation. The system’s architecture consists of the following modules as depicted in Figure 7:

- Smartdevice interface: It links smart devices and triggers the exchange of messages.
- Graphical user interfaces: They register smart devices with the environment.
- Basis module: It supports activities like parsing for import and export and creating annotations.
- Collaboration module: It triggers user registry, object distribution, data exchange and transaction handling.
- Application interface: It holds user-specific viewports, user roles, and a tool and functionality collection for the tasks/usable devices.
- Data storage: It collects all application-specific values with impact links between processes, and it contains all session logs for recording and recovery.

The sequence diagram shown in Figure 6 describes the interaction and order of interaction of the system’s modules. The graphical user interface (top layer) assists the user when choosing and aggregating the needed plug-ins and devices, triggering the *system registry* (Figure 7), the *user registry* (Figure 8), and finally program execution.

The system registry interface allows a user to choose and aggregate the needed application model and used devices. The corresponding system registry module links

all appropriated resources and plug-ins to the program and starts the user registry. This module associates roles, viewports, and rights with the user and connects the registered user/devices with the server. Technical details of the main system are provided in [61]. The main application, running in a CAVE system, communicates via WIFI [2] using TCP/IP protocols [23] with the mobile device application. The main application is initiated via the CAVE system, which starts the server and initiates message handling. The devices get connected to the server. Interaction with the main application is made possible via mobile devices that directly communicate with the main server through a local WIFI network. Task model-dedicated data and information are stored in a MySQL database [55], which is updated when changes of the main application are performed, caused by continuously incoming requests of mobile device applications, see Figure 9. The database is persistent, i.e., database tables are created once and can be used in each collaboration session without prior creation of the database structures. The database manager MySQL includes an InnoDB storage engine as consistency model that adheres closely to the ACID model [20]. The ACID model describes the four properties atomicity, consistency, isolation, and durability, used as major guarantees of transaction paradigms within database applications. Data is not corrupted and results are not distorted by exceptional conditions, such as software crashes or hardware malfunctions. Consistency checking and crash recovery mechanisms are included, and data reliability for several users is ensured.

Communication smart devices and basis. Once connected to the main application, the smart devices send dedicated messages to the main application. A message-handler included in the main application, transfers the incoming messages from the smart devices to so-called tools. Those tools are included within the se-

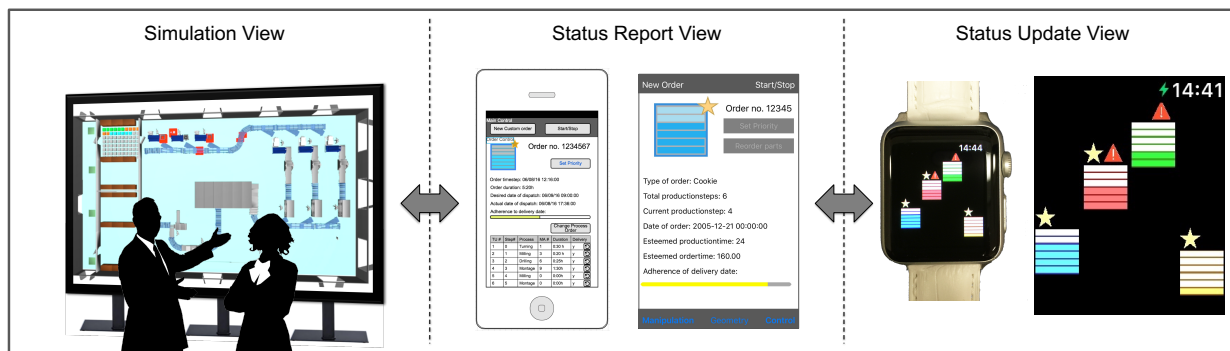


Fig. 5 Collaboration setup: Simulation view containing a virtual reality application as public display; status update view enables monitoring of own process; status report view provides explanations of performance and interaction with the system.

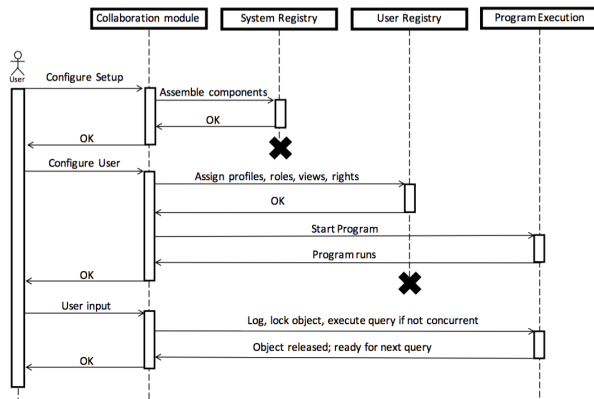


Fig. 6 Sequence diagram of framework.

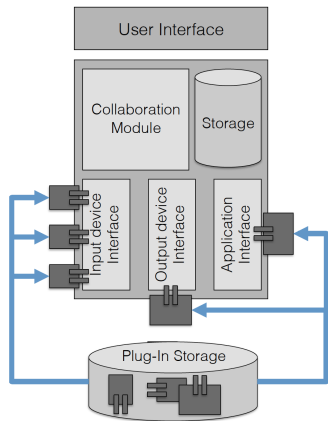


Fig. 7 Main components: User Interface, toolkit, plug-in storage. The system registry links all appropriated resources and plug-ins to the program.

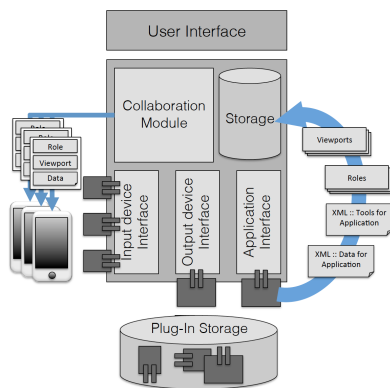


Fig. 8 User registry - association of roles, viewports, and rights.

lected task model and provide functionalities and visual representation for both the simulation view and the clients. Initially, those tools are created and listen

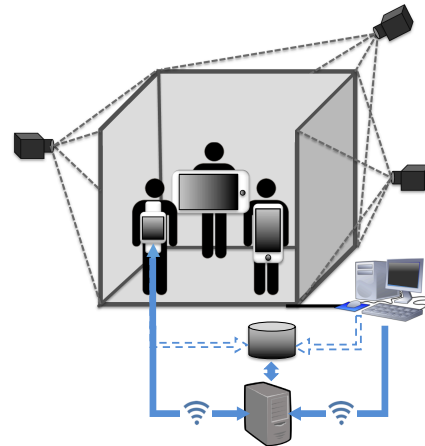


Fig. 9 Setup and communication channels of collaboration framework.

for incoming messages, which triggers the functionality in the main application. Both the main application and the smart devices are directly connected to the database system and can trigger update of the database entities. This is crucial to not only update the visual representation in the simulation view but also to update the underlying structures and information for all other clients (smart devices). In our prototypical implementation, we provide connection, message exchanges, and task dedicated interaction and visualization techniques on five different scaled smart devices: iPhone5 (3,5" [31], iPhone 6 + (4,7" [32], iPad mini (7,9" [30], iPad2 (9,7" [29], and Apple Watch Sport [28]. The implementation on client side is done with native user elements and integrated web-views that allow platform independence.

Communication across smart devices. The watch used as distributed user interface is no independent client in the system, but the enlargement of the smart phone. As independent device, the watch does not represent enough information and does not directly communicate with the main application or the database system. However, this small scaled device is used additionally to enhance the interaction capabilities by including the performance of natural and intuitive gestures in the VR world based on arm-movements, as described in [62]. Distributing the status-update view to smart watches leads to several challenges. The prototypical implementation is performed on iOS devices, specifically on an Apple Watch Sport 38mm and Apple iPhone5. All task model-dedicated data and information are requested by the phone and continuously sent to the watch via Bluetooth connection [21]. The watch cannot access the database directly, leading to a high

overload of communication between the devices. Moreover, the display size of the watch is merely 272x340 pixels with 326dpi. The visualization of the information has to be minimized and all information must be captured at a glance. So far, direct exchange and communication between other clients is not completely realized, but will be included in further work.

5 Case Study – Event-driven Production Control and Factory Planning

5.1 Task Definition

Using a real-world scenario in production control, we illustrate the impact and interdisciplinary collaboration requirements, and discuss the implemented system components in detail.

Companies in high-wage countries must be highly efficient and innovative in manufacturing to remain competitive. Market competition increased through opening economic regions, e.g., in Eastern Europe or the Far East [3]. Mass products are offered for lower prices by countries in these regions. Companies in high-wage countries have to adapt in this evolving competitive setting, as they might become obsolete and be destroyed otherwise [12]. The domain of factory planning tackles these problems. A strategy to address this challenge is a shifted focus on highly specialized, customized products [3]. By moving to such a more individual customer-oriented production setting a company follows a make-to-order processing paradigm, where the production of a parts starts with the arrival of an order [33].

Nevertheless, customers still require products with high quality to be offered at a low price and a short delivery time. For a company this means that it must be flexible in offering individual products in a short amount of time while still remaining economical [41]. High flexibility within a production process makes necessary a concentration on production planning and control. A production process can be planned but only in rare cases the production is planned. Events that lead to such a deviation within the production process, are, for example, machine breakdown, missing parts or manufacturing of unusable parts/products [42]. In such circumstances, a planned optimal production process cannot be realized.

A company must react as quickly as possible to events (new customer orders or deviation from a planned optimal production process) to minimally deviate from a stated production plan. Such events cause a gap between the current state of a production process and the planned state of production. An alternative production plan must be created in this situation, leading to

the concept of “production control.” Production control regulates such conditions within the order processing, i.e., it determines the sequence of sub-processes that should be executed [63]. Should changes in the production plan arise, the production control will be responsible to carry out modifications. The first step identifies the current state of production. This step must be executed rapidly to minimize, as much as possible, the deviation between the planned and the actually observed current state of production [41]. To react rapidly a continuous view of the state of production is necessary, and all related information must be continually recorded and available. Such a set-up makes it possible to adapt a production plan quickly once a disturbing event is recognized. Production control must be an automatic process to detect undesired events and adapt the production plan accordingly.

To meet these requirements an event-driven production control (EDPC) was developed. The EDPC system uses an extended bill of materials in order to shorten the reaction time in case of occurrence of an (undesired) event. The bill of materials is extended with additional information for each part (e.g., required production station, size, mass, set-up time and production time). This extension makes it possible to store information within the bill of materials that is necessary for the production control. The system approach taken by Kasakow et al. [37] uses the production of a turbo charger as an example. By placing an order, the EDPC uses the content of the order and creates an appropriate extended bill of materials. Necessary production tasks to be done to satisfy the customer order derive from this bill of materials. According to Kasakow et al. [37] have shown that the EDPC can realize an automated production control. It is possible to derive all necessary actions to be taken within a production process (e.g., creation of production orders, arrangement of the production sequence), based on a customer’s bill of materials and the information of the current state of production.

A disadvantage of this EDPC is the acceptance by users. The acceptance of a system depends on the experience of a user. Here, a user is a planner of a production. The more experience a planner has with automation errors, the more she/he wants to supervise and monitor the system. But a planner trusts automation only when it is fully reliable. The reliability of an automated system leads to the required or desirable amount of supervision and monitoring effort. A lack of reliability reduces the acceptance of an automated system [75]. In case of misbehavior or breakdown of the automated EDPC, a planner needs to have access to an uncomplicated solution to interfere with this production control to ensure a smooth operation of the production process

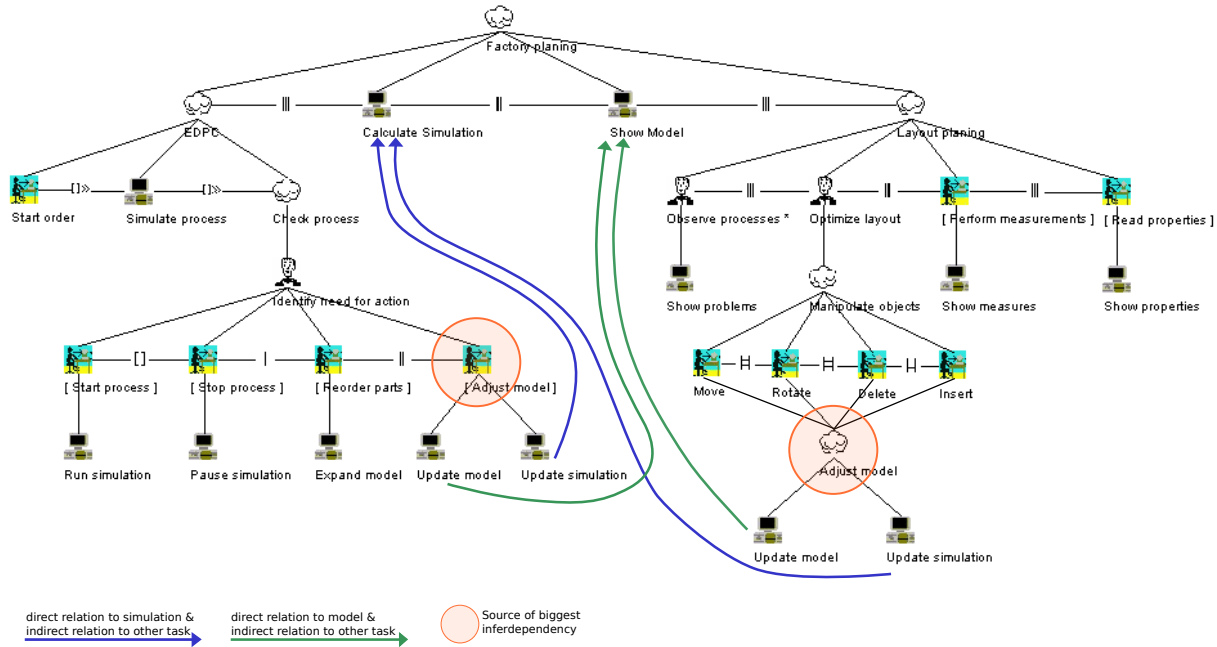


Fig. 10 CTT diagram of combined task model representing task structure and interdependencies.

or to carry out tasks that are not part of the EDPC (e.g., implementation of a rushed order) [5], [37]. An ideal system allows the planner to monitor the status of the current production and provides all necessary information to optimize production, to ensure a smooth operation of production.

Task model interdependency. Factory planning, as domain field of EDPC, is characterized by the parallel consideration of multiple aspects such as production resources, production process and technology, and products while anticipating uncertainty and future developments over the factory life-cycle [72]. These aspects usually result in different partial-models with specific information content (e.g., layout model, process model) and components of the factory (e.g., building, machinery, foundation, media), which need to be analyzed in combination. The different partial solutions are usually developed by various stakeholders, but typically interfere and require each other [66]. The major tasks regarding collaborative factory planning are [77]:

1. Assembling multiple, domain-specific points of view
2. Bilateral problem introduction
3. Joint discussion and integrated decision making

In order to implement a real-world collaboration process, two domain-specific tasks of EDPC and layout planning were chosen, both tasks being relevant for factory planning. Two different task models were created using the task world ontology introduced in [74],

included in the system: Factory layout planning and event-driven production control.

The concur task tree (CTT) diagram [24] depicts the simplification of both task models and their interdependency. It is shown in Figure 10¹. All shown tasks are further refined as task world models, formulated as tasks performed by an actor having a specific role. Tasks can use objects and trigger, or are triggered, by events, see Figure 4. The CTT diagram shown in Figure 10 shows a refined definition of the tasks based on the task world ontology. For better readability, the detailed and graphical notation of the task models, based on the task world ontology, is not covered in this paper. The CTT diagram depicts a simplification of the tasks that are performed for factory planning. Interdependencies are found in sub-tasks incorporating the combination of influence and dependence between two elements, one- or bi-directional. Adjusting the model as sub-task in EDPC changes the underlying dataset for EDPC and factory layout planning, either due to changes in the simulation or manipulation of the model itself. Manipulating objects in the course of a factory layout planning sub-task has a direct influence on the production flow and simulation within EDPC, clarifying the interdependen-

¹ Usually, in a CTT several nodes are not connected to the same child node. For better readability, the user sub-tasks *Move*, *Rotate*, *Delete* and *Insert* are connected to the grouped task *Adjust Model*. More precisely, each of the user sub-tasks should have a connection to the system activities *Update model* and *Update simulation*.

dencies between both tasks. Layout planning concerns the task of "deciding on the best physical arrangement of all resources that consume space within a facility," which is performed when there is a change in the arrangement of resources [59]. Improvements of the overall production performance can be achieved concerning the following parameters:

- Time
- Energy
- Cost
- Organization
- Efficiency
- Productivity
- Information flow
- Material flow

EDPC provides information regarding the following parameters and provides explanations related to their interplay:

- Quality
- Time
- Cost
- Energy consumption
- Utilization of tools and machines
- Optimal path of material flow
- Factory layout suggestions

Other processes and tasks also influence these parameters. Impact on material flow and overall production performance results when changes of the order are performed, but also between material flow and layout changes (two differing task models).

The dynamic model in the existing prototype combines the two task models of event-driven production control and factory layout planning. Based on the listing of improvement parameters of both tasks above, it is easy to see, that both tasks have several interdependencies between each other, e.g., changing machine positions and paths has an direct impact on material flow, production time, transportation time, waiting time, and path utilization. Below we describe the information/data that is displayed in each of the views and devices for this case study.

5.2 User roles and rights

The simulation view of the framework provides an overview of the underlying manufacturing system, consisting of the building, storage areas, machines, human resources, and conveyors, see Figure 3, right. The smart-devices are used to control the scene and execute the functionalities in the large screen setup. The following functionalities are currently implemented for a desktop and

CAVE setup, and for mobile devices. Supported functionalities include:

- Manipulation: rotate, pan, and zoom of single objects
- Navigation: rotate, pan, and zoom of the whole model; first-person view and navigation; selection of pre-defined views; hide/show object-groups;
- Examination: measurement of distances and dimensions; textual output of object-information
- User feedback: highlighting and vibration
- Collaborative features: making annotations, inserting comments, marking areas, and creating a visual snapshot

In order to identify different users, the following roles are defined and associated with the implemented functionalities. Each functionality is associated with exactly one role, implying existence of distinct roles. One or several roles can be associated with one user/actor.

- Factory layout planner, basic
 - measurement of distances, dimensions
 - textual output of machine and facility
 - creation and removal of machines
- EDPC, basic
 - start new product order
 - start/stop production simulation
 - re-order machining parts
- Manipulator
 - rotate, pan, and zoom (single objects)
 - hide/show object groups
- Collaborator
 - making annotations
 - creating comments
 - marking areas
 - creating visual snapshot
- Navigator
 - rotate, pan, and zoom (whole model)
 - first-person view and navigation
 - selection of pre-defined views

5.3 Distributed user interface

Simulation view. The public large display devices present the simulation view. This view shows a manufacturing system in the context of dedicated work areas, machines, workstations, and transportation paths, for example. Each object in the scene can be selected, moved, rotated, duplicated, or removed by a user. A selected object is highlighted in the user's color and locked for other users until it has been released. Coloring the selected object provides awareness of the various users. Locking an object ensures that the same object

cannot be manipulated by several users at the same time.

In the simulation view each order is associated with several transport units. Each transport unit depicts one production step of the order, and it is visualized by cubes color-coded per production order (red), see Figure 11. The transport units start in the “commission site” where they load materials and needed production parts, moving to the machines where the production process takes place, and finally coming back to the commission site delivering the final products. The simula-

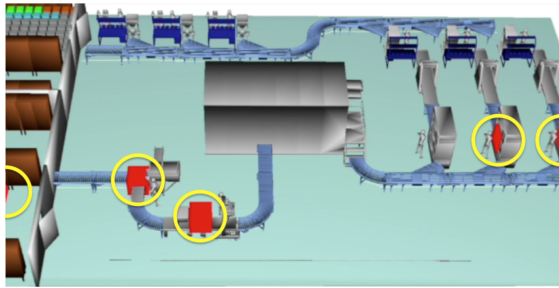


Fig. 11 Transport units in virtual manufacturing system are dedicated to one product order.

tion view visualizes material flow of the production, and it also provides hints about transportation time, transportation paths, wait times, and machine capacity, supporting the modeling and simulation of material flow. The simulation explains path utilization and suggests possible layout changes. Users can highlight specific areas and make annotations in the user’s color. Those markings and annotations can be shown and hidden in the visualization view.

Status update view. This view in the task model of EDCP must quickly provide an overview of the production progress, indicating potential problems and status of production goals. While the simulation view sheds light on overall production performance, the status update view shows explanations concerning a single order’s production progress.

It is important that a user has insight into the status of the various status points to be achieved, in our example related to the delivery date for orders and explanations regarding the goal achievement. In the status update view one glyph represents the data set associated with one order of a customer. Figure 12 provides an overview of the glyph design.

One glyph represents one custom order and associated transport units. The color of the glyph is identical with that of the associated transport units in the simulation view. The position of the glyph indicates

two dimensions of the data set. Positioning along the x-axis reflects the time-stamp of posted orders in sequence in analogy to the reading direction from left to right. Positioning along the y-axis shows status of the planned delivery date. A glyph positioned near the bottom represents the case where the planned delivery date is achieved, based on the accumulated performance; a glyph positioned near the top indicates that is not possible to satisfy the projected delivery. This positioning is based on the analogy to read from top to bottom. Glyphs positioned at the top are critical for the process and have to be detected quickly.

Different shapes of the glyphs reflect different product types. In analogy to the graphical control element status bar, stacked-up bar layers, representing transport units of the order, depict the glyph. The number of layers represents the number of production steps that have to be performed to produce the corresponding product type. Filled bars show the number of steps that have been performed, while unfilled bars show the number of process steps still to be performed.

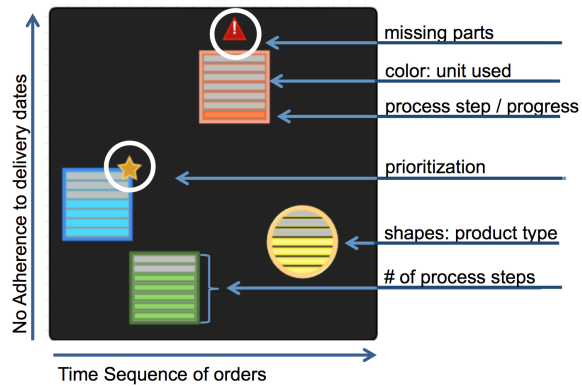


Fig. 12 Glyph design for event-driven production control.

A production step is only feasible when needed parts and material are in stock and ready to be collected in the commission site. Missing parts lead to wait time in the production of the order. This condition is indicated with a red triangle on top of a glyph. In general, orders are processed in the order of them being placed. The first posted order has the first position in the machining process. The transport unit asks for the machining position after picking up materials and production parts. Missing parts or long transportation paths can change this order, which can have an impact on wait time again. To avoid this situation and to process new urgent postings, orders can be prioritized, which defines the transport units associated with that order for the first machining positions. A small star on the top of a

glyph indicates the prioritization. Awareness indication of other users is not integrated in the status update view to reduce information over-load and, instead, provide a clear overview of the ongoing production processes.

Status report view. A tap on a glyph in the status update view opens the corresponding status report view on the smart phone application, (see Figure 13). The status report view, in contrast to the status up-



Fig. 13 Touch input applied to the status update view on the smart watch opens a dedicated status report view on the smart phone.

date view, provides detailed information about one individual order, and potential undesirable behavior is described together with suggested possible adjustments.

The view is integrated in the IN2CO interaction application. The user interface is divided into three parts: Main control, order control, and view tab bar, see Figure 14. The main control contains buttons allowing one to post a new order and start/stop the simulation of the material flow in the simulation view. The view tab bar contains bar buttons to the existing view widgets. When the status report is not visible, no updates of the report are performed to save computational resources. The status update view, however, is continuously updated and rendered. The order control constitutes the main part of the report. The glyph is visualized in the left-upper corner to reflect the relationship to the status update and simulation views. Next to the tex-

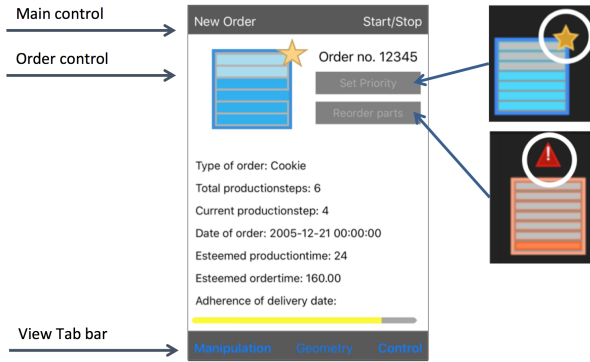


Fig. 14 Status report and dashboard application contains referring custom order glyph, detailed information of the order production, and graphical control elements.

tual representation of the data set, a color-coded status bar is rendered as a graphical control element, located at the bottom of the interface. Instead of visualizing the progress of an order, the status bar represents the predicted probability of achieving the planned delivery date, predicted based on the accumulated performance. This is done similarly to the positioning of glyphs along axes in the status update view. The color of a status bar indicates the degree to what a desired state has been achieved.

In this view it could be of interest to provide information about users observing the same production process, to avoid redundant re-ordering of parts or provide helpful information for discussion. However, buttons for re-ordering parts and prioritization are enabled when already selected, to ensure that the stock is filled and/or the production process is prioritized. We believe that it might be desirable to highlight single lines in the status report view, which could be synchronized with other users and devices to indicate potential bottlenecks or communicate interesting information.

5.4 Distributed Collaboration

A possible collaboration session within factory planning is sketched following. Two users are jointly located in the CAVE-System and register with the system, as depicted in Figure 15:

One user is assigned to the task of layout planning (further called planner) and associated the roles of *Factory layout planner basic*, *Manipulator*, and *Collaborator*; the other one is assigned to the task of production control (further called controller) and is associated with the roles *EDPC basic*, *Manipulator*, *Collaborator*, and *Navigator* (R1). The joint goal is to find a layout of the facilities interior in which the resources are opti-

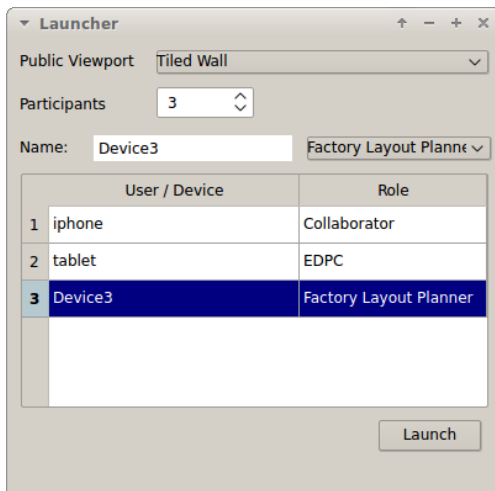


Fig. 15 Default graphical user interface facilitates the registration of users and devices, associated with roles.

mal arranged and the production program is optimized. While the planner has information of spacial constraints and environmental conditions like illumination, temperature, etc., the controller has information about the machining sequence, production program, and inventory. For both users it is important to find the shortest paths between the single machining steps. Both users are able to move machines and the controller can initiate product orders further more (R3). The orders are visualized in the simulation view, which enables the user to track the material flow with the current layout. It can be easily identified if machines have to be rotated or if machines are not used and should be probably removed. At any time, production parts can be missing, leading to waiting-phases of the product in the machine. In the simulation view it can just be recognized, that a product is in the machine, but not if it is getting produced. The controller gets a haptic feedback on the watch, if parts are missing. With a quick sight on the watch, he can identify problems in the processing and reorder parts to counteract the potential delay (R4)(R7). Both users could detect, that waiting products in machines blocks the machine. If following orders are delayed too, the machine describes a bottleneck initiating a demand of action. In joint discussions, in which the ideas and thoughts of both users are considered (R2) (R4), they decide to install a waiting area next to "bottleneck-machines" and an alternative conveyor system, so that missing parts can delivered to the machine. The planner performs the installation of the new resources, which is commented and observed by the controller (R5) who can track the impacts of this changes as positive or negative influences on the production program on the watch (R6).

6 Expert Evaluation and Discussion

In order to verify the system we performed an expert evaluation in two steps. First, we conducted an assessment based on the property checklist method [36] and afterwards performed the method of Assessment of experience [18] in close collaboration with three domain experts.

6.1 Property checklist

The method of property checklist is a structured way to do an evaluation, in which the expert goes through a checklist of design goals for different product properties. In our case, the product properties correspond to the seven postulated requirements in chapter 3. As described in section 5.4, all requirements are realized and the users are facilitated to perform collaborative work in an efficient and natural manner.

- ✓ R1 Member can coordinate activities.
- ✓ R2 Member can communicate with each other.
- ✓ R3 Member can perform activities.
- ✓ R4 Member can switch between collaboration styles.
- ✓ R5 Member can determine others' activities.
- ✓ R6 Member can understand the impact of changes made.
- ✓ R7 Member can make adjustments based on impact.

6.2 Qualitative assessment

Nestler et al. [54] proposed a qualitative assessment approach used when a comparative evaluation cannot be performed, and when baseline efficiency values/rates cannot be used to benchmark the system. No other existing setups could serve for benchmarking when less functionality and interaction capabilities are provided. If one were to compare the usability of the proposed multi-modal interface with the usability of the earlier EDPC system, the new approach would be inferior. Therefore, the evaluation method of Nestler et al. is a viable alternative, based on a reliable questionnaire for assessing technology acceptance. According to Nestler et al., general considerations useful for qualitative usability evaluations are: (1) Most usability problems are detected with three to five subjects. (2) It is unlikely that additional subjects reveal new information. (3) Most severe usability problems are detected by the first few subjects. Due to the limited number of available experts in our domain, we involved three participants for the qualitative usability evaluation.

In close collaboration with three EDPC experts, we performed an experimental user study. The experts performed the collaboration session as described in section 5.4. Each expert performed the experiment three times

- two times with two participants involved, and once with three participants involved, leading to four experiments. The number of participants was not sufficient to gather significant results concerning efficiency or effectiveness of the prototype. Nevertheless, we were able to gain insight into general users' experiences with our system. We measured a general usability score U as introduced in [54]. After performing the experiments, we solicited expert feedback using questionnaires as proposed in [54] concerning four categories, see [18]: (1) ease-of-use; (2) user satisfaction; (3) usefulness; and (4) intention to use of our system. For the qualitative evaluation the interviewer used open-ended questions and did not interrupt the subject. The aim of the interview was to discuss all perceived problems with the subject in order to detect usability issues.

6.3 Quantitative Assessment

According to [49] the qualitative results of the assessment provide a performance quantification basis, resulting in a scalar usability value U . The usability categories are adjusted on a three-point scale: (a) positive comment (1.0); (b) neutral comment (0.5); and (c) negative comment (0.0). The mean of the values is calculated. We obtain a quantitative rating of all categories on a scale from 0.0 to 1.0. Moreover, the usability categories are weighted, summing up to 100, to express the importance of a category in a specific application. Formally, the usability score is calculated by multiplying all weights $w(s)$ with the quantitative scores $v(s)$, with the value of U being between 0.0 and 1.0:

$$U = \sum v(s)w(s) = \sum U_c \frac{w_c}{100} \quad (1)$$

6.4 Results

During the experiment execution, a brief look at the watch provided insight into the production process, allowing one to determine whether actions were needed to interfere or not. At the same time, the team-members could observe the material flow in the simulation view, detect potential bottlenecks, and make suggestions for an optimized factory layout. As both task models of EDPC and factory layout planning are integrated in the prototype, the experts were able to adjust the layout and track the impact on production on the watch (at a glance), and establish priorities and initiate re-ordering of parts with the smart phone application. It was simple to stop/pause the simulation view and balance the current outcome. Furthermore, the experts could leave the physical setup and track changes on the

watch for monitoring purposes due to the WIFI connection with the server and database system.

(1) Ease-of-use. Since our experts were familiar with the general control capabilities of smart devices they could focus on the task instead of the control mechanisms. The implemented user interface elements are easy to learn and match to the dedicated functionalities. Using the watch to get immediate feedback of the progress and potential delays was stated as intuitive. It was easy to recognize by the users if they had to intervene in the process or if they could focus on the group-work. Switching between group- and individuals' tasks is performed smoothly and without interruptions of others performances. Only the interaction capabilities with the smartphone could be enhanced with more intuitive techniques.

(2) User satisfaction. Each user in the evaluation could control the scene within the restrictions of his task and was able to actively participate in the collaboration. The users did not feel strained using the collaboration environment and could focus on solving their tasks. On the contrary, the users took delight in using the system and felt immersed in the scene. Overall, each user was satisfied, which increased the motivation of the users to perform the tasks and the collaboration in the team.

(3) Usefulness. The system supported the users to perform both group and individuals' tasks of both use cases and facilitated the recognition of impacts and the cooperation between the users. We have identified several questions that could be answered by using our system in this specific application domain. Example questions are:

- How many orders do exist?
- Where do you spot potential hazards?
- How many production steps does an order have?
- How many missing parts do exist?
- Which order should be prioritized?
- How many priorities exist?
- Do orders have the same number of production steps?
- How many steps have been performed already?
- Which order was placed most recently?

The usefulness of the system for the specific use case and for collaboration activities is verified.

(4) Intention-to-use. All experts expressed their desire to use our setup in the future, recognizing our system's value for a planner when checking on production status and intervening to optimize it. One participant

provided a neutral comment, stating that the setup was too sophisticated for the problems he was concerned with.

The resulting quantitative scores and weights leading to a usability value $U = 0.902$ are summarized in Table 1:

Category	(1)	(2)	(3)	(4)
w(s)	0.25	0.25	0.25	0.25
v(s)	0.833	1.0	1.0	0.833
w(s) v(s)	0.201	0.25	0.25	0.201
Σ	0.902			

Table 1 Weights and scoring of usability categories to calculate usability score U of the system.

The qualitative and quantitative assessment results shown are very good. It is possible to conclude that our framework leads to a more efficient and successful collaboration in the case of factory planning. Due to the modular concept of our framework, different scenarios and task models can be integrated, facilitating collaborative work and decision-making in other domains.

7 Conclusions and Future Work

We have described and prototyped a general-purpose framework that can be adapted to the specific requirements of a specific application to support efficient collaborative design, simulation or visual data analysis—done simultaneously by distributed or co-located teams using diverse mobile devices. The devices provide three views of the data to be processed collaboratively: (1) a simulation view; (2) a status report view; and (3) a status update view. These views serve the purpose of providing overview, detail and performance views. Our approach goes beyond the known characteristics of existing “Overview-plus-detail” techniques. The watch analogy employed by us provides a user with the information explaining the impact of user-induced changes made to a production process, for example, in a natural and intuitive manner. Comparable frameworks do not support an active manipulation of a simulation considering different task models and interdependencies. When geographically separated from the collaboration system, users can monitor processes and actively perform changes to them to improve process progression. Visualizations provide high-level insight into a process’ status, and the status report view leads to a deeper understanding of the effects resulting from optimizing the production process. The performance view shown

on the watch display explains whether one should take action in an ongoing process or not. We have implemented an event-driven production control (EDPC) application as case study and successfully demonstrated the use and advantages of our framework for a specific example, achieving an overall usability value of 0.902.

Concerning future work, we want to perform a broad user study to evaluate glyph design; the accuracy and usability of the visualizations used on the watch; and the added value resulting from using the watch in a collaborative team setup. We will also indicate on the watch who has performed specific changes using haptic feedback and pop-up symbols.

References

1. Craig Anslow, Stuart Marshall, James Noble, and Robert Biddle. SourceVis: Collaborative software visualization for co-located environments. *2013 1st IEEE Working Conference on Software Visualization - Proceedings of VISSOFT 2013*, 2013.
2. IEEE Standards Association et al. 802.11-2012-ieee standard for information technology–telecommunications and information exchange between systems local and metropolitan area networks—specific requirements part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications. *IEEE Std*, 802, 2012.
3. Thomas Bauernhansl, Michael Ten Hompel, and Birgit Vogel-Heuser. *Industrie 4.0 in Produktion, Automatisierung und Logistik: Anwendung: Technologien. Migration*. Springer-Verlag, 2014.
4. Sebastian Baumgärtner, Achim Ebert, Matthias Deller, and Stefan Agne. 2d meets 3d: a human-centered interface for visual data exploration. In *CHI’07 Extended Abstracts on Human Factors in Computing Systems*, pages 2273–2278. ACM, 2007.
5. Bernd Beckert, Daniela Buschak, Birgit Graf IPA, and Martin Hägele. *Automatisierung und robotik-systeme. Studien zum deutschen Innovationssystem Nr. 11-2016*, 2016.
6. Charles C Bonwell and Tracey E Sutherland. The active learning continuum: Choosing activities to engage students in the classroom. *New directions for teaching and learning*, 1996(67):3–16, 1996.
7. J. Borchers, M. Ringel, J. Tyler, and A. Fox. Stanford interactive workspaces: A framework for physical and graphical user interface prototyping. *Wireless Commun.*, 9(6):64–69, December 2002.
8. Rita Borgo, Johannes Kehler, David H Chung, Eamonn Maguire, Robert S Laramee, Helwig Hauser, Matthew Ward, and Min Chen. Glyph-based visualization: Foundations, design guidelines, techniques and applications. *Eurographics State of the Art Reports*, pages 39–63, 2013.
9. William Bricken. Virtual reality: Directions of growth notes from the siggraph’90 panel. *Virtual Reality: Directions of Growth*, 1990.
10. Patrick JP Brown. Process-oriented guided-inquiry learning in an introductory anatomy and physiology course with a diverse student population. *Advances in physiology education*, 34(3):150–155, 2010.
11. Steve Bryson, Steven K Feiner, Frederick P Brooks Jr, Philip Hubbard, Randy Pausch, and Andries van Dam.

- Research frontiers in virtual reality. In *Proceedings of the 21st annual conference on Computer graphics and interactive techniques*, pages 473–474. ACM, 1994.
12. Hans-Jörg Bullinger, Dieter Spath, Hans-Jürgen Warnecke, and Engelbert Westkämper. *Handbuch unternehmensorganisation: strategien, planung, umsetzung*. Springer-Verlag, 2009.
 13. L Bunch, M Breedy, JM Bradshaw, M Carvalho, D Danks, and N Suri. Flexible automated monitoring and notification for complex processes using karmen. In *Proceedings. 2005 IEEE Networking, Sensing and Control, 2005.*, pages 443–448. IEEE, 2005.
 14. Jane Burdett and Brianne Hastie. Predicting satisfaction with group work assignments. *Journal of University Teaching and Learning Practice*, 6(1):61–71, 2009.
 15. Stefano Burigat and Luca Chittaro. Visualizing references to off-screen content on mobile devices: A comparison of arrows, wedge, and overview+ detail. *Interacting with Computers*, 23(2):156–166, 2011.
 16. Stefano Burigat and Luca Chittaro. On the effectiveness of overview+ detail visualization on mobile devices. *Personal and ubiquitous computing*, 17(2):371–385, 2013.
 17. Thorsten Büring, Jens Gerken, and Harald Reiterer. Usability of overview-supported zooming on small screens with regard to individual differences in spatial ability. In *Proceedings of the working conference on Advanced visual interfaces*, pages 233–240. ACM, 2006.
 18. Emanuelle Cavalcante, Luis Rivero, and Tayana Conte. Evaluating the feasibility of max: A method using cards and a board for assessing the post-use ux. *International Journal of Software Engineering and Knowledge Engineering*, 25(09n10):1759–1764, 2015.
 19. Luca Chittaro. Visualizing information on mobile devices. *Computer*, 39(3):40–45, 2006.
 20. Oracle Corporation. Mysql :: Mysql 5.7 reference manual :: 15.2 innodb and the acid model. (Accessed on 04/07/2017).
 21. Myra Dideles. Bluetooth: A technical overview. *Crossroads*, 9(4):11–18, June 2003.
 22. Matthias Finke, Nima Kaviani, Ivy Wang, Vincent Tsao, Sidney Fels, and Rodger Lea. Investigating distributed user interfaces across interactive large displays and mobile devices. In *Proceedings of the International Conference on Advanced Visual Interfaces, AVI '10*, pages 413–413, New York, NY, USA, 2010. ACM.
 23. Behrouz A. Forouzan. *TCP/IP Protocol Suite*. McGraw-Hill, Inc., New York, NY, USA, 2 edition, 2002.
 24. W3C Working Group. Concur task trees ctt, 11 2012. (undefined 5/9/2017 15:0).
 25. Charles C Heckscher. *The collaborative enterprise: managing speed and complexity in knowledge-based businesses*. Yale University Press, 2007.
 26. Cynthia Lake Howell. Student perceptions of learner-centered education. *Online Submission*, 2006.
 27. Arief Ernst Hühn, Vassilis-Javed Khan, Andrés Lucero, and Paul Ketelaar. On the use of virtual environments for the evaluation of location-based applications. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 2569–2578. ACM, 2012.
 28. Apple Inc. Apple watch (1st generation) - technical specifications. (Accessed on 04/07/2017).
 29. Apple Inc. ipad 2 - technical specifications. (Accessed on 04/18/2017).
 30. Apple Inc. ipad mini - technical specifications. (Accessed on 04/18/2017).
 31. Apple Inc. iphone 5 - technical specifications. (Accessed on 04/07/2017).
 32. Apple Inc. iphone 6 plus - technical specifications. (Accessed on 04/18/2017).
 33. Herbert Jodlbauer. *Produktionsoptimierung*. Springer Science & Business, 2008.
 34. Matthew Johnson, Jeffrey M Bradshaw, Paul J Feltoovich, Catholijn M Jonker, Birna Van Riemsdijk, and Maarten Sierhuis. The fundamental principle of coactive design: Interdependence must shape autonomy. In *Coordination, organizations, institutions, and norms in agent systems VI*, pages 172–191. Springer, 2011.
 35. Matthew Johnson, Jeffrey M Bradshaw, Paul J Feltoovich, Catholijn M Jonker, M Birna Van Riemsdijk, and Maarten Sierhuis. Coactive design: Designing support for interdependence in joint activity. *Journal of Human-Robot Interaction*, 3 (1), 2014, 2014.
 36. Patrick W Jordan. *Designing pleasurable products: An introduction to the new human factors*. CRC press, 2002.
 37. Georg Kasakow and Jan C Aurich. Realising digital connectivity by using interdependencies within a production process. *Procedia CIRP*, 52:80–83, 2016.
 38. Daniel F Keefe, Ankit Gupta, Daniel Feldman, John V Carlis, Susi Krehbiel Keefe, and Timothy J Griffin. Scaling up multi-touch selection and querying: Interfaces and applications for combining mobile multi-touch input with large-scale visualization displays. *International Journal of Human-Computer Studies*, 70(10):703–713, 2012.
 39. Kiyoshi Kiyokawa. Human communication in collaborative augmented reality systems. In *Emerging Technologies of Augmented Reality: Interfaces and Design*, pages 236–260. IGI Global, 2007.
 40. Gary Klein, Paul J Feltoovich, Jeffrey M Bradshaw, and David D Woods. Common ground and coordination in joint activity. *Organizational simulation*, 53, 2005.
 41. Jürgen Kletti and Jochen Schumacher. *Die perfekte Produktion: Manufacturing excellence durch short interval technology (SIT)*. Springer-Verlag, 2011.
 42. Jrgen Kletti. *MES - Manufacturing Execution System*. Springer Berlin Heidelberg, 2015.
 43. Olga Kulyk, Dhaval Vyas, Onno Kubbe, Achim Ebert, et al. Task modeling for collaborative authoring. In *Proceedings of the 29th Annual European Conference on Cognitive Ergonomics*, pages 171–178. ACM, 2011.
 44. Philip A Legg, David HS Chung, Matthew L Parry, Mark W Jones, Rhys Long, Iwan W Griffiths, and Min Chen. Matchpad: Interactive glyph-based visualization for real-time sports performance analysis. In *Computer Graphics Forum*, volume 31, pages 1255–1264. Wiley Online Library, 2012.
 45. Scott W Lester, Bruce M Meglino, and M Audrey Korsgaard. The antecedents and consequences of group potency: A longitudinal investigation of newly formed work groups. *Academy of Management Journal*, 45(2):352–368, 2002.
 46. Thomas W Malone and Kevin Crowston. The interdisciplinary study of coordination. *ACM Computing Surveys (CSUR)*, 26(1):87–119, 1994.
 47. Nicolai Marquardt, Till Ballendat, Sebastian Boring, Saul Greenberg, and Ken Hinckley. Gradual engagement: facilitating information exchange between digital devices as a function of proximity. In *Proceedings of the 2012 ACM international conference on Interactive tabletops and surfaces*, pages 31–40. ACM, 2012.
 48. Nicolai Marquardt, Ken Hinckley, and Saul Greenberg. Cross-device interaction via micro-mobility and f-formations. In *Proceedings of the 25th annual ACM symposium on User interface software and technology*, pages 13–22. ACM, 2012.

49. Philipp Mayring. Qualitative sozialforschung. *Eine Anleitung zu qualitativen Denken*, 5, 2002.
50. N Menck, X Yang, C Weidig, P Winkes, C Lauer, H Hagen, B Hamann, and JC Aurich. Collaborative factory planning in virtual reality. *Procedia CIRP*, 3:317–322, 2012.
51. Daniel Mendes, M Sousa, B Araujo, A Ferreira, H Noronha, P Campos, L Soares, A Raposo, and J Jorge. Collaborative 3d visualization on large screen displays. In *Powerwall-International Workshop on Interactive, Ultra-High-Resolution Displays-ACM CHI*, volume 2013, 2013.
52. Erwin Moerland, Sebastian Deinert, Fernaß Daoud, Jochen Dornwald, and Björn Nagel. Collaborative aircraft design using an integrated and distributed multidisciplinary product development process. In *30th Congress of the International Council for Aeronautical Sciences*, 2016.
53. Brad A Myers, Choon Hong Peck, Jeffrey Nichols, Dave Kong, and Robert Miller. Interacting at a distance using semantic snarfing. In *International Conference on Ubiquitous Computing*, pages 305–314. Springer, 2001.
54. Simon Nestler, Eva Artinger, Tayfur Coskun, Yeliz Yildirim-Krannig, Sandy Schumann, Mareike Maehler, Fabian Wucholt, Stefan Strohschneider, and Gudrun Klinker. Assessing qualitative usability in life-threatening, timecritical and unstable situations, 10. In *Workshop Mobile Informationstechnologien in der Medizin (MoCoMed 2010)*, 2010.
55. Oracle-Corporation. *mysql :: mysql 5.6 reference manual :: 14.2 innodb and the acid model*, 2016.
56. Harshada Patel, Michael Pettitt, and John R Wilson. Factors of collaborative working: A framework for a collaboration model. *Applied ergonomics*, 43(1):1–26, 2012.
57. Randy Pausch, Dennis Proffitt, and George Williams. Quantifying immersion in virtual reality. In *Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, pages 13–18. ACM Press/Addison-Wesley Publishing Co., 1997.
58. Sebastian Pelurson and Laurence Nigay. Multimodal interaction with a bifocal view on mobile devices. In *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction*, pages 191–198. ACM, 2015.
59. Robert Dan Reid and Nada R Sanders. *Operations management: an integrated approach, Fourth Edition*. Wiley Hoboken, NJ, 2010.
60. Virpi Roto, Andrei Popescu, Antti Koivisto, and Elina Vartiainen. Minimap: a web page visualization method for mobile phones. In *Proceedings of the SIGCHI conference on Human Factors in computing systems*, pages 35–44. ACM, 2006.
61. Franca A Rupprecht, Bernd Hamann, Christian Weidig, Jan Aurich, and Achim Ebert. In2co-a visualization framework for intuitive collaboration. *Eurographics Conference on Visualization (EuroVis)*, 2016.
62. Franca Alexandra Rupprecht, Achim Ebert, Andreas Schneider, and Bernd Hamann. Virtual reality meets smartwatch: Intuitive, natural, and multi-modal interaction. In *Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems*, pages 2884–2890. ACM, 2017.
63. Günther Schuh, Tobias Brosze, Ulrich Brandenburg, Stefan Cuber, Michael Schenk, Jerome Quick, Carsten Schmidt, Jan Helmig, Maik Schürmeyer, and Niklas Hering. Grundlagen der produktionsplanung und-steuerung. In *Produktionsplanung und-steuerung 1*, pages 11–293. Springer, 2012.
64. Stacey D Scott, Karen D Grant, and Regan L Mandryk. System guidelines for co-located, collaborative work on a tabletop display. In *ECSCW*, volume 3, pages 159–178, 2003.
65. Julian Seifert, Adalberto Simeone, Dominik Schmidt, Paul Holleis, Christian Reinartz, Matthias Wagner, Hans Gellersen, and Enrico Rukzio. Mobisurf: improving co-located collaboration through integrating mobile devices and interactive surfaces. In *Proceedings of the 2012 ACM international conference on Interactive tabletops and surfaces*, pages 51–60. ACM, 2012.
66. Navid Shariatzadeh, Gunilla Sivard, and Danfang Chen. Software evaluation criteria for rapid factory layout planning, design and simulation. *Procedia CIRP*, 3:299–304, 2012.
67. Thomas B Sheridan. Teleoperation, telerobotics and telepresence: A progress report. *Control Engineering Practice*, 3(2):205–214, 1995.
68. Ben Shneiderman. The eyes have it: A task by data type taxonomy for information visualizations. In *Visual Languages, 1996. Proceedings., IEEE Symposium on*, pages 336–343. IEEE, 1996.
69. Martin Steiger, Jürgen Bernard, Sebastian Mittelstädt, Hendrik Lücke-Tieke, Daniel Keim, Thorsten May, and Jörn Kohlhammer. Visual analysis of time-series similarities for anomaly detection in sensor networks. In *Computer Graphics Forum*, volume 33, pages 401–410. Wiley Online Library, 2014.
70. Eric Sundstrom, Kenneth P De Meuse, and David Futrell. Work teams: Applications and effectiveness. *American psychologist*, 45(2):120, 1990.
71. Matthew Tobiasz, Petra Isenberg, and Sheelagh Cpendale. Lark: Coordinating co-located collaboration with information visualization. *IEEE transactions on visualization and computer graphics*, 15(6):1065–1072, 2009.
72. T Tolio, Darek Ceglarek, HA ElMaraghy, A Fischer, SJ Hu, L Laperrière, Stephen T Newman, and József Váncza. Species-co-evolution of products, processes and production systems. *CIRP Annals-Manufacturing Technology*, 59(2):672–693, 2010.
73. Gerrit van der Veer and Martijn van Welie. Task based groupware design: putting theory into practice. In *Proceedings of the 3rd conference on Designing interactive systems: processes, practices, methods, and techniques*, pages 326–337. ACM, 2000.
74. Martijn Van Welie, Gerrit C Van der Veer, and Anton Eliëns. An ontology for task world models. In *Design, Specification and Verification of Interactive Systems 98*, pages 57–70. Springer, 1998.
75. Mark Vollrath. *Ingenieurpsychologie: Psychologische Grundlagen und Anwendungsgebiete*. Kohlhammer Verlag, 2015.
76. Matthew O Ward. A taxonomy of glyph placement strategies for multidimensional data visualization. *Information Visualization*, 1(3-4):194–210, 2002.
77. Christian Weidig, Péter Galambos, Ádám Csapó, Péter Zentay, Péter Baranyi, Jan C Aurich, Bernd Hamann, and Oliver Kreylos. Future internet-based collaboration in factory planning. *Acta Polytechnica Hungarica*, 11(7), 2014.