A design view of capability

Wenjuan Wang, Alex Duffy, Ian Whitfield, Shaofeng Liu and Iain Boyle
Department of Design, Manufacture and Engineering Management,
University of Strathclyde
wenjuan.wang@strath.ac.uk; alex.duffy@strath.ac.uk; ian.whitfield@strath.ac.uk; shaofeng.liu@strath.ac.uk; iain.m.boyle@strath.ac.uk

Abstract

In order to optimise resource deployment in a rapid changing operational environment, capability has received increasing concerns in terms of maximising the utilisation of resources. As a result of such extant research, different domains were seen to endow different meanings to capability, indicating a lack of common understanding of the true nature of capability. This paper presents a design view of capability from design artefact knowledge perspective. Capability is defined as an intrinsic quality of an entity closely related to artefact behavioural and structural knowledge. Design artefact knowledge was categorised across expected, instantiated, and interpreted artefact knowledge spaces (ES, IsS, and ItS). Accordingly, it suggests that three types of capability exist in the three spaces, which can be used in employing resources. Moreover, Network Enabled Capability (NEC), the capability of a set of linked resources within a specific environment is discussed, with an example of how network resources are deployed in a Virtual Integration Platform (VIP).

1 Introduction

As the noun format of capable, capability originated from Latin "Capax", which means "able to hold much" [1]. Generally, capable means having attributes required for performance or accomplishment, and capability of something or someone means the quality of being capable [2].

With different criteria of performances and accomplishments, capability has been endowed with different meanings in different domains, such as military, industry, design, business management, etc. To respond to the rapidly changing environment within which the forces operate, the UK MoD proposed Network Enabled Capability (NEC) [3], with capability being its fundamental element. New approaches to the design, acquisition, and management of systems that support capability are therefore required so as to maximise utilisation of limited resources within such a dynamic environment. A common view of capability from a knowledge perspective could facilitate such systems engineering support of NEC.

The objective of this paper is to explore the nature of capability from a human being's point of view, and present a model of capability from a design artefact knowledge perspective. From a post-positivism view [4], fundamental design artefact knowledge includes functional, behavioural, and structural knowledge distributed among three spaces, i.e., the expected, instantiated and interpreted design artefact knowledge spaces (ES, IsS, and ItS). ES composes of designers' expectations towards a designed artefact, such as what components it will contain, how it will function and behave. IsS contains the design artefact knowledge that has been specified by designers and could be realised in a future implementation. Lastly, ItS exists in designers' minds which is built up from their interpretation of the artefact being designed. These three design spaces contain design artefact knowledge in different states [4]. Capability is an intrinsic quality of an entity able to deliver a desired effect. It is closely related to the entity/artefact's structure and behaviour. Hence it is hypothesised that capability of an artefact also exists within the three spaces.

Within a network environment, a resource's networked capability (i.e. a resource's capability in a network environment) and networked resources' capability (i.e. the capability(ies) resulting from resources that are networked) require further consideration in order to maximise resource utilisation. A Virtual Integration Platform (VIP) provides resource management through two approaches incorporating such concepts.

The remainder of the paper is organised as follows. Different definitions and models of capability are presented in section 2 followed by a comparison among them. Section 3 presents a post-positivism view of function behaviour structure, which builds the basis for the study of capability. Section 4 provides a design view of capability from artefact knowledge perspective, and presents two resource deployment approaches in network environment. At last, some concluding remarks are given in section 5.

2 Current capability models

With different criteria of performances and accomplishments, capability has been endowed with different meanings in different domains, such as military, industry, design, business management, etc.

In the UK **military** domain, capability is defined as the ability to execute a specified course of action within the MOD Architecture Framework [5]. Specifically, it is the enduring ability to generate a desired operational outcome or effect, and is related to threat, physical environment, and contributions of coalition partners [6]. The Development, Concepts and Doctrine Centre (DCDC) offers a high level of Defence Capability Framework with seven “capabilities” as the primary elements of the military domain: command, inform, prepare, project, protect, sustain, and operate [7] (see Figure 1). It could be observed that these “capabilities” reflect the desired goals of various activities carried out in the military domain.

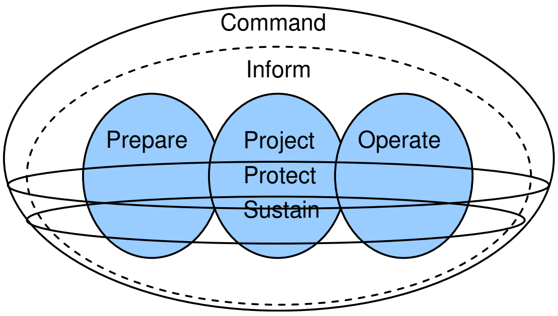

Figure 1. Military capability [7]

The **industrial** perspective of capability is generally defined around the five elements of resources: people, process, products, technology, and facilities (P3TF) (see Figure 2). The products are the focus of the model because each of the other elements contributes to the development and sale of the products [7]. It could be perceived that these five elements are the resources utilised within organisation, which possess certain capabilities and contribute to the business success of the industry.

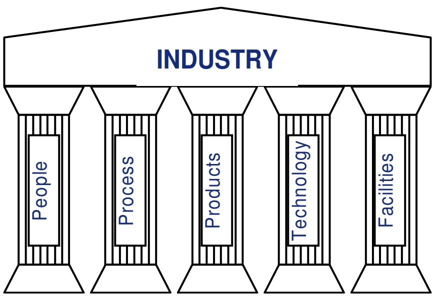

Figure 2. Industry capability [7]

In engineering **design**, based on the design activity model developed by O’Donnell [8], Haffey defined the capabilities possessed by a resource, or collection of resources as the tasks and activities that a resource(s) is perceived as being capable of undertaking within a given context [9]. According to this definition, the capability of a resource could be regarded as including the activity it could be used for undertaking along with the activity’s outputs resulted from execution of the activity (Figure 3).

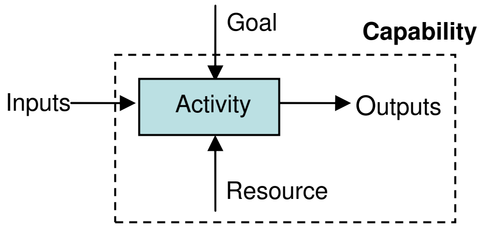

Figure 3. Capability in design (Activity model from [8])

It can be seen that the above three definitions model capability from different aspects of a design activity. Table 1 summarises the comparison among them, and Figure 4 illustrates such comparison based on O’Donnell’s activity model. The comparison shows that the capability discussed in different domains reflects different scope of capability based on the domains’ interests.

Table 1. A comparison of different definitions of capability

Research domain	Capability definition comparison
Military	Goals of various **activities** are defined as **seven types of capability**.
Industry	Different **resources** that possess capability contributing to business success are defined as **five types of capability**.
Design	The **activity** that a resource could undertake and the **related outputs** are defined as **capability**.

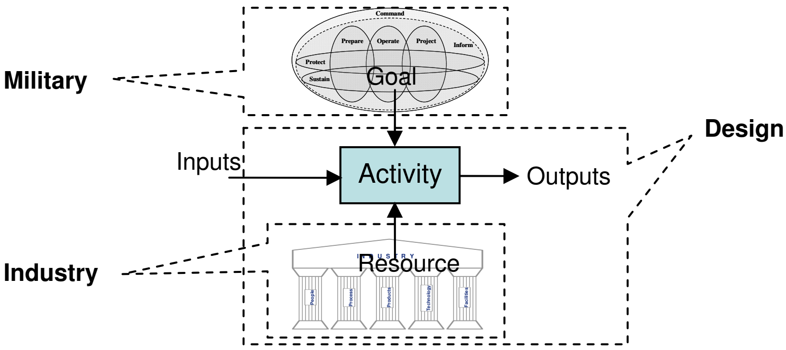

Figure 4. Different scope of capability discussed in different domains

In addition to the aforementioned three definitions, Daw offered a system view of capability by reconciliation of different definitions in one structure architecture [7]. However, a unified definition of capability was not given. To support a common understanding of capability, the following sections discuss capability from artefact knowledge perspective.

3 A post-positivism view of function behaviour structure (P-FBS)

Since its recognition in the 1950s [10], post-positivism has provided an alternative to the traditional positivism approach for conducting disciplined inquiry. Positivism is a philosophy that regards reality as existing while being independent of human being's thought and behaviour, which can be studied as natural objects [11]. However, one major criticism of this philosophy is that "it does not provide the means to examine human beings and their behaviours in an in-depth way" [11]. In contrast, post-positivist researchers believe that there exists a real world independent of human mind. However, reality exists in the mind of human beings [11]. Much of the nature of design research is similar to cognitive psychology or sociology due to the involvement of people, society, and organisations. Hence, this research was conducted considering the human perspective.

As mentioned earlier, fundamental design artefact knowledge can be represented as functional, behavioural, and structural knowledge elements distributed among ES, IsS, and ItS. However, from a post-positivism view, a function only exists in the ES and ItS, and structure only exists in the ES and IsS. Consequently, causal relationships among function, behaviour, and structural are limited to where these fundamental artefact knowledge elements exist [4].

3.1 Design artefact knowledge

Strictly speaking, design **requirements** (R) don't belong to artefact knowledge. However, they are the origins of the artefact knowledge, and are descriptions of constraints or specifications. Generally, requirements can be derived from some **motivating needs or desires** (M) of the customers/designers.

The **function** (F) of an artefact is its intention, purpose [12, 13] or duty [14]. From a post-positivism viewpoint, artefact function is a subjective and situated concept. Depending on whether it is derived from designers' intentional expectation, or their interpretation of the artefact being designed, artefact function can be categorised into expected function (F_e) in the ES and interpreted function (F_{it}) in the ItS. The former stems from R, the latter is derived from the artefact instantiated structural and behavioural knowledge (see structure and behaviour part). The F_e and F_{it} can be used to evaluate the designed artefact by judging whether the designed artefact can provide the F_e.

Simulating how an artefact works, **behaviour** (B) describes what the artefact does, and how it achieves its functions [15]. An artefact functions in specific environments and therefore its behaviour is the effect of an artefact's interaction with its environment [12]. In comparison with function, artefact behaviour could be either an objective or subjective concept. On the one hand, it can be derived by objective qualitative physics [16]. On the other, it can also be derived by subjective observation. Viewed in this regard, three types of behavioural knowledge can be employed in defining an artefact. The first is called expected behaviour (B_e) in the ES, which is the attributes expected from the artefact's structure and can be derived from its F_e. The second is instantiated behaviour (B_{is}) in the IsS, which is also called behaviour of structure [17]. This type of behaviour is derived directly from the artefact structure that the designers are currently working on, representing all the possible behaviour an artefact can exhibit in a specific environment. The last one, interpreted behaviour (B_{it}) in the ItS, refers to the behaviour exhibited by an artefact observed and interpreted by designers within a specific working environment, which is an explanation or analysis of an artefact according to the designers' expectation. Accordingly, B_{it} can then be used to evaluate the design by comparing with B_e.

Generally defined as the artefact's components, form and their physical relationships, **structure** (S) describes distinctive attributes that identify the artefact, and their interactions [18]. With a post-positivism viewpoint, structural knowledge exists either in relation to the designers' expectation towards what the artefact structure will or should be, or in relation to the state that has been instantiated by designers for the current artefact. Consequently, an artefact's structure can be classified into expected structure (S_e) in the ES and instantiated structure (S_{is}) in the IsS.

Designing is a constrained activity [17] which is restricted by various **constraints** (Ct). Design constraints include various design specifications, needs, performance criteria, objectives, etc. [19], which can represent conditions that are set restrictions in relation to the F, B and S.

The cause-effect links among the aforementioned R, F, B, and S form the **causal relationships** (CR) that can reflect the evolution of design artefact knowledge.

3.2 A post-positivism view of FBS

The above discussion reveals that function of an artefact exists in the ES and ItS; behaviour exists in all three spaces; and structure only exists in the ES and IsS. Taken together, there are seven fundamental artefact knowledge elements: F_e, B_e, S_e, B_{is}, S_{is}, B_{it}, and F_{it}. In consequence, the CR among F, B, and S are limited to where these elements exist. The existence of the artefact knowledge elements and their causal relationships in the three spaces thus form a post-positivism view of FBS (P-FBS) [4]. As shown in Figure 5, R can be derived from motivations (M) in the ES. F_e then could be deduced from R, and B_e from F_e. S_e can be derived from B_e by synthesis. Then S_e can be embodied to S_{is} in the IsS, and B_{is} could be derived from S_{is} in this space. Based on the B_{is}, designers could interpret B_{it} from it, and this could then be interpreted to F_{it}. Once B_{it} and F_{it} are derived, a comparison between B_e and B_{it}, and F_e and F_{it} can reveal whether the design satisfies R. If the design is plausible, design description (D) can be documented in the IsS. Detailed explanations of the model can be found in reference [4].

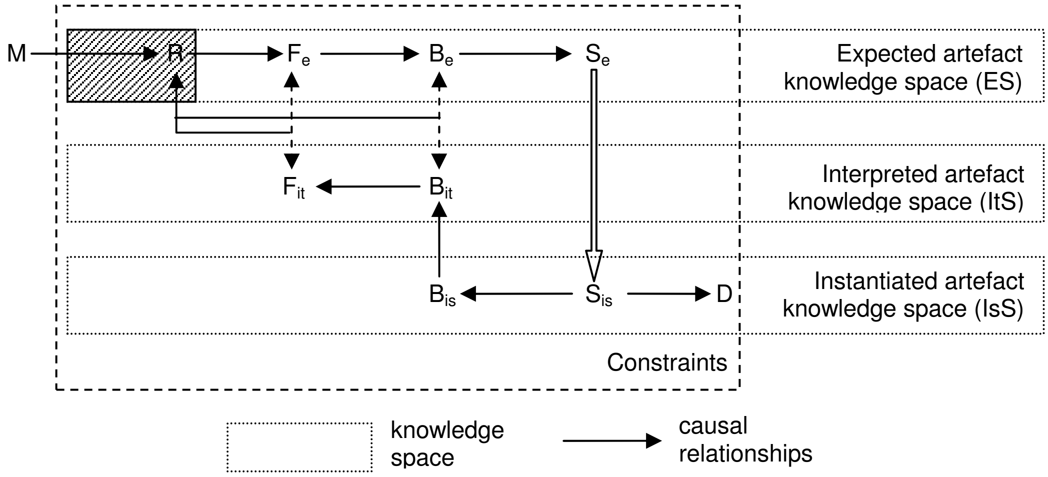

Figure 5. Post-positivism view of FBS (Adapted from [4])

4 Design view of capability

As a follow-up to the above discussion, this section discusses capability from a design viewpoint taking an artefact knowledge perspective.

4.1 Capability definition and modelling

As mentioned in section 3.2, F, B, and S are fundamental artefact knowledge elements. The capability of an object within a specific environment is its all possible ability to achieve certain performance or accomplishments, which is fulfilled or revealed through certain behaviours that the object can provide in the environment. Capability does not exist independent of the existence of a structure. Hence it is an intrinsic quality of the structure. To evaluate whether the object possesses specific capability, or has been brought to all of its possible capabilities in a specific environment, performance of the behaviour it carries out in such an environment can be used as a criterion.

Specifically, capability of an object is closely related to its structure and behaviour, and it can be defined from three aspects: First, it is an intrinsic quality or property of an object which shows its possible ability to perform or to accomplish something with respect to a set of criteria; Second, it can be revealed or exhibited through the object's behaviour within a particular environment; And last, it can be measured by the behaviour's performance. Figure 6 illustrates the relationships between an object's capability and its structure, behaviour, and performance.

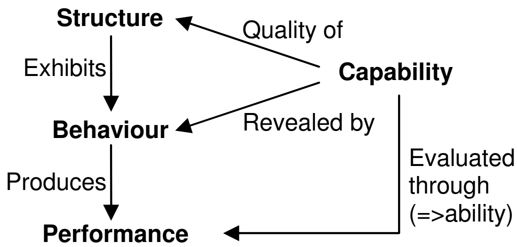

Figure 6. Capability definition

Based on the above discussion, capability is thus defined in this paper as:

an intrinsic quality of an entity able to deliver a desired effect.

From designers' point of view, behavioural knowledge distributes among the ES, IsS, and ItS. Since capability of an entity is exhibited through its behaviour, it can also be categorised into three types: expected capability (C_e), interpreted capability (C_{it}) and potential capability (C_p) (See Figure 7). C_e is revealed by B_e, which shows the artefact's capabilities expected by the designers within a specific environment. C_{it} is exhibited by B_{it} that is potential behaviour perceived by the designers. C_{it} thus shows the designers' interpretation of a designed artefact's capability within a specific environment. C_p refers to the capabilities revealed by a designed artefact's B_{is} that includes all the potential capabilities the artefact possesses within all possible environments.

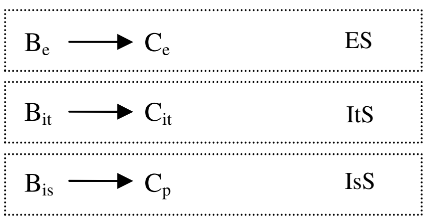

Figure 7. Design view of capability

To illustrate the nature of capability, the protocols of a "Roadside furniture" design project was analysed in order to observe any existence of these three types of capability. The project was carried out by a 4th year Produce Design student in the Design, Manufacture and Engineering Management department, University of Strathclyde, from Septemper 2005 to April 2006. As part of the design, "Post Installation" has been applied for a British patent, with the application filing number 0613906.7. An analysis of the protocols verifies the existence of these three types of capability in design. In this regard, Table 2 lists three examples, which are extracted from the protocols.

Table 2. Examples of capability

Expected capability (C_e)	"Because you are actually going to design something, and one of the benefits would be the **ability to be replaced**, **be recycled**, and **positioned really easily**,…"
Interpreted capability (C_{it})	"…The actual barrier itself could **encourage better green cross codes**, crossing road in a safer manner. And the installation mechanism ..."
Potential capability (C_p)	"They protect you from something, **barriers are protective**."

Having defined capability from the artefact knowledge perspective, the next section discusses how this definition can be used for deploying resources in an operational environment.

4.2 Using capability for deploying resources

In order to derive the right effect in the military operational domain, it is imperative to allocate the right resource to an activity based on the right reason, at the right place, and at the right time [20]. Coates et al. [21] have proposed an approach to resource management during a dynamic process in their Design Co-ordination System (DCS). The approach optimises the utilisation of resources (which are workstations executing tasks in a network environment), through monitoring resources and assuring that they can be continuously utilised in such a process. To conduct an activity effectively, this paper proposes that if the deployment of an object/resource is regarded as a design process, the above presented C_e, C_{it}, and C_p could be applied in designing the deployment of appropriate resources with the consequence of achieving certain goal(s) from the deployers' viewpoint. A resource could be any object deployed in the operational environment, such as people, hardware equipment, software, information, procedure, etc. To conduct the activity, resources possessing certain capabilities are required. Thus, resources can be selected by comparing the required capabilities for conducting the activity and the C_e of the deployable resources. During this process, the range and extent of capability should be considered [9. p310]. Once the resources have been deployed, their C_p will be explored during the activity execution. As a result, C_{it} could be observed from the output of the activity (see Figure 8).

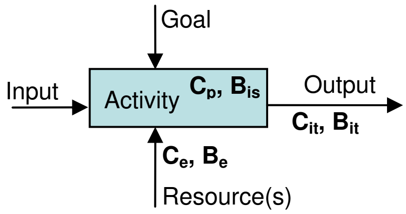

Figure 8. Resources deployment

It should be mentioned that sometimes, if there are no suitable resources available for an activity during its deployment, substitute resource(s) that might possess expected capability can also be chosen for conducting the activity. However, whether the deployment is at the risk of failing the activity needs to be carefully considered in such a situation.

4.3 Measuring capability

Once appropriate resource(s) have been deployed for a design activity, its performance can be measured following the activity's execution. In this regard, O'Donnell [8] formulated an E^2 design performance model which evaluates the activity by its effectiveness and efficiency. Effectiveness is the degree to which the output of the design activity meets its goal. Efficiency is the relationship between what has been gained (from input to output) and the level of resource(s) used. If the activity is regarded as any activity carried out in the operational environment, the capability of a resource could be measured by using the performance of the activity (Figure 9). Depending on whether the effectiveness and efficiency of the activity match the deployers' expectation, resource(s) could be reallocated for better performance if possible.

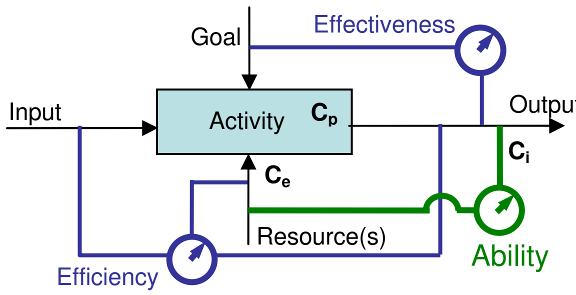

Figure 9. Measuring capability

Definitions of capability [5] are often mixed with ability. Though ability has a close affinity with capability, it is proposed here that it is the result of an assessment of the effects that a resource can produce. Thus, it provides an indication of an assessed capability. For example, a warship may be designed to have the capability of landing two helicopters at the same time. However, we determine its ability to do so after it has done it, through, for example, sea trials. As shown in

Figure 9, ability can be presented as an assessment of a resource to produce an output.

4.4 Network enabled capability (NEC)

In an operational environment, multiple resources often work together in a dynamic co-operative manner. This implies that the capabilities of resources will be different in such an environment, compared to when they work separately. NEC is the capability derived from a set of resources/objects (people, equipment, process, software, information, etc.), which are linked via various types of relationships. It has received increasing attention from the UK MoD in order to maximise the performance of limited resources in such a dynamic operational environment [20].

As shown in Figure 10, in a network environment (Context'), resources (R1, R2, R3) could be connected via different types of links in the form of information (I), energy (E), material (M), [22] or psychology (P). These links affect each resource's capability in the network environment. Consequently, each resource possesses networked capability (C1', C2', C3') which is different from that (C1, C2, C3) in an independent (Context) one.

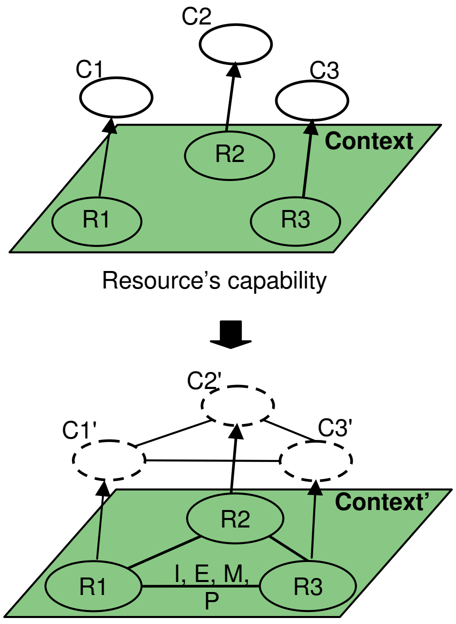

Figure 10. Resrouce's networked capability

The combination of these networked capabilities of individual resources can then be termed networked resources' capability, which is the holistic capability possessed by these resources in such an environment. Hence two types of NEC could be observed from a dynamic network environment. The first is a resource's networked capability which is the capability possessed by individual resource in the network environment (C1', C2' and C3' in Figure 11). The second is networked resources' capability (the holistic capability of all the linked resources), which emerges in a network environment ($C^{\#}$ in Figure 11). Therefore, in such environment, one main consideration of deploying resources is how to enhance the resource's networked capability and networked resources' capability.

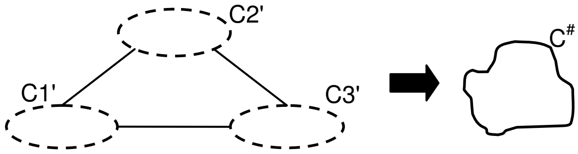

Figure 11. Resource's networked capability and Networked resources' capability

4.5 Virtual deployment of resources

To illustrate how to deploy resources in a network environment, based on a resource's networked capability and networked resources' capability, this section introduces a Virtual Integration Platform (VIP) that is capable of scheduling and allocating design activities to organisationally and geographically distributed resources (e.g. designers) [23]. To achieve this, the platform consists of a number of components that contribute to the engineering management and co-ordination of data, resources, activities, requirements, and processes. The information required to schedule and allocate activities to designers is defined in terms of: the designers' capability to perform particular design activities; commitment in terms of the design activities that it is currently performing, and capacity to perform more than one design activity at the same time as well as the effect of increased capacity on capability.

Previous approaches have been developed to automatically deploy network distributed resources to activities [21], however these approaches have generally been applied either within the context of real-time allocation of computational resources using automated design tools, or in the planning of human resources within future design projects and not for the real-time deploying a combination of human and computational resources. The resource management procedure presented here is based upon this previous research and involves: the determination of the design activities that need to be undertaken on the basis of the goals that need to be achieved; identification of the resources with respective capabilities that can undertake these design activities; and, the use of a genetic algorithm to optimally allocate the activities to the resources based on the networked resources' capability. Since the focus of the procedure is toward the real-time allocation of design activities to designers, additional human issues with respect to scheduling are considered. These issues aspects include: consideration of the improvement in performance as a result of the experience gained from undertaking the activity; provision of a training period to allow inexperienced designers the opportunity to improve their performance without their performance being assessed; and the course of action to take when a designer is either unwilling or unable to perform an activity.

A Process Control Tool (PCT) has been developed within the VIP to manage the real-time enactment of processes through interaction with the VIP to allocate activities to designers. The PCT is capable of managing multiple processes each containing thousands of activities, as well as multiple designers logged onto the VIP each capable of performing various different subsets of the complete range of activities to various degrees of efficiency. The relationship between the PCT and other components within the VIP can be seen within Figure 12.

In order to provide efficient resource allocation, the PCT is required to ensure that the most appropriate designers are scheduled for the activities. A resource management model was created within the PCT providing the following functionality:

- Manage the login administration processes onto the VIP.
- Configure activities and tools to be integrated within the VIP providing a mapping within the PCT of a designer’s capability.
- Allow activities to be allocated to users of the VIP.
- Communication with other designers.

The resource model contains a list of all of the designers that are registered to use the VIP. Designer details are provided relating to their name, login, password, and email address. Additional contact information may be provided for the designer’s department, company, address, and telephone number. Designers are defined within the PCT as having capability, commitments (information related to which activities they are currently undertaking, and have undertaken in the past), and as being project managers. When a new project is created within the PCT, a project manager is associated with it. The designer’s IP address is also obtained when the designer logs onto the VIP, and is used for all communication.

Through the VIP user interface, a designer can configure and integrate a design or simulation tool into the VIP, and in doing so, map the use of the tool to an activity within a process contained within the PCT – updating the designer’s capability automatically. The capability defined for each designer allows the VIP to keep track of the number of times that the designer has performed the activity in the past, as well as the associated durations. This information is used within both the local and global scheduling processes introduced below. When a designer configures a tool and registers a new capability, the PCT has a provision to allow the activity to be undertaken a number of times during a training period without the activity durations being considered in any subsequent planning. This training period is used within process scheduling to ensure that designers that have configured new capability (which could potentially take a number of undertakings before the designer becomes competent in performing the activity), are not always overlooked in preference for a designer that is more experienced and capable of performing the activity in less time.

Two separate approaches have been implemented within the PCT to determine the most appropriate resource for any particular activity based on a resource’s networked capabilities, and the networked resources’ capability: local and global scheduling. The PCT can be easily configured to use either of the two approaches with each approach having the benefits and shortcomings as described within the following sections.

The local scheduling approach determines the most appropriate designer to allocate an activity to according to the sequence of enactment and the resource’s networked capability, which is the designer’s capability in the network environment. When an activity requires scheduling, the PCT queries the resource model and generates two lists of designers that are capable of

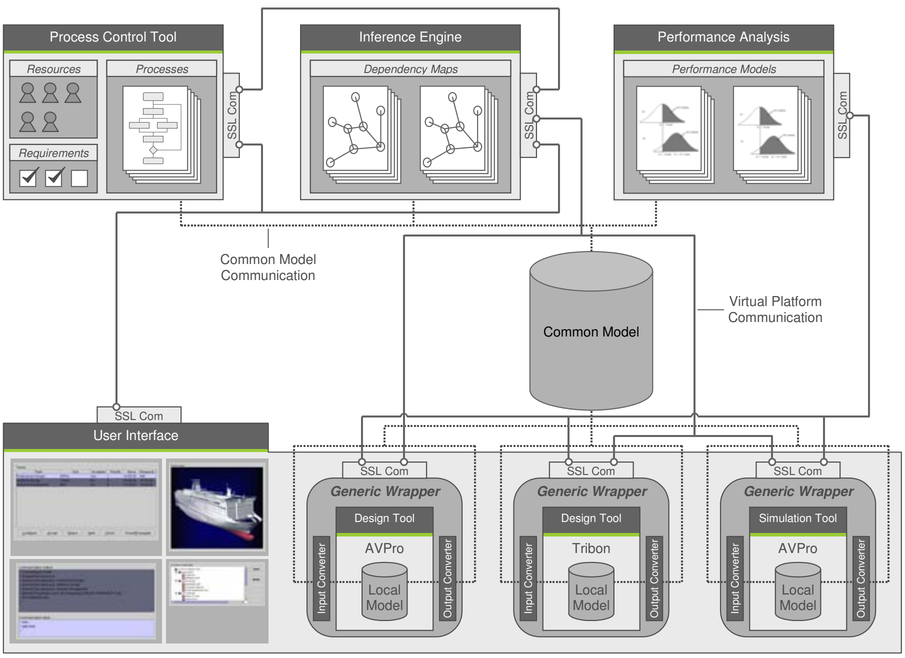

Figure 12. VIP components

performing the activity. The first list represents designers that have completed the training period for the activity (and are therefore assumed to have the ability and be competent), and the second list represents the designers that are within their training period for that activity (and are therefore assumed to have the expected capability). The lists are ranked using information relating to how much time each designer expects to undertake the activity. Bias is always given towards designers that are within the training or "probationary" period for the activity for two reasons: it gives all designers the possibility of becoming more competent; and it ensures that the scheduling algorithm does not always allocate the most efficient designer to an activity, without at least giving newly capable designers the opportunity to improve their efficiency. If more than one designer is within the "in training" list, the designer is chosen at random.

The PCT may however fail to locate an appropriate designer for the following reasons: there are no designers currently online that can perform the activity; or there are no designers that can perform the activity. The control of the process is managed in these circumstances by the "Email Offline Resources", "Allocate to Managers", and "Bypass" control options – Figure 13. If an appropriate online designer could not be identified, and the "Email Offline Resources" option was checked, the PCT would generate a designer list for offline designers. The most appropriate offline designer would be selected, and the PCT would automatically send an email to the designer to inform that they have been scheduled for an activity, and request that the designer logs onto the VIP. The PCT would pause the activity and change its state to "pending". When the scheduled designer, or any other designer that was capable of undertaking the activity, next logged onto the platform, the PCT would automatically re-start and allocate the activity.

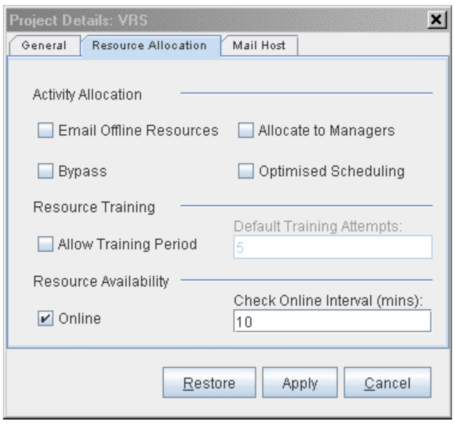

Figure 13. Resource allocation configuration

When an appropriate designer is scheduled, the PCT communicates with the designer to allocate the activity. The PCT determines the time-spent by the designer undertaking the activity taking into account the other commitments. There can be no guarantee however that the most appropriate designer for each individual activity, would also be the most appropriate designer when considered from a global perspective. This approach, cannot guarantee an optimum process lead-time, however it does provide additional functionality for managing activities that cannot be allocated, and for improving the designers' efficiency through a training period. The approach may however be improved by determining the most appropriate designers for the future activities which would produce an optimum lead-time with respect to all of the activities that are currently being enacted (ignoring future activities) at any point in time.

For global scheduling, the PCT considers the requirements of all of the activities within all of the active processes simultaneously with the aim of minimising the lead-time. The networked resources' capability is considered in this approach while deploying resources. When scheduling multiple processes, the PCT automatically generates a new schedule (for all active processes) whenever it attempts to start an activity that is not currently within its schedule. The schedule represents a mapping between each of the activities that require performing, the designers that will perform them, and the time period in which the activity would be performed.

A Genetic Algorithm (GA) is used to enable the optimisation of the schedule. The GA initially creates a population representing a number of plans, which consist of a randomly generated sequence of activities for each process. A schedule model is then used to select the most appropriate designer for each activity within this random sequence. The global scheduling approach does not consider whether the designer is currently online, since it may be scheduling a designer for an activity many days into the future. It also makes no consideration for bypassing activities – any activity that cannot be allocated to a capable designer, would be allocated to a project manager. The designers are selected using a similar basis as the approach used for local scheduling, with the exception that consideration needs to be given to the fact that the activities are not to be enacted immediately (as is assumed within local scheduling). Consideration therefore needs to be given for a designer's future commitments.

The schedule provides an evaluation of the plans for each process and produces an estimate for the combined lead-time of all of the processes. The GA uses conventional techniques such as selection, crossover and mutation, over a number of generations in order to refine the plans and generate a schedule that has a near-optimum allocation of activities to designers with respect to process lead-time. The networked resources' capability thus can be considered in such a situation.

Global scheduling is a dynamic approach, reacting to the changing process demands, as well as considering the most appropriate designers in order to minimise the lead-times of all of the active processes. Whenever an activity is completed, the scheduled designer is removed from the schedule, and a new optimal plan is produced in order to maximise resources' utilisation at any time. A shortcoming of this approach is that the scheduling algorithm does not consider the availability of designers during working hours, which is compounded by the fact that the designers may be distributed across various time-zones, as well as the possible variation in the schedule and potential un-availability of a scheduled designer some time the future. These issues could however be addressed by continually assessing the deviation from the schedule and re-scheduling when the deviation exceeds pre-defined limits [24].

5 Conclusion

Capability has been endowed with different definitions within different domains. As the fundamental element of Network Enabled Capability (NEC), capability needs a common understanding as a basis for further systems engineering support of NEC in order to maximise resources' utilisation in a dynamic environment. Viewing resource deployment as a design process, this paper defines capability from a design artefact knowledge perspective as an intrinsic quality of an entity able to deliver a desired effect. Three types of capability were defined from a human being's view point, i.e., expected, potential, and interpreted. In turn, they can be used for resource deployment and evaluation. In addition, NEC was discussed in the paper, which includes a resource's networked capability and networked resources' capability. A virtual deployment of resources was presented with two resource deployment approaches addressing these two concepts.

Acknowledgement

This work has been supported by Universities UK (ORS), University of Strathclyde (IRS), and British Federation of Women Graduates (BFWG). We would like to express sincere appreciation to Miss Laura Crawford and Prof. Norman McNally for their help and support in the design protocol analysis. The VIP mentioned within this paper was the research conducted in VRShips, a project funded by the European Commission (grant No. G3RD-CT-2001-0056), which is part of the Fifth Framework Programme for Research, Technological Development and Demonstration. The opinions expressed are those of authors and should not be construed to represent the views of VRShips partners.

6 References

[1] Harper, D. *Online etymology dictionary*. 2001 [cited 2007 12 September]; Available from: http://www.etymonline.com

[2] Merriam-Webster, *Collegiate Dictionary*. 2005.

[3] MOD, *Joint Services Publication 777 - Network Enabled Capability*. 2004.

[4] Wang, W., Duffy, A. and Haffey, M., *A post-positivism view of function behaviour structure*, in *International Conference of Engineering Design '07*. 2007: Paris.

[5] MOD, *MOD Architecture Framework* http://www.modaf.org.uk/wgloss/70/glossary, Last accessed April 2008

[6] MOD, *What is capability management.* http://www.ams.mod.uk/aofcontent/operational/business/capabilitymanagement/capabilitymanagement_whatis.htm, Last accessed April 2008

[7] Daw, A.J., *New process and structure thinking for capability development*, in *9th International Command and Control Research and Technology Symposium*. 2004: Copenhagen, Denmark.

[8] O'Donnell, F.J. and Duffy, A.H.B., *Design performance*. 2005, London: Springer.

[9] Haffey, M.K.D., *An approach, insights and methodology for performance improvement through process activity management*, in *Design Manufacture and Engineering Management*. 2007, University of Strathclyde: Glasgow.

[10] Popper, K.R., *The logic of Scientific Discovery*. 1959, London: Hutchinson.

[11] Crossan, F., *Research philosophy: towards an understanding.* Nurse Researcher, 2003. 11(1): p. 46-55.

[12] Hybs, I. and Gero, J.S., *An evolutionary process model of design.* Design Studies, 1992. 13(3): p. 273-290.

[13] Qian, L. and Gero, J.S., *Function-behaviour-structure paths and their role in analogy-based design.* Artificial Intelligence for Engineering Design, Analysis and Manufacturing, 1996. 10(4): p. 289-312.

[14] Hubka, V., *Principles of Engineering Design*. 1st ed. 1982, Zurich: Butterworth & Co Ltd. 118.

[15] Gorti, S.R., Gupta, A., Kim, G.J., Sriram, R.D. and Wong, A., *An object-oriented representation for product and design processes.* Computer-Aided Design, 1998. 30(7): p. 489-501.

[16] Bobrow, D.G., ed. *Qualitative reasoning about physical systems*. Artificial Intelligence. 1984, MIT Press: Cambridge.

[17] Gero, J.S., *Design Prototypes: A Knowledge Representation Schema for Design.* AI Magazine, 1990. 11(4): p. 26-36.

[18] Kuipers, B., *Commonsense reasoning about causality: Deriving behavior from structure.* Artificial Intelligence, 1984. 24(1-3): p. 169-203.

[19] Ullman, D.G., Dietterich, T.G. and Stauffer, L.A., *A Model of the mechanical design process based on empirical data.* Artificial Intelligence for Engineering Design, Analysis and Manufacturing, 1988. 2(1): p. 33-52.

[20] Andrews, J.D. and Henshaw, M.J., *A research proposal in systems engineering addressing the question: "Are you prepared for NEC?"* 2005.

[21] Coates, G., Duffy, A.H.B., Whitfield, R.I. and Hills, W., *Real-Time Co-ordinated Resource Management in a Computational Environment*, in *14th International Conference on Engineering Design (ICED `03)*. 2003: Stockholm, Sweden.

[22] Zhang, Y., *Computer-based modelling and management for current working knowledge evolution support*, in *CAD Centre, Design Manufacture and Engineering Management*. 1999, University of Strathclyde: Glasgow.

[23] Whitfield, R.I., Duffy, A. and Coates, G., *Real time resource scheduling within a distributed collaborative design environment*, in *16th International Conference on Engineering Design (ICED `07)*. 2007: Paris, France.

[24] Coates, G., Duffy, A.H.B., Whitfield, R.I. and Hills, W., *An integrated agent-oriented approach to real-time operational design co-ordination.* Journal of Artificial Intelligence for Engineering, Design, Analysis and Manufacturing, 2003. 17: p. 287-313.