Original citation:

Permanent WRAP url:
http://wrap.warwick.ac.uk/61707

Copyright and reuse:
The Warwick Research Archive Portal (WRAP) makes this work of researchers of the University of Warwick available open access under the following conditions. Copyright © and all moral rights to the version of the paper presented here belong to the individual author(s) and/or other copyright owners. To the extent reasonable and practicable the material made available in WRAP has been checked for eligibility before being made available.

Copies of full items can be used for personal research or study, educational, or not-for-profit purposes without prior permission or charge. Provided that the authors, title and full bibliographic details are credited, a hyperlink and/or URL is given for the original metadata page and the content is not changed in any way.

Publisher’s statement:
This is an Accepted Manuscript of an article published by Taylor & Francis Group in International Journal of Production Research on 23/05/2014 available online at http://www.tandfonline.com/10.1080/00207543.2014.918287

A note on versions:
The version presented here may differ from the published version or, version of record, if you wish to cite this item you are advised to consult the publisher’s version. Please see the ‘permanent WRAP url’ above for details on accessing the published version and note that access may require a subscription.

For more information, please contact the WRAP Team at: publications@warwick.ac.uk

http://wrap.warwick.ac.uk/
Resource selection ontologies in support of a recipe-based factory design methodology

K. Agyapong-Kodua*, Csaba Haraszkó, István Németh

(k.agyapong-kodua@warwick.ac.uk, haraszko@manuf.bme.hu, inemeth@manuf.bme.hu)

1WMG Department, University of Warwick, Coventry, CV4 7AL, United Kingdom

2Department of Manufacturing Science and Technology, Budapest University of Technology and Economics, Budapest

Egry József utca 1, H-1111, Hungary

Abstract

Factory (re) design involves the appropriate utilisation of product, process and resource knowledge in the determination of suitable configurations of physical factory facilities which have the potential to meet industrial and organisational requirements. Different independent semantic modelling standards exist for products, processes and resources but to automate and facilitate the selection of factory resources, there is the need to semantically integrate resource capabilities with product and process requirements. This paper defines a ‘recipe-based’ approach to designing factories and it is based on the assumption that capabilities and competences of reusable components (or building blocks) of factory resources can be semantically modelled and matched with product-process requirements. This approach will enable Factory Designers and Engineers to decide on relevant resource systems within finite, valuable and defined sets of requirements.

Keywords: Semantic modelling, Digital factory design, ontologies, resource selection
1.0 Introduction

Manufacturing Enterprises (MEs) are faced with intense competition arising from internal and external changes, which induce dynamics directly or causally on products, processes and their associated resources (Agyapong-Kodua and Weston, 2011, Ajaefobi et al., 2010, Weston et al., 2006). Many causally and/or temporarily related factors may induce these changes and lead to complex and difficult to predict dynamics (Agyapong-Kodua, 2009, Sterman, 2000, Richardson, 1999, Wolstenholme, 1999, Scholz-Reiter et al., 2004). Currently, in factory design scenarios, life cycle engineering of production systems is required to fully understand the changes that are likely to impact the various phases of products, processes and resource systems design (Weston et al., 2009). This is necessary to ensure that MEs:

(i) remain competitive during their lifetime, by realising high value added products in a timely and cost effective way

(ii) extended their useful lifetimes (thereby mitigating investment risks) despite general industry trends

To achieve this, several factory design methods have been reported (Pahl et al., 2007, Kroes, 2002). Clearly, there have been many advances in factory design technologies stemming from the use of mathematical and graphical simulation models (Chung, 1999, Heragu and Kusiak, 1988, Kim and Kim, 2000, Hamann and Vernadat, 1992); artificial intelligence (Mir and Imam, 2001, McKendall and Shang, 2006, Balakrishnan et al., 2003) and other knowledge-based tools (Grubic and Fan, 2010, Todorova and Stefanov, 2009, Fernández López, 1999, Schreiber et al., 1999, Horrocks and Harmelen, 2001, Dean et al., 2002, Grüninger et al., 2010, Fallside and Walmsley, 2004, McGuinness et al., 2000, Ribière and Charlton, 2000). Despite these advances:

1. Existing factory design approaches largely depend on the skill of the designer and can be very expensive. Although this is the case, very good examples of factory designs already exist and when these are aligned with product requirements, lots of time and investment on factory development tools can be saved. This can however be achieved if knowledge related to: 1) products; 2) processes and 3) resources is captured and semantically integrated to generate first stage high level needed factory solutions

2. There is limited interaction between stakeholders of the factory design process leading to long design lead times

3. As a result of (2), decisions are not concurrent, leading to potential errors in factory design decisions

Resource selection ontologies for factory design
4. The causal impacts of products, processes and resource changes on each other can be phenomenal

5. Limited alternative resource analyses is usually conducted rendering the design process ‘static’ because the true capabilities of resources are not carefully modelled.

To support resource systems analyses, literature (Byer and Weston, 2003, Cui and Weston, 2010, Ding and Weston, 2007, FP6-DIFAC, 2006, Ajaefobi and Weston, 2005) reports several techniques for developing conceptual and detailed models of resources. Some of the reported resource systems modelling approaches are based on scientific management techniques; socio-technical systems modelling techniques; approaches centred on modelling human behaviours; enterprise modelling linked to modelling functional elements of resources; developing ‘resource-based views’ as part of a ‘dynamic capabilities approach’; modelling ‘competency resource aspects of individuals’ (CRAI) (Ajaefobi et al., 2010, Weston et al., 2009). Despite the potential of these present-generation resource systems modelling approaches, there is limited application when reasoning about and/or predicting possible future behaviours of resources under changing workload conditions (Agyapong-Kodua et al., 2012).

The impetus for developing the recipe-based factory design approach therefore came from three related observations, namely:

(1) Many complex factory models exist today which address aspects of the factory design process. Also a number of standards and proprietary models attempt to cover the whole development cycle. These standards and models are too complex to be applicable in Small and Medium Enterprise (SME) domain. They are further not available for information sharing between different application providers.

(2) To develop a useful factory model, in general, it is necessary to model (qualitatively and quantitatively) ‘products’, processes’ and ‘resource systems’ aspects of the factory and to consider current and possible change requirements and impacts that arise from these changes.

(3) Current factory modelling tools especially the resource systems toolkits provide modelling constructs that enable modelling manufacturing systems in context but are less advanced with respect to:

(i) Explicit specification of all resource sets: humans, machines and IT systems; more especially integrating the capabilities of these resource components into a unified resource system meeting factory design requirements.

Resource selection ontologies for factory design
(ii) Adjusting virtual resource systems dynamically to changing process requirements
(iii) Computer executing semantically selected resource systems and their behaviours in real production environments.

Based on the above observations on state of the art factory design solutions, this paper proposes that at least first hand virtual factory models can be developed by capturing and semantically describing ‘models of requirements’ of products and processes to maintain consistency in concepts and logic. At a second stage, resource systems are also modelled independently giving close attention to their capabilities and/or competences. Based on these separate models, resources can be selected and matched with the product-process requirements and vice versa. This will enable candidate resource systems with certain attributes to be selected to meet specified requirements. This is what has been described in this paper as the ‘recipe-based’ factory design approach.

As a way of demonstrating this approach, the paper focuses on semantic models of resource systems that can be matched at a conceptual level, then computer executed in support of stages of systems engineering via use of a suitable platform. The paper applies the ontoBroker (Ontoprise, 2011) as the main connecting platform whilst the resource systems are described with the objectLogic language in the OntoStudio environment (Ontoprise, 2011). Further to this, relationships, rules and logics for products, processes and resource integration have been built into the models so that useful results can be achieved by querying the semantic model.

2.0 Some advances in digital factory design methods

Research into product design methods has led to the development of design methods such as Design for Manufacture (Stoll, 1986, Scarr, 1986, Kobe, 1990, Boothroyd and Dewhurst, 1986); Design for Assembly (Boothroyd and Dewhurst, 1986), Design for Disassembly and Recyclability (Kuo et al., 2001), Design for Maintainability (Vujosevic et al., 1995), among others. Most of these methods consist of a set of sequential processes which need to be realised for optimal design output. Clearly, product design impacts on facility design and vice versa, hence product and facility design are causally related.

Factory design consists of the design and implementation of physical facilities and organisational infrastructure in view of realising products (Tong et al., 2003). It therefore consists of the congruent design of processes, resources and flows associated with products. Tong, et al. (2003), however, argue that factory design extends beyond product design requirements and must include the actual

Resource selection ontologies for factory design
Resource selection ontologies for factory design, process definition, sequence selection, organisational infrastructure design and implementation, required for the product realisation process (especially the internal processes) of several types of products in a specialised area. Thus, it is not limited to the specification of product realisation sequences alone but instead the detailed specification of processes, resources and systems required for economic realisation of products (AMICE, 1993). This may therefore involve the extensive consideration of an entire supply chain with key focus on the design and location of internal facilities to help realise enterprise goals. As a result, proponents of business process re-engineering (Davenport, 1993, Bernus and Nemes, 1996, Gunasekaran and Irani, 2010, Vernadat, 2002, Hammer and Champy, 1993) have indicated that it is always necessary to model the conceived business case in a virtual world; experiment and modify resources and flow sequences until optimal solution is obtained (Gunasekaran and Kobu, 2002, Agyapong-Kodua et al., 2009, Agyapong-Kodua et al., 2012). Other researchers have indicated that being able to achieve this in an efficient, timely and cost effective manner requires the use of a coherent set of well experimented modelling tools focussed on products, processes, resources, information, cost, value, etc (Agyapong-Kodua et al., 2009, AMICE, 1993, Chen and Tsai, 2008, Lohse et al., 2006, Agyapong-Kodua et al., 2012).

A number of factory design methods (Iqbar and Hashmi, 2001, Sheth, 1995, Wiendahl et al., 2002) and techniques (Womack and Jones, 2003, Bicheno, 2000, Kosanke, 1996, Hines et al., 2004) have evolved over time with the objective of rapid design, coherent integration of processes and resources, virtual simulations and layout optimisation. These techniques and methods have led to the development of technologies such as applied in 3D and digital factory mock-up tools like Tecnomatix FactoryCAD (Siemens, 2000) for the representation of factory objects and simulation of performance of different arrangements of production resources. Also reported is the use of FactoryFLOW (CIM Technologies 2000) to optimise manufacturing facilities based on material flow requirements, distances between resource centres, cost, material and resource travel times. Other products from Tecnomatix help in the full animation, and ‘walk through’ and ‘inspect’ functions of Engineers. Different ergonomic analysis can be performed with the Mockup and Flythrough software (Siemens, 2000). Other tools from Tecnomatix which provide support for aspects of the overall factory design process are JACK and Factory OPT. Similar applications exist in the Delmia (Dassault Systemes) suite of software for the design of factories. A number of discrete event simulation tools have also aided the design and analysis of facility layouts. Further literature on this can be found in (Rahimifard and Weston, 2007, Chatha and Weston, 2005, Rahimifard and Weston, 2007, Chatha and Weston, 2005,
Ajaefobi, 2004, Agyapong-Kodua et al., 2012, Ajaefobi et al., 2010, Khalil and Weston, 2010). Also immersive virtual reality has been used to provide a new working and assessment environment for the planning and design of factories (Wiendahl et al., 2002).

Further to the above tools, mathematical and artificial intelligence modelling techniques have been used for the design and optimisation of factory layout solutions. Literature has specified the use of artificial neural networks (Chung, 1999, Heragu and Kusiak, 1988, Kim and Kim, 2000); construction heuristics such as CORELAP (Lee and Moore, 1967), ALDEP (Seehof and Evans, 1967), COFAD (Tompkins and Reed, 1976), SHAPE (Hassan, 1994); improvement heuristics such as CRAFT (Armour and Buffa, 1963), FRAT (Khalil, 1973) and DISCON (Drezner, 1987); tabu search algorithms (Chiang and Kouvelis, 1996), simulated annealing algorithms (Chwif et al., 1998, McKendall et al., 2006), genetic algorithms (Pierréval et al., 2003, Dunker et al., 2005, Balakrishnan et al., 2003), slicing tree (Shayan and Chittilapilly, 2004, Wu and Appleton, 2002), space filling curves (Wang et al., 2005), ant colony algorithms (Solimanpur et al., 2005, Baykasoglu et al., 2006) and hybrid genetic algorithms (Mir and Imam, 2001, McKendall and Shang, 2006, Balakrishnan et al., 2003).

Clearly, these mathematical and artificial intelligence methods are suitable for optimising factory designs. The main challenge however is that they do not provide mechanisms for checking poor designs and they sometimes only end up optimising poor designs. It is therefore appropriate to enact methods which will facilitate the development of good factory designs which can be optimised at a later stage.

Another approach commonly reported in the literature deploys process mapping technologies (Womack and Jones, 2003, Bicheno, 2000, Hammer and Champy, 1993, Hines et al., 2004). Serrano et al. (2008) after evaluating the performance of process mapping tools in manufacturing systems design indicated that in broad terms, ‘static’ process engineering tools which can support the design of factories can be broadly classified into flow diagram charts; structured systems and architectural systems. He classified value stream mapping (Womack and Jones, 2003, Duggan, 2003, Lee, 2005, Hines et al., 2004) and all the process mapping tools such as process activity maps (Hines and Nick, 1997, Bicheno, 2000), Customer maps (Bicheno, 2000, Hines and Nick, 1997), product variety maps (Bicheno, 2000), cost flow maps (Bicheno, 2000, Hines and Nick, 1997, Hines et al., 2004), etc as flow diagram charts. IDEF0 (Icam DEFinition Zero) (NIST, 1993), SADT (Structured Analysis and Design Technique) (Marca and McGowan, 1988) and SSADM (Structure Resource selection ontologies for factory design
System Analysis and Design Method) (Ashworth, 1988) were classified as structured systems whilst GRAI (Graphes a’ Resultats et Activite´ s Interrelie´ s) (Doumeingts et al., 1998), CIMOSA (Open System Architecture for CIM) (Kosanke, 1996, AMICE, 1993, Vernadat and Berio, 1999, CEN/ISO, 19440), PERA (Purdue Enterprise Reference Architecture) (Williams, 2002) and the other enterprise modelling tools were classified as architectural systems.

These process mapping and enterprise modelling tools capture processes associated with the development of products but do not directly associate product and process requirements. More critically, they are functional in modelling operation activities and not fully deployed in the factory design phases. Integrated semantic requirements for resource and process coupling are not fully defined. Because most of these tools are research-based, they have had limited applications in industries worldwide.

Most recently, based on the European Factory of the Future (FoF) agenda, some research activities focussed on various aspects of the future factory have emerged. Some with promising outcomes include ‘Modular Plant Architecture’ (FP5-GROWTH, 2000), ‘A configurable virtual reality system for Multi-purpose Industrial Manufacturing Applications’-IRMA (FP5-IRMA, 2010), ‘Digital Factory for Human-Oriented Production System’-DIFAC (FP6-DIFAC, 2006) and ‘Virtual Factory Framework’-VFF (VFF, 2009, Pedrazzoli et al., 2007). Work in these projects have led to the termed “Digital Factory” which is used to refer to the network of digital models, methodologies and applications required for the realisation and integration of activities within the full life cycle of a factory (VFF, 2009). The main objective behind the digital factory model is that perceived operations and controls in a factory must be modelled and experimented until expected key performance indicators such as cost, quality and lead time are favourable.

In the context of achieving this goal, the product-process-resource interfaces and dependencies must be fully understood to enable flexible, fast and reactive integration. In the past, attempts to achieve this have been faced with challenges associated with data which was vast, incompatible, dispersed and complicated. Hence combining, managing and integrating them was difficult. There is therefore the driving need to develop and integrate tools for appropriate data integration and reflection.

Despite these advances, the interplay between product, process and resource design is still under consideration. Most product design activities do not extend to facility design; therefore systems are currently designed separately. More critically, products and facilities can be designed by different
stakeholders without a common database, making appropriate overall design solutions difficult to
obtain. In current manufacturing systems, product designs may be required to fit into existing
manufacturing systems with limited changes to the facilities. This means that knowledge of
available manufacturing resources and their capabilities or competencies must be readily available
for the use of system designers.

3.0 Recipe-based factory design method

From the above review, the authors and their co-teams (COPERNICO, 2010, VFF, 2009) have
perceived that models for efficient design, control and management of digitised virtual factories
require:

1. Explicit characterisation of data with their semantic relationships
2. Support for inter-document references (cross-references) to ensure proper referential
   consistency.
3. Efficient modelling and management of distributed data so support the realisation of (2)
4. Efficient integration of different knowledge domains (Factory, Building, System, Resource,

To achieve this and also help solve some of the problems outlined in section 2.0, researchers on the
Cooperation Environment for Rapid Design, Prototyping and New Integration Concepts for the
Factory of the Future project (COPERNICO NMP-2008-3.4-1 Rapid design and virtual prototyping
of factories) have proposed and tested a ‘recipe-based’ approach to modelling digitised factories.

Apart from the general semantic requirements described through the COPERNICO and VFF
projects (COPERNICO, 2010, VFF, 2009), the authors and their colleagues (Agyapong-Kodua et
al., 2013) have reported that from a methodological perspective, to rapid-prototype a digital virtual
factory, there is the need to (COPERNICO, 2010): 1. capture user requirements; 2. develop a library
of ‘factory-recipe’ solutions; 3. support semantically matching of factory requirements with the
populated factory solutions; 4. conduct further factory simulation and analyses for optimal
performance.

The paper considers therefore that factory design methods have provided useful knowledge sets on
factory layouts and the associated resources, therefore these good examples of layouts and their
resources can be captured and aligned with their production requirements. This will help reduce the

amount of effort required for planning of new or reconfigured factories. By doing so factory design knowledge can be captured and reused to facilitate the design process. This is the thinking behind the recipe based factory design approach. Basically, the concept relies on the derivation of a library of ‘pre-defined layout and configuration recipes’ which are semantically matched carefully to a set of production system requirements, so that based on semantic rules, logics and appropriate matching of ‘concepts’, possible solutions can be pulled from existing databases of recipes and their associated modelling libraries. A factory recipe refers to predefined ‘patterns of solutions’ matching the product and process requirements of a production system. The basic idea behind such an approach is to provide current and future factory designers with abstract descriptions of reusable components (or building blocks) of factories and also allow them to select among predicted suitable sets of resource systems (people, machines and computers). Most importantly, when production system requirements change, these resource systems can dynamically be reconfigured to meet the requirements. To a larger extent, recipes of factory solutions comprise various systems of layouts, people, production and assembly machines, utility systems and computers which are often configured based on different organisational structures, constraints, demand, and data so that they function appropriately to meet production systems requirement.

To ensure the generation of useful patterns of solutions (recipes), a systems engineering approach consisting of the decomposition of production system requirements into ‘lower modular requirements’ termed as ‘sub and sub-sub requirements’, and then matching it with similarly hierarchically decomposed factory recipes, is deployed. This approach has a number of benefits because depending on the level of granularity, different levels of solutions can be achieved. In traditional systems engineering approaches, a ‘v-model’ (Blanchard and Fabrycky, 2008) consisting of the matching of hierarchically decomposed production requirements with a bottom-up solution list through continuous synthesis and evaluation is adopted. The major difference noted in the approach reported here is the equal top-down decomposition of both production requirements and factory recipes as shown in Figure 1. At every stage of the decomposition, results derived from the integrated layers of recipes are synthesised and evaluated and particularly compared with expected indicators of the production requirements.
The recipe-based factory design approach dwells on models and methods required to enable the convergence of meaning across the life cycle of virtual factory development. This is achieved through a common high level semantic language which acts as a communication backbone between product, process and resource sets of data. This backbone is supported by models for the characterisation of equipment resources, recipes for factory designs and a library of best practice examples. The application of these integrated datasets will enable quicker configuration of new or modified factories. The selected factory recipes are connected to a virtual factory environment which generates the full model of the proposed factories for detailed assessment using various expert tools.

The factory recipe approach starts with a classification mechanism which has been largely published by the authors and their colleagues (CHRISTEN-ROSE et al., 2012). This is shown in Figure 2 as ‘cladistics’ data. The cladistics data provides practical classifications of production systems with the objective to systematically organise them. A dedicated tool for the specification and selection of layout types have been developed and published in (Nemeth et al., 2012). The logic is that, after the selection of the factory layout type, data required for full generation of factories and their simulations are collected. This led to the development of a Requirement Engineering tool which seeks to describe product structures and process types. The Requirement Engineering framework is basically a web-based graphical user interface (web-GUI) which is integrated to the semantic engine to support reasoning. Two main results are achieved from the Requirement Resource selection ontologies for factory design.
Engineering tool. Data required for full simulation of the factory is captured and transferred via an enterprise service bus (ESB) to factory simulators. The second part of the results is data describing the product and process requirements.

The schema of the data captured by the Requirement Engineering tool is mapped onto carefully developed semantic data models. These models are built with OntoStudio ontology creator and hosted on the OntoBroker reasoner (see Figure 3). The ontology is used to represent structured data, define rules on the data, and finally query it. To realise the ontology, the semantic reasoner, OntoBroker, is used to load the ontology and also interface with external data sources. They also provide query interfaces and execute rules. The overall architecture shown in Figure 3 describes the various tools that the ontologies needed to interface including the factory recipe library.
The internal representation language used is called ObjectLogic. The structure of OntoBroker can be seen in Figure 4. At the top of the figure the applications connect to OntoBroker. OntoBroker provides several interfaces to clients such as SOAP, native TCP, or Java API. At the bottom the external data sources are collected. OntoBroker maps the data of external systems such as SQL databases or web services to ontology concepts. This way it can perform reasoning on external data. The core reasoner OntoBroker can be extended with many modules and connectors that integrate external data sources, and other kinds of OSGi bundles.
The semantic model contains ontologies for describing products, processes and resources. The resource concepts are built around ‘capabilities’ and through the support of relevant selection rules these resources can be chosen to meet the product-process requirements automatically. The scientific innovation achieved through the deployment of this technique therefore is that based on the specification of product and process requirements, resources of capabilities and competencies meeting the product-process requirements can be selected automatically.

4.0 Illustrative resource selection ontologies in support of recipe-based factory design

The semantic model for recipe-based factory design consists of product, process and production system ontologies as shown in Figure 5. The factory resources are described with the production system ontology. The production system ontologies are interconnected with other concepts in the domain of product and process ontologies. The assumption is that a carefully captured production system requirement can conveniently be assigned to relevant factory recipes so that based on well defined queries, solutions can be generated. In the extended view of the COPERNICO project, these queries will be prompted through a graphical user interface which will also receive the results derived from the semantic model.
The production system ontology consists of concepts for ‘systems layout’, ‘equipment and human resources’ (see Figure 6). The system layout ontology concept describes the various possible layout types whilst the equipment and human resource concepts consist of the potential equipment and human resource skill types, capable of fulfilling the requirements of the systems layout. The capabilities of the concepts related to the production system ontology are described in their ‘attributes’. The connections between different concepts are expressed in ‘relations’.

The topology of manufacturing equipment was defined to consist of Production units; Assembly units; Material handling units; Finishing systems; Preparation systems; Storage systems; Packaging systems; Computers, IT and support systems. They were modelled as shown in Figure 7.
The core semantic resource elements modelled consisted of ontologies describing some manufacturing equipment such as conveyors, robots, bending, drilling, milling, turning, press machines. These ontologies are expandable to include many other resource sets. An ontology for a resource contains ‘concepts’ of some characteristics. For example the PressMachines concept (see Figure 8) has several properties and attributes such as MachineHeight, MachineLength, MachineName, Power, PressForce, OperationType, PressHeadNumber. Some of these properties are for making connection between the PressMachines concept and for interfacing with external databases (named with an addition syllable “Data”, it means this data is from Database). Hence these ontologies can serve two purposes: 1. providing results through direct reasoning from the logics and instances built directly into the OntoBroker or 2. integrating with external databases and providing reasoning through the OntoBroker.
A library was populated and it contains several manufacturing system components such as robots, machine tools (turning, milling, drilling, press–machines etc.), manual workstations, conveyors, human operators, etc. Several geometric and non-geometric data of components – such as name, power, speed, load, range, weight etc. are stored in these databases. For example a snapshot of the HeidiSQL imported “robots” data is shown in Figure 9.
In the case example related to the development of resource selection ontologies for company ABC, an SME located in Sheffield, UK, a database of already existing manufacturing resources was considered. Integrating the domain ontology in the OntoBroker with the MySQL database required that the domain ontology be modified to suit the mapping of the MySQL schema. Usually, the imported MySQL schema appears as a form of ontology as shown in Figure 9. Integration of OntoBroker and MySQL database required that the domain ontology be modified to suit the mapping of the MySQL schema. The imported MySQL schema appears as a form of ontology as shown in Figure 10. This ontology contains all the tables of the database as concepts and the properties of these concepts are the columns of each table in the Database. The connection between this imported ontology and the database is defined by rules. The connection with the domain ontologies had to be defined by mapping properties of database ontology to the corresponding properties of domain ontologies in order to use data from database while querying ontologies. It means when the queries search any data then these information will come directly from the database what could be modified at any time.

Figure 9: Robots table in SQL Database
An example showing how the mapping was achieved for PressMachine ontology is shown in Figure 11. The left side of the mapping view on the picture shows the database ontology as the ‘source’ and the right side shows the domain ontologies of equipment and their properties as the ‘target’.

Resource selection ontologies for factory design
5.0 Resource capabilities and queries

In the semantic model, the capabilities of equipment are detailed in their ‘attributes’. Attributes basically describe the characteristics of the specified concepts. Through the ‘relations’ aspect of the ontology, equipment with specified attributes can be ‘linked’ to specific production requirements and based on semantic rules appropriate selections can be made. As shown in Figure 12, a snapshot of part of the attributes for various equipment is shown. In addition to the attributes, the relations layer is also shown. As can be seen in Figure 12, various capabilities meeting different production requirements are defined. Different equipment types are also modelled and linked with their capabilities. Through the relationship entities, various equipment sets can be assigned to specific product-process requirements.
Resource selection ontologies for factory design

Figure 12: Integrating systems attributes with process descriptions
Some queries were defined with example results shown in Figures 13 and 14. Any number of query restrictions can be defined in arbitrary combinations. The results can be seen on the bottom of the screen shots with each result having a unique identifier which is for the unambiguous identification of data in the SQL database. For example, the OntoBroker can be queried for robots with maximum load capacity of 550kg, 800kg and 420kg and available for use. These queries in the final tool, yet to be reported, will be received through a web-based graphical user interface. The result is as shown in the bottom part of Figure 13.

![Figure 13: Test query and results for robots](image)

Based on a process requirement, a transport activity with minimum width ‘350mm and 200mm’ and ‘available’ shows the two curved-shape conveyor types (see Figure 14).
The previous sections of the paper have described the concepts behind the recipe-based approach to factory design. Within this concept semantic models are embedded for selecting resources meeting product and process requirements. The reasons for selecting OntoStudio and OntoBroker as the modelling environment and platform respectively have been reported in one of the author’s previous publications (Agyapong-Koduă et al., 2013). In this publication, the authors explained that OntoStudio provides adequate modelling formalisms in support of the ObjectLogic language which is flexible in its expression of concepts, attributes and relations. It is particularly useful because of its backbone reasoner and repository, OntoBroker.

Therefore, this paper focussed on demonstrating how ontology-based resource models can be integrated with existing databases of factory resources so that based on their semantically described capabilities, queries requiring specific resources to meet product-process requirements will provide results based on enhanced reasoning through the OntoBroker. Through this approach, designers are

6.0 Observations and conclusions

The previous sections of the paper have described the concepts behind the recipe-based approach to factory design. Within this concept semantic models are embedded for selecting resources meeting product and process requirements. The reasons for selecting OntoStudio and OntoBroker as the modelling environment and platform respectively have been reported in one of the author’s previous publications (Agyapong-Koduă et al., 2013). In this publication, the authors explained that OntoStudio provides adequate modelling formalisms in support of the ObjectLogic language which is flexible in its expression of concepts, attributes and relations. It is particularly useful because of its backbone reasoner and repository, OntoBroker.

Therefore, this paper focussed on demonstrating how ontology-based resource models can be integrated with existing databases of factory resources so that based on their semantically described capabilities, queries requiring specific resources to meet product-process requirements will provide results based on enhanced reasoning through the OntoBroker. Through this approach, designers are
enabled with sets of resources to choose from. In the models presented, 3D models of the resources are linked to the resource list so that they can be applied in simulation environments. This shows therefore that structural models and their encapsulated data can be re-used via the use of suitable ‘in context’ semantic models of types outlined in section 4. It follows also that resource systems can be semantically modelled so that their capabilities can be matched with product and process requirements. The full semantic description of product-process requirements can be found in (Agyapong-Kodua et al., 2013).

The methodology has strength in integrating several data sources and based on appropriate mapping logic, to assign resources matching product-process requirements. As a result, it allows the synergistic use of various kinds of mental, structural and dynamic systems models which facilitate complexity handling and lead to better and faster dynamic analysis of factory models. In view of this, when modelling in context, existing factory data can be semantically matched onto the resource ontologies to provide rapid support to resource selection.

In some other complementary papers (Agyapong-Kodua et al., 2013, Zendoia et al., 2013), the authors describe the use of Requirement Engineering tools to capture initial product and process data. These datasets are saved in a central database which is integrated with the OntoBroker semantic reasoner to support decision making. These extensions were made primarily to provide support for rapid and effective capture, visual representation and validation of resource systems, within any given factory design setting. The purpose for doing so was to create a semantic resource model which can be fleshed out over time and integrated with other databases.

The authors and their colleagues are currently working on the development of graphical user interfaces (GUI), which will now interrogate the semantic model and provide user friendly results at multiple levels.

Acknowledgements

The authors acknowledge the support of the European Commission through the 7th Framework Programme under NMP-2008-3.4-1 Rapid Design and Virtual Prototyping of Factories (COPERNICO; contract number 229025-2). The work reported in the paper has also been supported by the project “Talent care and cultivation in the scientific workshops of BME” that is supported by the grant TÁMOP-4.2.2.B-10/1--2010-0009.
References


Resource selection ontologies for factory design


Resource selection ontologies for factory design


Resource selection ontologies for factory design


NIST 1993. *INTEGRATION DEFINITION FOR FUNCTION MODELING (IDEF0)*. *Federal Information Processing Standards Publications (FIPS PUBS), National Institute of Standards and Technology (NIST)*.

Resource selection ontologies for factory design


Information Technologies, Faculty of Mathematics and Informatics, University of Sofia “St. Kliment Ohridski.


