Association for Information Systems
**AIS Electronic Library (AISeL)**

Wirtschaftsinformatik Proceedings 2007 | Wirtschaftsinformatik

February 2007

# A Service-Oriented Grid-Based Infrastructure for Supporting Virtual Prototyping in Manufacturing

Manfred Grauer
*University of Siegen*, grauer@fb5.uni-siegen.de

Julian Reichwald
*University of Siegen*, reichwald@fb5.uni-siegen.de

Thomas Barth
*University of Siegen*, barth@fb5.uni-siegen.de

In: Oberweis, Andreas, u.a. (Hg.) 2007. *eOrganisation: Service-, Prozess-, Market-Engineering*; 8. Internationale Tagung Wirtschaftsinformatik 2007. Karlsruhe: Universitätsverlag Karlsruhe

ISBN: 978-3-86644-094-4 (Band 1)
ISBN: 978-3-86644-095-1 (Band 2)
ISBN: 978-3-86644-093-7 (set)

# A Service-Oriented Grid-Based Infrastructure for Supporting Virtual Prototyping in Manufacturing

Manfred Grauer, Julian Reichwald and Thomas Barth

Information Systems Institute
University of Siegen
57076 Siegen
{grauer,reichwald,barth@fb5.uni-siegen.de}

**Abstract**

Virtual prototyping in manufacturing gains more and more importance, since companies competing in industry noticed the increasing cost pressure alongside higher quality requirements. The utilization of virtual prototyping techniques provides a possibility to reduce the total development costs and the time to market, while simultaneously innovations of both products and processes increase the overall quality of the manufactured parts. The monetary effort and the lack of know-how especially in small and medium sized enterprises which is required to operate such systems lead to a disadvantage in the more and more globalized market. This paper states the problem of supporting virtual prototyping processes in the manufacturing industry and gives a solution approach by utilizing service-oriented architectures and concepts from grid computing. A pilot implementation of the proposed architecture is introduced and evaluated by case studies in various manufacturing domains.

## 1 Introduction

Companies competing in the manufacturing industry range from small and medium sized enterprises (SMEs) up to global players. They noticed the increasing quality requirements on their products (e.g. allowed tolerances in the product's geometry compared to the specification) alongside the pressure of lowering development and production cost to compete in the market.

Virtual product development, namely the utilization of numerical simulation and optimization techniques, can help to achieve the mentioned business goals. But since the utilization of virtual product development techniques requires dedicated hardware and expensive software systems as well as highly skilled system operators it is nearly unaffordable – especially for SMEs – to efficiently incorporate virtual prototyping in their product research and development. Besides, operating such systems goes far beyond the core competencies of such enterprises and should not be encouraged as an in-house solution.

The grid computing paradigm (s. [BeFH03] or [FoKe04]) has proven to utilize powerful distributed computing and storage resources in different – mainly scientific – scenarios efficiently. Current projects like the Large Hadron Collider Computing Grid [LCG06], dedicated to process and distribute the huge amount of data produced by the Large Hadron Collider which is currently being built at CERN, show the popularity and the acceptance of the approach. Alongside grid computing, service-oriented architectures (SOA) have been of research interest in the past (s. [DJMZ05] or [SiHu05]). The idea of loosely coupling software components while also ensuring a platform-independent and standardized way of communication (e.g. via web services), SOAs are far more flexible than monolithic software systems. Furthermore, process automation can be achieved by utilizing workflow engines which take care of invoking previously specified services in a defined execution order as well as passing necessary arguments to them and receiving their calculation results.

The coupling of service-oriented architectures and grid computing therefore can lead to a highly dynamic approach of interconnecting standardized grid services in an *on-demand* manner, lowering the need (especially for SMEs) to operate costly virtual prototyping resources in-house, but enabling them to access such resources whenever needed (s. www.migrid.de). The effect is a reduction of the total cost of ownership for both IT infrastructure as well as employment costs for the required specialists and a quality improvement for products and manufacturing processes.

This paper is organized as follows: In the next chapter, the process of simulation-based optimization in manufacturing as a basis for virtual prototyping is stated and the problems in supporting such processes adequately by IT systems are figured out. Afterwards, a solution approach based on service-oriented architectures and concepts from grid computing is given and the necessary services for the process support are identified. A pilot implementation of the proposed infrastructure and some of the identified services are given based on de-facto standard middleware for grid computing. The functionality of the implemented system is then evaluated in a case study from the metal casting domain by optimizing a casting process of a gas turbine blade. Additionally, two more scenarios from virtual prototyping in industry are presented: The simulation of a sheet metal deep drawing process and the DMU (Digital Mock Up) kinematics simulation of a deep-drawing process including machinery for transferring parts between different stages of manufacturing. Afterwards, related work is discussed and finally several conclusions and topics for future work are presented.

## 2 Virtual Prototyping – The Problem Presentation

A virtual prototyping process in industrial practice is a lengthy course of action where physical prototypes are built, evaluated, and changed until product and process models are found which suit the predefined needs. Joint multistage optimization problems which commonly appear in the domain of sheet metal forming [GBGR05] or the highly complex tasks in metal casting process optimization [JBRG06] become arbitrarily difficult to solve.

Optimization algorithms coupled with numerical simulations provide a way to apply virtual prototyping on products and processes. The algorithm generates multiple alternative parameter sets of a given input model, thus generating a number of process designs, which are then evaluated. Since a typical model evaluation lasts from hours to days, the designs should be evaluated in parallel for efficiency reasons. At this point, powerful computing resources are required and efficient, scalable optimization algorithms have to be used to assure the highest degree of system load and by that the economic utilization of resources.

The complexity of a virtual prototyping process in manufacturing requires sophisticated know-how and therefore the involvement of various experts (see Fig. 1) and institutions from different problem domains, commonly geographically distributed and in different companies .

- The **domain expert** with the knowledge of the specific manufacturing problem (e.g. metal forming). This person resides in the enterprise and can be seen as an end user or service consumer respectively.
- Specialists from **computational engineering** (having knowledge about the simulation systems used for the numerical calculations as well as the CAD software suites which are used for the geometry models) as well as **experts in numerical optimization** (knowing about the mathematical algorithms). Typically, they are geographically dis-

tributed and do not necessarily belong to the same enterprise. Since these specialists offer their know-how to the end user, they can be seen as service providers in the overall process.

- A **telecommunication engineer** who operates and maintains the required IT infrastructure (hardware, software, local/global networking to handle communication between collaborating partners) and acts as a provider of telecommunication services.
- **Software integrators** which try to integrate domain-specific (e.g. CAD/CAE tools from computational engineering) and generic enterprise functionalities (e.g. Customer Relationship or Supply Chain Management systems) across organizational and technical boundaries in a distributed environment.

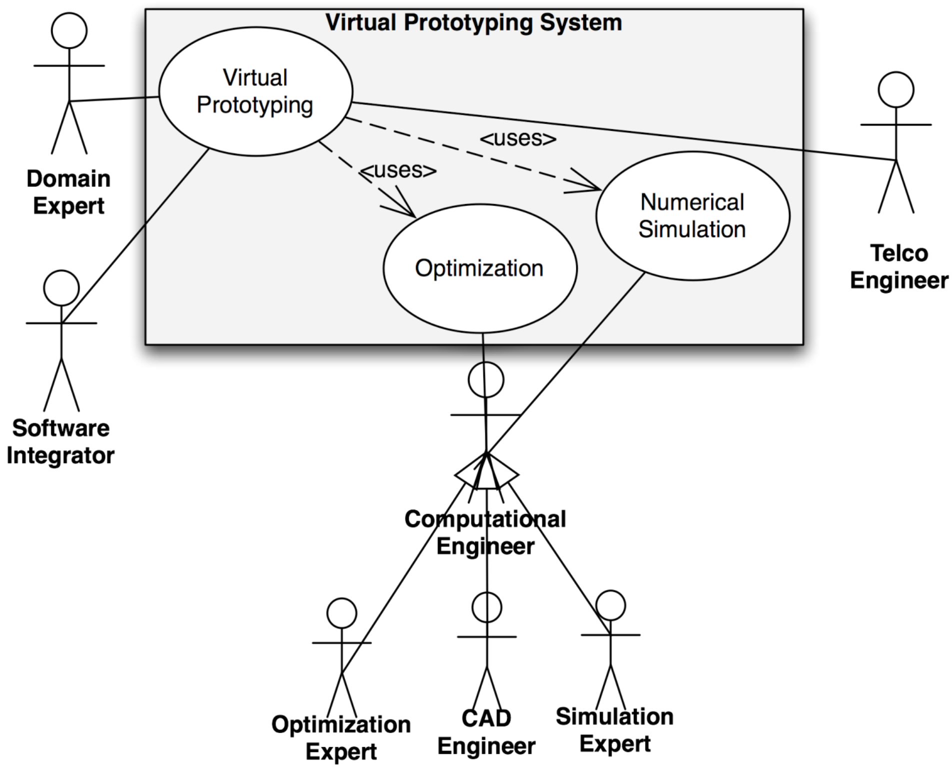


Figure 1: UML Use Case diagram of the incorporated actors in a virtual prototyping process.

As an example, a typical process in the product and process design in manufacturing, especially in the metal forming domain (e.g. sheet metal forming like deep drawing, metal casting), depicted in Fig. 2, will be used and analyzed here. The process can be divided into five phases: In the **first phase**, the real-life metal forming process has to be mapped to a process model, incorporating the specific details of the process (e.g. geometry, materials etc). For this task, not only the domain expert with the appropriate know-how of the real-life process has to be involved.

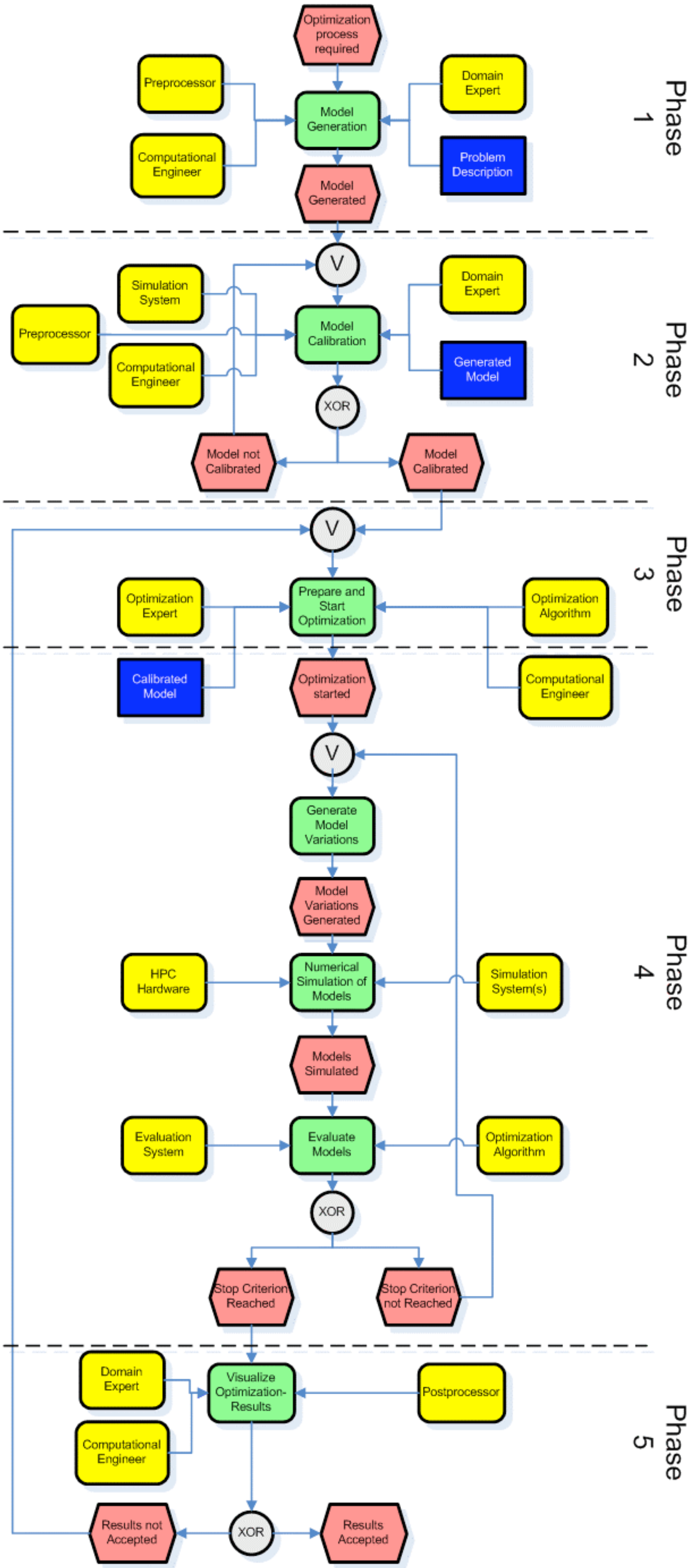

Figure 2: Core Phases of virtual prototyping presented for industrial forming by its business process

Moreover, a computational engineer having knowledge of the simulation software package and the preprocessing tools is needed. When the initial model creation is completed, the **second phase** is entered. The model needs to be calibrated, i.e. model parameters will be adjusted and single simulation runs are made to verify that the model reflects the reality. Again, the domain expert and computational engineer are required in close cooperation to calibrate the previously generated model. Since the calibration process is particularly difficult, the model is continuously checked if it suits the needs. Hence, the output of single simulation runs is permanently reviewed by all participants. The calibration process iterates as long as the model is not yet adequately calibrated. During the **third phase**, preliminary jobs (e.g. transferring the calibrated models to the compute nodes, setting algorithm parameters etc.) for preparing the optimization run are performed. The optimization sub-process in **phase four** is entered next. The optimization algorithm now takes care on generating a number of model parameter sets which need to be simulated and evaluated. As long as the stopping criteria of the optimization algorithm is not yet reached, this sub-process iterates until the model is sufficiently solved. **Phase five** consists of the post processing which incorporates the collaborative viewing between the domain expert and computational engineers. If the resulting model is not sufficient, the optimization cycle is re-entered.

The variety of prerequisites (software systems and competencies) in a geographically distributed scenario in each phase of the process is a challenging task for a sufficient software support. An infrastructure that adequately supports the whole process should be able to handle the tight cooperation among the involved participants. Collaboration aspects have to be respected as well as discovering and handling distributed data-, hard- and software resources. Furthermore, the model parameterization and generation needs to be supported in a generic way so optimization algorithms as well as metal forming simulation packages may be coupled together arbitrarily and in a flexible way without adjusting the code.

Since the whole process is geographically distributed and inter-organizational, heterogeneous resources and various technical (possibly proprietary) infrastructures are typically involved. The problem solving environment has to take notice of that and offer a way to interconnect the enterprises using open standards for communication, hence leading to an open, extensible system.

## 3 Solution Concept for the Virtual Prototyping Grid-Environment

The specific requirements for the supporting IT system are directly derived from the process shown in figure 2. First, the environment needs to be able to provide the computing capacities of high-end workstations and clusters over a limited period of time. This compute power must be available on demand without remarkable delays. Furthermore, the user must be able to monitor the current state and get intermediate results while the optimization is in progress. To prevent the general distribution of proprietary data and the implied risks, secure data transfer and remote file handling is mandatory. Alongside the computing infrastructure, the technology has to provide support for collaboration, so the geographically distributed experts are able to set up the initial models, calibrate them or discuss the results. Finally, it must be possible to adopt the overall process to changes (e.g. switch to a new simulation system or another optimization algorithm). In phases one, two and five, collaborative aspects between the involved parties dominate the process. Collaborative aspects and grid process generation is already discussed in [FSFR06], so infrastructural topics will be the main focus here.

The utilization of common hard- and software systems showed to be inapplicable to provide an acceptable solution strategy for the aforementioned scenario in the metal forming industry. The excessive runtime of numerical simulations require the utilization of high performance multi-processor machines or compute clusters, which exceed the financial possibilities of SME´s. The transformation from fixed costs into variable costs by outsourcing such simulation jobs seem to be an adequate solution, alongside letting the enterprises concentrate on their core competencies. The spatially distributed process across different enterprises therefore requires a supporting IT infrastructure taking notice of the different hard- and software systems as well as the flexible combination of the components.

Service-oriented architectures built with standardized web services while also adopting concepts from grid computing provide a promising way out of the dilemma [BHMN04]. The loose coupling of components provides the high degree of flexibility needed to support the process. Services, which can be seen as single components in the overall software system, are provided by enterprises specialized in this field. These enterprises have to care about the proper functionality of the service they offer as well as service updates or availability. The business logic, knowledge and complexity is encapsulated and hidden by standardized, self-describing interfaces, which can be accessed by service consumers via standard internet connections and by using standard protocols and communication techniques.

The utilization of workflow engines can even more increase the degree of flexibility. By deploying whole workflows as a web service and providing them to others, complex service compositions and choreographies can be reused by other services or workflows. The main characteristics of a web service based service-oriented architecture adopting the grid paradigm lead to a network of services which provides the core services for the aforementioned simulation and optimization process, which are namely

- **File transfer services** which are potentially needed in all phases of the mentioned process, since input models, result files etc. have to be transferred to and from the compute nodes.
- **Services providing the appropriate optimization algorithms**. As stated in [GBGR05], direct search methods capable of constraint handling and with good scalability behaviour in distributed environments are a preferred choice here to ensure the highest possible degree of resource utilization.
- **Services providing the numerical simulation and evaluation**. Since most simulation packages consist of native, closed-source legacy code compiled for a predefined hardware architecture, a way of accessing such software packages in a service-oriented way according to the underlying technology has to be found.

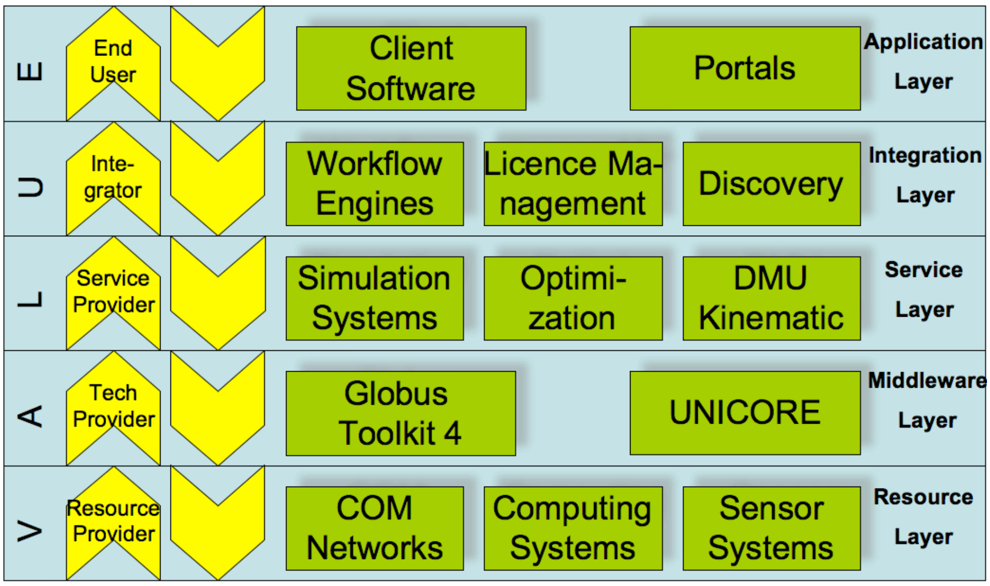


Figure 3: A service-oriented grid-based infrastructure and the corresponding value chain [Por85]

Figure 3 shows the proposed infrastructure split into five layers, according to the key personnel and institutions which are responsible for the system to work. At the same time the corresponding value chain [Por85] is indicated. In higher layers, the amount of value added for the end user is increasing. Since the scope of this paper is focused on applications and the virtual prototyping process, infrastructural details like security aspects, billing, accounting, licensing or service-level agreements are not considered here.

## 4 Implementation of the Virtual Prototyping Grid-Environment

To adequately provide a support for the aforementioned process, a grid-based service-oriented infrastructure was designed and implemented. The Globus Toolkit (GT4) [Glob06] is the de-facto standard for building Web Service Resource Framework (WSRF) based grid infrastructures and was chosen as the underlying middleware. GT4 allows the utilization of standard grid services (e.g. the Monitoring and Discovery Service (MDS) or Reliable File Transfer (RFT)), but also the implementation of custom and – due to WSRF – stateful grid services. Since the file transfer and discovery services are already present in the infrastructure, the services for the simulations and the optimization algorithms have to be implemented. Since GT4 is (in its core parts) implemented in Java, the simulation and optimization services were also implemented in that language. First steps of the implementation were already presented in [GBGR05], where a test problem (representing the mixed-integer multistage optimization problem in sheet metal forming) could be optimized by using a client which realized both the optimization algorithm as well as the workflow implementation. A comparison of the problem solving time by using Distributed Polytop Search (DPS) on the one hand and the sequential quadratic programming (SQP) implementation of Matlab® has been made (see table 1). The table indicates the feasibility of the SOA-implementation and a good scalability of the DPS-method in distributed sys-

| **Optimization method** | **# of CPUs** | **Optimal objective function value *f(x*,4)*** | **Number of objective function evaluations** | **Wall-clock time [seconds]** |
|---|---|---|---|---|
| SQP (Matlab®) | 1 | 26.9497 | 3166 | 47.16 |
| DPS on SOA Architecture | 1 | 26.9842 | 19868 | 927.63 |
| | 10 | | | 87.85 |
| | 50 | | | 17.32 |
| | 100 | | | 10.83 |
| | 200 | | | 6.24 |

Table 1: Comparison of DPS (SOA) and SQP (Matlab) when solving the test problem for a mixed-integer multistage sheet metal forming problem [GBGR05], using a service-oriented implementation of DPS

tems. Nevertheless, results based on the aforementioned prototype should be seen as proof of concept, and not a comprehensive study on scalability or the quality of optimization algorithms.
Next, numerical simulations for manufacturing problems were implemented, furthermore the optimization method has been deployed as a web service as well to allow arbitrary interconnections. Implementation details for the optimization service and for a metal casting service (as one example for a simulation service) are given in sections 5.1 and 5.2.
The whole infrastructure has been deployed on a 296-CPU cluster computer, consisting of 148 compute nodes with 2 CPUs, 2 GB main memory and 80 GB hard drive per node. The operating system for each node is SuSE SLES9.3. A redundant head node as well as a redundant storage node is used for cluster administration, control and storage access. The nodes are interconnected by Gigabit Ethernet.

### 4.1 The Optimization Service

This service (s. Fig. 3, "Service Layer"), is an implementation of the distributed polytop search (DPS) which belongs to the class of direct search method (s. [Wri95], [BeTs89]). The DPS was designed regarding efficiency and scalability in distributed systems. During its runtime, it requires an a priori unknown number of evaluations of both the objective function and corresponding constraint functions. The service has to save its state each time an evaluation request occurs, and it passes the data set which is to be evaluated immediately back to the service caller instead of directly invoking the simulation service. This behavior opens the possibility to use workflow engines to keep control over the whole workflow execution. Considering these conditions, the service makes use of the possibilities provided by WSRF. Being instantiated following the factory design pattern, the instance service operates on a set of resources (s. Fig. 3), allowing the service to keep its actual state, even when it is actually not in use. Furthermore, internal state variables checkpoint the calculation state, so the algorithm can be set on hold and resume at a given place in the code [GBGR05].
Besides a service operation which allows a client to set necessary parameters needed by the polytop search, the grid service operation `iterate(IterateRequest)` takes care of starting and resuming the algorithm at the appropriate position - according to its internal state and according to the input data inside the `IterateRequest` data structure. A resulting data set is returned immediately after invoking the operation, telling the client if further evaluations are needed or if the DPS reached a predefined stopping condition.

## 4.2 The Simulation Services for Metal Casting and Sheet Metal Forming

Since the implementation of the sheet metal forming service is analogous to the metal casting service, the casting service is described in detail as an example. The main purpose of this service (s. Fig. 3, "Service Layer"), is to wrap the metal casting legacy software CASTS® [LaNS98] as a Grid service. However, the CASTS Service does not only provide a service-wrapped version of CASTS, but it also takes care of the following operations: It is capable of modifying the input model of the casting process according to a set of parameters passed to the service. This parameter set can be the input received from optimization services (e.g. the DPS implementation mentioned above). The service executes the CASTS legacy application on a number of different execution platforms. Since a cluster computer is used in this case, a simulation request is leading the internal execution subsystem to incorporate the local resource manager Torque [TORQ06] and the scheduling system Maui [MAUI06]. The execution state of the cluster job is monitored and exposed by the Casts Service. The execution subsystem is highly modularized so that the service also works on single workstations without local queuing/scheduling. The service also provides functionality to evaluate the simulation result (which is done by CritCASTS, a legacy software system bundled with CASTS) and determining the objective function value as well as the constraint function values, needed for simulation-based optimizations. The utilization of WS-GRAM [Glob06] for the job execution was discouraged because of the aforementioned extra functionality which goes beyond the capabilities of WS-GRAM and leaving the process logic implementation up to the service consumer, which presupposes inexistent know-how.

In the next section, the aforementioned prototypical implementation is used to demonstrate the feasibility of the approach when applying it in the domain of metal casting. Furthermore, additional industrial scenarios from sheet metal forming are described focusing on optimal product (section 5.2) and process design (section 5.3).

## 5 Case Studies from Manufacturing and Results

### 5.1 Virtual Prototyping in Metal Casting

As an example, a casting process of a gas turbine blade was used to evaluate the implemented infrastructure (see figure 3). The evaluation of the simulation results have been done from the following points of view [JBRG06]:

- The probability of local freckle (shrinkhole) information at the surface of the turbine blade. Freckle probability was estimated based on the temperature gradients calculated by the simulator,
- the degree of curvature of the solidification front. It should be as horizontal as possible in order to achieve a high quality directional solidification,
- the ratio G/v (temperature gradient over solidification speed) must be greater than a critical value and
- the process time.

Goal of the optimization was an improved withdrawal profile (temperature gradient over solidification speed) for the casting process of the turbine blade. The withdrawal process is parameterized by eleven constrained design variables representing withdrawal velocities.
The optimization algorithm service has been parameterized to utilize 16 CPUs by generating 32 new withdrawal sets in each iteration.

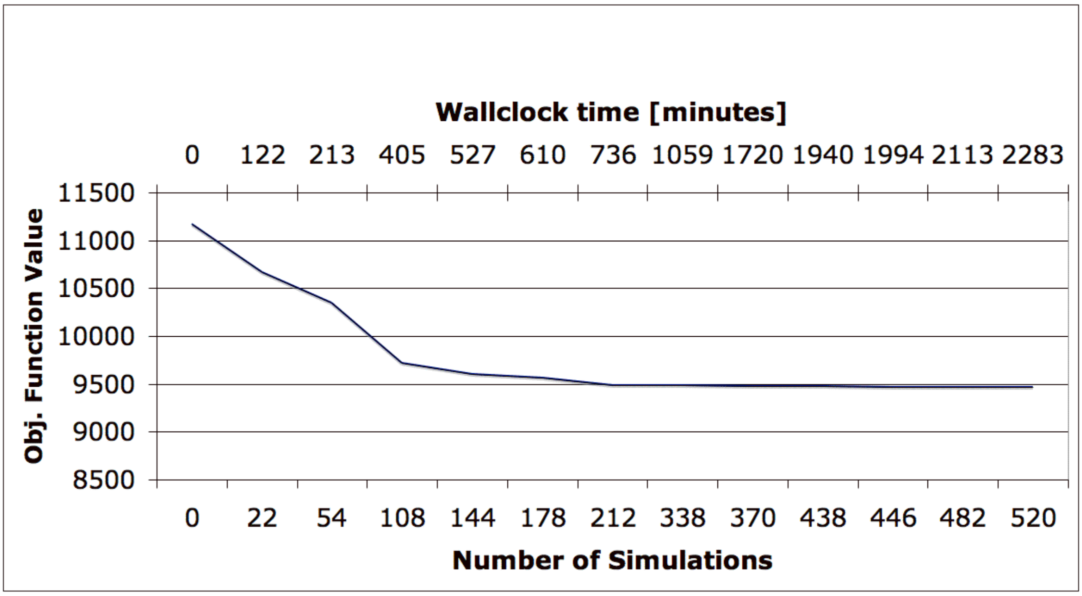


Figure 4: Objective function value in dependence to the total number of simulations and the total time in minutes

Figure 4 shows the objective function in dependence to the total number of simulations needed to yield the function value. Since the global minimum of the objective function is unknown, the calculated quality has to be interpreted by metal casting experts. Nevertheless, an optimization achievement can be noticed by reducing the objective function by 15.24%.

## 5.2 Virtual Prototyping in Sheet Metal Forming

Sheet metal forming copes with the forming process of solid state metal sheets, where the forming process aims on form, surface, and material properties. Risks in planning the manufacturing process can be predicted and reduced immediately and materials can be used in a more efficient and cost-effective way. The software system FETI-INDEED® [KWRD01] has been utilized to simulate deep-drawing processes with the ability of partitioning the assembly parts as well as friction observance and rezoning (mesh refinement during simulation runtime).
As an example from sheet metal forming, the calculation of a cylindric cup for the automotive industry was calculated with FETI-INDEED.
As can be seen in figure 5, the cup was partitioned into 16 parts by using the FETI method. A higher processor usage leads to a decreased processing time from 44:34 hours (2 CPUs) down to 06:41 hours (36 CPUs).

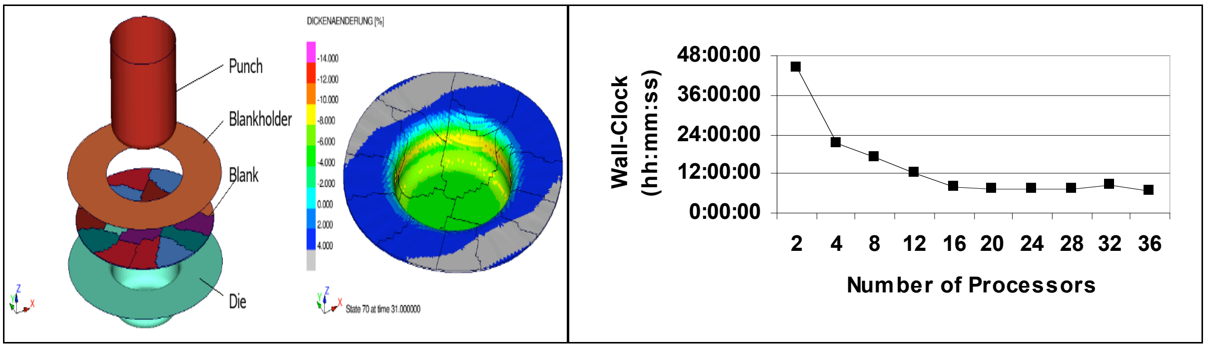


Figure 5: Model of a cylindric cup, subdivided into 16 parts, calculated with FETI-INDEED and according wall clock graph by varying the number of CPUs.

## 5.3 Virtual Prototyping of DMU Kinematics

Beyond sheet metal forming simulations, kinematics of the production process itself and the behavior of automatic transfer equipment can be simulated and visualized ([Cu05], [SchMe06]). The superposition of movements by transferring assembly parts to the next working stage and

the automatic transfer equipment can reduce the overall time needed for the whole process to avoid collisions, which depends on various parameters (e.g. transfer speed, stamp speed, angle etc). A trade-off between collision-free production processes and a maximum turnout has to be found. By utilizing such dynamic simulations, the component delivery to the next working station can be visualized and – if collisions appear – the parameters and tools can be adjusted early in the overall production process to achieve a collision-free manufacturing process again.
Figure 6 shows a simulation of a sheet metal forming press with a 630 tons maximum pressure, 4000 mm x 1700 mm table size and 500 mm maximum travel. Transfer equipment for assembly part movement between the working stations has been applied to the press.

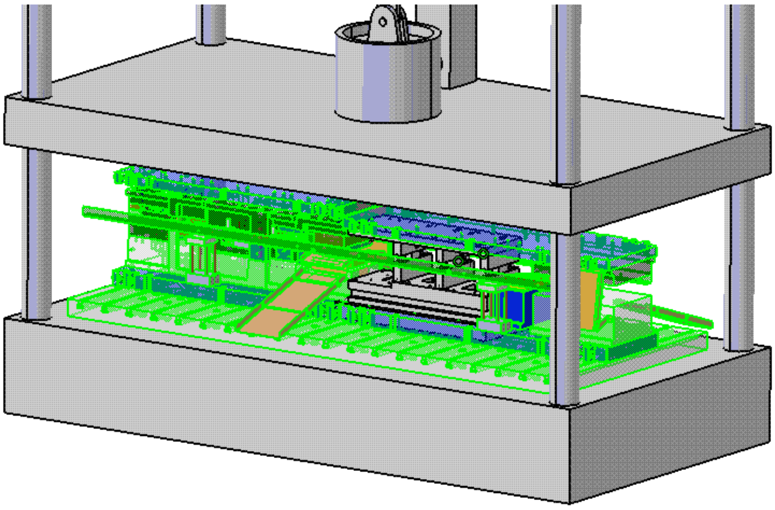

Figure 6: DMU kinematic-simulation of a sheet metal forming press and corresponding transfer equipment, visualized by CATIA V5 [SchMe06]

## 6 Related Work

Supporting business processes with software systems and especially service-oriented architectures realized with web services have received considerable attention in both academia and industry. Several other research projects try to cope with similar subjects in related fields.
The **Geodise** project [XSCK04; SONK04] focuses on optimization, design and fluid dynamics, especially in aerodynamics. Its main goal is to provide a distributed problem solving environment (PSE) for engineers working in the mentioned fields by utilizing e.g. MATLAB and adding Grid functionality to it. Although first Geodise implementations were based on Globus

Toolkit version 2, the core Geodise Toolbox is now part of the managed program of the Open Middleware Infrastructure Institute (OMII) [OMII06].

A **Grid-enabled problem solving environment for engineering** in design where distributed parties are able to collaborate has been introduced by Goodyer et al [GBJS06]. The system enables its users to start Grid jobs on Globus Toolkit based hosts, but the main focus is put on collaborative application steering and result visualization using the gViz library [BDGS04] instead on automatic or semiautomatic simulations (or virtual prototyping respectively).

The **P-GRADE** Portal [SiKa05] aims to be a workflow-oriented computational Grid portal, where multiple clients can collaboratively participate in design, development and execution of a workflow as well as multiple grids may be incorporated in the workflow execution. The P-GRADE Portal is based on Globus Toolkit version 2 for basic grid operations (such as file transfer and job execution), the workflow execution is done by a proprietary implementation. But since P-GRADE does not rely on web service based grid infrastructures, it can be stated that P-GRADE is a system based on proprietary standards and therefore will not provide the high degree of flexibility needed to adequately support the aforementioned virtual prototyping process in an adequate way.

Summarizing, the presented approaches emphasize either technical aspects from domain-specific numerical simulation or workflow management in engineering. Hence, the solution approach presented in this paper is focused on the integration of domain-specific applications from computational engineering, workflow management in engineering, and aspects from enterprise application systems as well. This integration provides the capability to consider technical as well as business aspects simultaneously.

## 7 Conclusions and Future Work

In this paper, the need for utilizing virtual prototyping techniques in manufacturing industry was shown and the complexity of a virtual prototyping process was indicated with all of the incorporated institutions, key actors and virtual prototyping resources (i.e. hard- and software). A solution approach of a layered service-oriented grid-based infrastructure and the according value chain was proposed to adequately support the virtual prototyping process in manufacturing enterprises. A prototype implementation was presented and computational results from the problem domains of metal casting, sheet metal forming, and digital mock-up kinematics were shown.

As future work, the single simulation- and optimization runs should be extended and coupled such that a complete production process chain can be simulated in by the system. Beyond the optimization of a process chain the design of the optimal layout of a complete manufacturing cell, a production line comprising multiple manufacturing cells, and finally optimizing the layout of the whole factory is envisaged.

## Acknowledgements

Parts of the work presented in this paper are partly supported by a grant from the German Ministry of Education and Research (BMBF) (D-Grid initiative, InGrid Project) and by the Ministry of Science and Research of Northrine-Westfalia out of the MiGrid-Project. Furthermore we would like to thank our industry partner ACCESS e.V., Aachen, GNS mbH Braunschweig and Co.Com Concurrent Computing GmbH, Siegen for providing both the simulation software systems and the dedicated know-how.

## References

[BeFH03] Berman, F.; Fox, G.; Hey, T. (eds.): Grid Computing: Making the Global Infrastructure a Reality, J. Wiley & Sons, Chichester 2003

[BeTs89] Bertsekas, D., P.; Tsitsiklis, J. N.: Parallel and Distributed Computation – Numerical Methods, Prentice-Hall, Englewood Cliffs 1989

[BDGS04] Brodlie, K., Duce, D., Gallop, J., Sagar, M., Walton, J., Wood, J.: Visualization in Grid Computing Environments. In: Proc. of IEEE Visual., pp. 155-162, 2004

[BHMN04] Booth, D., Haas, H., McCabe, F., Newcomer, E., Champion, M., Ferris, C., Orchard, D.: Web Services Architecture, W3C Working Group Note, http://www.w3.org/TR/ws-arch (visited 2006/07/24)

[Cu05] Culha, B.: Automatic Arrangement of Product Parts, Assemblies and Modules, in: Bouras, A.; Gurumoorthy, B.; Sudarsan, R. (eds.): Product Lifecycle Management PLM`05, Inderscience, Geneve, 2005

[DJMZ05] Dostal, W.; Jeckle, M.; Melzer, I.; Zengler, B.: Service-orientierte Architekturen mit Web Services – Konzepte-Standards-Praxis (in German), Elsevier Spektrum, München 2005

[FoKe04] Foster, I.; Kesselman C. (eds.): The Grid 2: Blueprint for a New Computing Infrastructure, Morgan Kaufmann 2004

[FSFR06] Friese, T., Smith, M., Freisleben, B., Reichwald, J., Barth, T., Grauer, M.: Collaborative Grid Process Creation in an Engineering Domain". In: Proc. of the $13^{th}$ IEEE Conf. on High Perf. Comp., Bangalore/India 2006 (to be published)

[GBJS06] Goodyer, C., Berzins, M., Jimack, P., Scales, L.: A Grid-enabled problem solving environment for parallel Computational Engineering Design. In: Advances in Engineering Software, 37:439-449, 2006

[GBGR05] Grauer, M., Barth, T., Gerdes, M., Reichwald, J.: On distributed virtual prototyping in the automotive supplier industry using service-oriented computing. In: Hahn, A. and Grauer, M. (Eds): Informations- und Wissensdrehscheibe Produktdatenmanagement (in German), GITO, 2005

[Glob06] globus.org: The Globus Alliance, http://www.globus.org (visited 2006/07/24)

[JBRG06] Jakumeit, J., Barth, T., Reichwald, J., Grauer, M., Friese, T., Smith, M., Freisleben, B.: A Grid-Based Parallel Optimization Algorithm Applied to a Problem in Metal Casting Industry, BIOMA 2006 (submitted)

[KWRD01] Kessler, L., Weiher, J., Roux, F.-X., Diemer, J.: Forming Simulation of High and Ultrahigh Strength Steel using INDEED with the FETI Method on a Workstation Cluster. In: Mori, K. (ed.) Proc. of NUMIFORM'01, Balkeman Pub., pp. 399-404, 2001

[LaNS98] Laschet, G., Neises, J., Steinbach, I.: Micro- Macrosimulation of casting processes. 4ieme école d'été de Modélisation numérique en thermique, 1998

[LCG06] cern.ch, Large Hadron Collider Computing Grid Project, http://lcg.web.cern.ch/LCG (visited 2006/07/24)

[MAUI06] clusterresources.com, Maui cluster scheduler, http://www.clusterresources.com/pages/products/maui-cluster-scheduler.php (visited 2006/07/24)

[OMII06] omii.ac.uk. Open Middleware Infrastructure Institute (OMII). http://www.omii.ac.uk. (visited 2006/07/24)

[Por85] Porter, M.: Competitive Advantage, New York, Free Press, 1985

[SchMe06] Schuth, M., Meeth, J.: Bewegungssimulation mit CATIA V5. Hanser, 2006

[SiHu05] Singh, M., P.; Huhns, M., N.: Sercive-Oriented Computing – Semantics, Processes, Agents, Wiley 2005

[SiKa05] Sipos, G. and Kacsuk, P.: Collaborative workflow editing in the P-GRADE. Proceedings of the MicroCAD Conference, Miskolc, Hungary, 2005

[SONK04] Song, W., Ong, Y.-S., Ng, H.-K., Keane, A., Cox, S., Lee, B.: A service-oriented approach for aerodynamic shape optimization across institutional boundaries. In: Proceedings of ICARCV, China, Kunming, 2004

[TORQ06] clusterresources.com, Torque Resource Manager, http://www.clusterresources.com/pages/products/torque-resource-manager.php, (visited 2006/07/24)

[XSCK04] Xue, G., Song, W., Cox, S., Keane, A.: Numerical Optimization as Grid Services for Engineering Design. In: Journal of Grid Computing, 2(3):223-238, 2004

[Wri95] Wright, M.: Direct Search Methods: Once scorned, now respectable. In: Griffiths, D.; Watson, G. (eds.), Proc. Dundee Biennial Conf. in Num. Analysis, pp. 191 - 208, Addison-Wesley, 1995