SSOAR

Open Access Repository
www.ssoar.info

Towards the integration of digital library applications via web services for use in a scientific workspace

Schmiede, Rudi; Stoll, Julia; Körnig, Stephan

Veröffentlichungsversion / Published Version
Sammelwerksbeitrag / collection article

Empfohlene Zitierung / Suggested Citation:
Schmiede, R., Stoll, J., & Körnig, S. (2004). Towards the integration of digital library applications via web services for use in a scientific workspace. In A. Nase, & G. v. Grootel (Eds.), *Putting the sparkle in the knowledge society: 7th International Conference on Current Research Information Systems* (pp. 125-137). Leuven: Leuven Univ. Press. https://nbn-resolving.org/urn:nbn:de:0168-ssoar-255803

Nutzungsbedingungen:
Dieser Text wird unter einer CC BY Lizenz (Namensnennung) zur Verfügung gestellt. Nähere Auskünfte zu den CC-Lizenzen finden Sie hier:
https://creativecommons.org/licenses/by/4.0/deed.de

Terms of use:
This document is made available under a CC BY Licence (Attribution). For more Information see:
https://creativecommons.org/licenses/by/4.0

gesis Leibniz-Institut für Sozialwissenschaften

Mitglied der Leibniz-Gemeinschaft

Towards the Integration of Digital Library Applications via Web Services for Use in a Scientific Workspace

Julia Stoll, Stephan Körnig, Rudi Schmiede
Darmstadt University of Technology, Germany

Summary

Our project "Generic Portals" [1] focuses on the possibilities and problems created by the usage of web-technologies to support web-based Digital Library applications. We observe a divergence of services in the sphere of structured resources and scientific libraries´ services. Publications in the form of digital resources have introduced new dynamics into the information exchange in the scientific communities. The changes in the publication process are formed by so-called structured resources, whereas the resource can be seen as an encapsulated object. For the exchange of information we have only to know the interface of the structured resource in detail. W3C has published a working draft which includes a language named Web Service Description Language (WSDL) for describing interfaces of services. A Web service interface is a shared connection between functional components of applications. We consider the requirements for the integration of DL-applications via Web services. We present an adapted process model for supply of DL-oriented Web services. Further we present a component-based approach to relate researchers´ activities to so-called DL-oriented Web services. We obtain criteria for the description of DL-oriented Web services based on the researchers´ working process. Finally, we present a brief overview what kind of Web technologies might be reused for a prototypical implementation of DL-oriented Web services.

1 Starting Position

Our project "Generic Portals" focuses on the possibilities and problems created by the usage of web-technologies to support web-based Digital Library applications (DL-applications). We especially look at Web services, for exchanging scientific results between disciplines all over the world, at any given time and place. We observe that the exchange of information, forms of discussion and the process of distributing scientific information have changed over recent years. We investigate the working process in the scientific community, which has adapted its communication to the use of Web technologies. We consider the applications of the World Wide Web (WWW), including search engines, and the recent developments to

[1] Project of Darmstadt University of Technology within the competence network „New Services, Standardization, Metadata" (bmb+f, i.e. the Federal ministry of education and research)

realize new Web services for DL-oriented applications, to support DL-services. On the one hand various DL-applications are realized and located in different organizations, institutions and universities. Many of these kinds of DL-applications are located in a library itself where services are only accessible on-site. Instead, we observe a divergence of services in the sphere of structured resources and scientific libraries´ services.

2 Changes in the Scientific Communication Process

2.1 Social and Technical Publication Processes

Scientific discussions constitute a social process. Doing research is based on discussing problems, defining tasks, finding solutions and exchanging results. Discussions are based on informal communication between researchers and cooperative teams. Cooperation is needed for answering complex questions. Internet-techniques are used worldwide for exchanging information and publishing research results on the WWW. Web-based joint-activity tools, like e-mail and search engines enable every researcher to look for recently published digital resources. A researcher is accepted in a scientific community, when other researchers read and refer to his/her scientific results. Furthermore, a member of the (scientific) community is defined by his/her activities. Publications in the form of digital resources have introduced a new dynamic into the information exchange of the (scientific) community. Such publications are available in different forms and places. Differences can also be observed in the process of publication. We identify two streams in the publication process for distributing scientific results. One of these methods can be characterized by the term "review-before-commit" and the other method by "review-on-commit". The method of review-before-commit describes the common way of publishing as done by publishing companies. The document is "closed" when it is printed/presented. The method "review-on-commit" is well established in the Open Source community (Raymond, 1999). The modern resource becomes a dynamic "object" and its publication a dynamic process. Using the method of "review-on-commit" the modifications are made by the author and/or by the reviewer in the existing document. The document is immediately presented again on the WWW. In principle, evaluation is performed throughout the distribution process. The process of publishing is "open" and thus visible to every researcher who is interested in the document in form of a structured resource (Stoll et al. 2003).

2.2 Structured Resources and Their Services as Encapsulated Objects

The changes in the publication process are formed by the definition of so-called structured resources. A structured resource is an encapsulated object, where the definition or description of its structure is separated from its content; the content

can be text, music etc. The most prominent document description language is XML (W3C - XML 2000). In practice we apply a so-called "black-box" concept. The approach of encapsulated objects is necessary for the exchange of information between different interfaces. To apply such concepts, we have to implement a syntactical description of the structured resource in form of an interface for encapsulating the information. This is the pre-requisite for the exchange of structured resources. For the exchange we do not need knowledge of the content in detail. On a comparable level of abstraction, we might consider services. In particular, Web services are designed for the integration of different Web applications, where the request and response of such services is seen as an encapsulated object. The W3C has recently[2] published a working draft which includes a language definition called Web Service Description Language (WSDL) (W3C 2003a). Using WSDL a service is defined by the definition of an interface and a textual description of this service. A Web service interface is a shared connection between functional components of applications (or (sub-)systems) defined by specific entities (W3C 2003b). This means we deal with the (sub)system-to-(sub)system communication based on a generalization of the communication process between existing web-based applications.

3 Requirements for the Integration of DL-Applications via Web Services

As already mentioned we observe a divergence of services in the sphere of structured resources and scientific libraries´ services. Information retrieved from an information system for distribution via the Web has to be an encapsulated object which is accessible by a user interface, like a browser. The distribution and the access of structured resources can be seen as DL-oriented services. Further the search of structured resources is a DL-oriented service. This means several DL-applications – like harvesting metadata, searches using metadata etc. – can also be seen as DL-services. In order to integrate these DL-services in a common scientific working environment we need specific "middleware". The development of the Web technology standards departs from the idea of middleware; in our context middleware is software, which supports the technical communication infrastructure between different applications. Middleware has to be platform independent. Concerning DL-applications, we aim to apply the new standards of Web services for the development of a specialized web-applicable DL-oriented middleware. As mentioned, the interfaces in form of encapsulated services have a certain relevance. In order to specify DL-oriented Web services we have to know the interfaces of typical DL-applications.

[2] 11 June 2003

We relate the existing DL-applications to their IT-system based realizations and to the possible use of Web services, and we determine the foundations of software systems, when we apply a component-based design. We study the researcher´s activities concerning the exchange of information in a scientific community. Additionally, the activities identified lead us to requirements, which we have to consider, if we implement DL-oriented Web services. Thus Web services can also be seen as encapsulated objects.

3.1 The Aims of a Component-based Approach

A software system is structured by subsystems or so-called components. Components themselves can be divided into modules, where each module has a specific functionality to guarantee the specific entities as described above. Components can contain a single module or can be a composition of different modules. A module has an interface and a component has an interface. We have to consider forms of interfaces so that module-to-module, module-to-component, component-to-component and/or application-to-application communication can take place. All these forms of communication are based on the approach of program-to-program communication. Program-to-program communications are different and depend upon the domain of applications. Our domain of applications is DL-applications. On the other hand we have to meet the needs of researchers to support them in the scientific communication process, where documents are "open" and "dynamic". The working draft, in which WSDL is specified, seems to be suitable for integrating the following classification of DL-applications in a so-called scientific workspace as a component-based system environment, which can be installed on a computer at the researchers working place.

3.2 The Researchers´ Activities and The Required Services

As mentioned above, a researcher is accepted in a scientific community when other researchers read and refer to his/her scientific results. A researcher in his/her scientific community is defined by his/her activities. This requires that the researcher´s publications are *searchable, discoverable, distributable, accessible* and *referable*. Furthermore, the author, the researcher, and/or the distributor of a resource has to be *authentic*. In some cases the author, the researcher and the distributor might be different persons, organisations or systems. In the scientific environment the author and the researcher is often the same person. From the viewpoint of the researcher advertising is done by personal communication, sending (structured) resources to colleagues or to conferences, or – later on – to a publisher, etc. We thus identify the following players, humans or systems: the researchers in their communities, the component-based (computer) system at the researchers working place – which we call a researcher´s workspace – and the DL-applications (available in an institution, or on the Web). Therefore we define the activities as program-to-

program communication between the researcher´s workspace and DL-applications. The following IT-system-based activities are typically done by the researcher using his/her computer:

1. *Searching* of structured resources might be done in different ways. We distinguish the search via metadata, a full-text search and the search for keywords (based on a thesaurus or an ontology).
2. When a *structured resource is discovered*, the response depends on the search form so that
   a. we can retrieve a set or sets of metadata using a DL-application, like an OPAC, or other information harvested and stored in some other (computer) system;
   b. we might retrieve a full-text, which might be the structured resource itself, including its content; or
   c. we use a stepwise refinement by traversing a given set of terms organized in the requested thesaurus or the ontology.
   Note that in 1. and 3., we do not retrieve a structured resource, which would contain the content in form of an article, music, etc.
3. The *distribution process* includes several other services like *disseminating, reachability, accessibility, and delivery and services for authentication of the structured resource and/or the user* whereas the user can be identical to the researcher.
   a. *Disseminating* can be done by sending mails, electronic newsletters, presentations on Web sites, etc. Therefore the services are mailing-lists, and (editing and advertising) systems for Web pages.
   b. *Reachability* requires that the discovered structured resource is available. The availability might be supported by an IT-system, where the resource found is at least executable online. The availability of the referred IT-system can be seen independently of the researchers workspace. The responsibility of being online lies with the provider of the structured resource. In the case of the execution of the discovered structured resource, the responsibility depends on other services such as the forms of delivery and/or ways of access, which we consider next.
   c. Studying the *delivery process*, we have to distinguish different approaches: a commercial and a more or less non-commercial concept.
   d. The delivery process can be based on clearing house mechanisms, which might be supported by the providers of the structured resources. In this case the responsibility for delivery also lies with the provider.
   e. In the scientific community many pre-prints are freely accessible, because the researchers are interested in initiating a discussion of their research topic within the specific scientific community. There are different ways of representation and storing. Some results are simply presented on a Web page located at a specific Web site, others are stored in a database, etc. If

these scientific resources are freely accessible, then the delivery process means that we need suitable representation methods for presenting the structured resources in different formats in a browser; Cocoon is a collection of transformers based on the concept that serves this need (Apache 2002).

f. The forms of *accessibility* differ as to whether we consider a commercial delivery process or the scientific approach, where the main intention is distribution of the researchers´ results by discussion.

g. In the scientific communication process the process of delivery is mainly equivalent to the process of access, whereas additional services have to be implemented. These services are components to guarantee the *authentication of the accessed structured resource* and *the authentication of the user* (resp. researcher) so that the researcher, who has requested a structured resource, can be secure that he/she receives the (original) requested resource. The authentication of a structured resource can be done e.g. by watermarking. The authentication of the user has to be guaranteed depending upon system management and the organisation of the users´ account management. There are several possible ways to access the system via a network located in an organization, an institution or a university, etc. In terms of modern applications we think about external access to wireless systems, where smart cards, and biometric identifications are discussed at the moment. For our purposes we assume that the researchers' system access and the corresponding account management are organized by the system administration of the university, where the researcher is working.

h. In the commercial delivery process access of the structured resource is controlled by the provider of the structured resources.

Some services are not necessarily recognized by the researcher, but they are needed in the DL-application environment. Such components are (sub)systems

4. for administration of metadata,
   a. harvesting of metadata,
   b. editing metadata, including the creation, the modification, deletion of metadata and their representation,
   c. distributing metadata between different metadata administration systems, (for metadata see (Lavoie, B. et al. 2002)
5. to serve thesauri and/or ontologies,
6. to provide authoring systems for editing different forms of structured resources, including the administration of annotations during scientific discussions for creating, searching, modifying or deleting such kind of annotations, and
7. to organize mailing lists, news-feeds or newsletters.

3.3 The Working Process in Simplified Technical Perspective

Obviously, the order of the researcher´s specified activities describes a working process. Some of these activities can be viewed as independent events. Others induce new events to complete the researchers aim; e.g. the search of metadata of a structured resource can be understood as a single activity, when the researcher only wants to know something about the existence of a resource, or as a composed activity which starts the process of initiating other events and finally leads to access of the resource. This implies that we distinguish between simple events and composed events so that the researcher´s activities can be represented as a process, whereas the process is described in the form of events and operations that relate the events to each other.
The program-to-program communication can be specified as operations on a specific set of events. In general, the complete working process is determined by a set of partial ordered events.
On the other hand the researcher´s working process corresponds to one or more operations, which might be implemented as a module, a component or an existing DL-application realized by a (sub)system. Further the approach to determine Web services is based on the definition of interfaces for encapsulated services whereby such services are realizations of the operations to support the researchers´ activities.

3.4 Basic Requirements for DL-oriented Web Services Related to the Researchers Activities

The application of Web services has its own specific requirements. These requirements for the specification of Web services are obviously similar to the conditions for the support of the researchers´ activities by (sub)systems as listed above, because we understand every DL-oriented Web service as an encapsulated service.
The specification of encapsulated specific DL-services requires the search of a DL-application, the availability of DL-oriented Web services, the authentication of Web services, the authentication of the researcher (respectively the user), the reachability of the DL-oriented Web service, the accessibility of the requested service, the guarantees for the application of the requested Web service (practicability), and the delivery of the requested results using a DL-oriented Web service.
Again, we might assume that the authentication of the researcher as a user of a computer is the responsibility of the organization at which the researcher is working. Additionally, we need a registry for DL-oriented Web services, where the services available are listed. A registry can be seen as yellow pages, where the available DL-oriented Web services are administered. The administration includes the definition of the Web service interface and the textual description, required by WSDL. Many other tasks are hidden. To identify these tasks of a registry we list the requirements in detail

1. the *search* of a DL-oriented service demands a search in the registry of the DL-oriented services;
2. the *reachability* of a DL-oriented Web service requires that the provider is online all the time;
3. the *accessibility* of the requested DL-application presupposes that the definition of the interface of the service is syntactically correct and semantically complete (otherwise the execution of the service can not be guaranteed).
   A requirement that has to be met by the registry and the provider serving the DL-application, is the *authentication* of DL-oriented Web services. This includes the guarantee that the requested service is the original Web service and that no interference of a third party is possible. We require that forbidden exchange of data can be recognized by the requester. The requester expects the response from such a registered service (and from nobody else). We need a secure connection during the execution of the program-to-program communication. – Other tasks are
4. the *practicability* of a DL-oriented Web service has the precondition that the requested service is executable; and
5. the *delivery* of the requested results which depends upon the requested service and the form of provision as described for commercial and non-commercial use above.

3.5 Levels of Abstraction, their Use and their Components

To determine the components of a researcher´s workspace, we have to distinguish different levels of abstraction and use. The levels of abstraction are

6. the *level of Web services,* where services – realized by accessible DL-applications – are encapsulated in the sense that the discovery, request and response of a service is independent of the concrete implementation,
7. the *application-level*, where the (re)use *of existing DL-applications* takes place via request and repsonse communication,
8. the *level of structured resources,* which contains the actual content, and
9. the *level of the external use of other applications*, which are not implemented in the workspace, like software which corresponds to the researchers´ needs, e.g. system management or network management.

As levels of use we can identify the following (sub)systems and components

1. the *network and system management*, *including the account management*, which should be supported by the researchers´ working environment itself;
2. the *middleware*, which includes the modules and components *for the support of the program-to-program communication*. The middleware is more of less hidden to the researcher;
3. the *DL-applications registered in the registry*, which are requested and might be known by the researcher (resp. the user). The DL-applications using the

encapsulation mechanism of Web services are visible by checking the registry; and
4. the *DL-applications*, which are *not necessarily registered* e.g. as mailing-lists;
5. the *programs for the representation and the transformation of results* for a presentation in a browser; and
6. the browser itself used as an *user interface*.

4 A Framework for DL-oriented Web Services

We are building a framework for DL-oriented Web services based on existing DL-applications. This framework contains at least two aspects: the *mappings of workflows* in a scientific environment supported by Web technologies, whereas the use of such technologies initiates new social processes (van der Aalst et al. 2002), and the *supply of the web-oriented infrastructure* in form of a component-based system for the implementation of a scientific workspace in a distributed environment. From the viewpoint of the Web service technology specified in (W3C 2003a, W3C 2003b), the development of DL-oriented Web services represents a specialization. This specialization is based on existing communication processes and workflows in a scientific community. As mentioned above, this requires a generalization of DL-applications, especially are policies needed, including conventions, which determine the standards for interoperable interfaces. The realization of such conventions has also to be initiated as a social process. Taking in account the social aspects – as described above – we (re)use existing Web technologies for the integration of existing DL-applications.

4.1 Models of Web Service Architectures Applied to DL-oriented Web Services

The model of Web services follows the paradigm of service-oriented architectures (SOA) (Baresi, L. et al. 2003, SOA 2004). Web services act as an intermediary between servers and clients, whereas the role of a server and a DL-oriented client-application can alternate. The task of a DL-oriented Web service is to supply a bidirectional application-based connection between the client-application and the server and vice versa. We assume that the client requests a DL-oriented Web service. It acts as a caller, and the Web service has to allocate and handle the communication process to the provider of a requested service. Then the characteristics of a communication process for DL-oriented Web services can be described as follows. Note that this specification of the request-response structure of DL-oriented Web services is more detailed than the process initiated by a Web service, which is specified in (W3C 2003b)

1. The communication process must be handled by the middleware to supply DL-oriented Web services;

2. the middleware has to accept a request of a client, which is initiated by a component in the component-based system of the scientific workspace;
3. the middleware has to locate and address the requested DL-application;
4. the request might be transformed into a protocol structure - e.g. in form of an encapsulated message using SOAP - which is distributable and is accessible by the requested DL-application;
5. the request must be send to the server, where the DL-application is located;
6. the request is executed on the requested server of the DL-application;
   a. therefore the encapsulated message has to be parsed on the server, especially if parameters are needed for the execution of the requested DL-application;
   b. the server fulfills the requested service by executing the DL-application;
   c. the potential result set has to be encapsulated such that
7. the server is able to send a response to the client, e.g. again in form a SOAP message;
8. the client has to parse the received message to obtain the content of the response.

Note that while processing 1.-8., at least a session handling – or better a transaction handling – between client and server might be needed in a distributed environment. From our viewpoint it is an open question, whether it is possible to implement an adaquate transaction handling for the considered request-reponse structure in a distributed environment, where the requested service is unknown at the beginning of the transaction.

4.2 Technologies for the Implementation of DL-oriented Web Services

At the present state the implementation of DL-oriented Web services does not yet fulfill the conditions of the framework specified above. We have started with a simple prototype implementation, because the social process to obtain a policy for DL-oriented Web services has still to be initiated.

As we have shown, to implement middleware in form of DL-oriented Web services, we need to distinguish between the definition of a service interface and the functionality of a service. The actual implementation of the functionality of each DL application is hidden by the caller (resp. user). A UDDI-registry is used to register applicable DL-oriented Web services. The UDDI-registry includes the definition of the interface of a DL-oriented Web service, and a description of the functionality of the DL-application, which might be requested via the interface. We use a specialized (sub)set of the Web Service Description Language to define the specification of DL-oriented Web services. The discussion, what kind of elements have to be included in a DL-oriented Web service language, is in process.

The UDDI-registry is also responsible for the localization of the requested DL-application. DL-applications are located via URI´s, which are mapped to specific

namespaces. The namespaces are included in a set of DL-oriented metadata, which are administrated in a XML-database in the same manner as in SozioNet (Meier et al. 2003). The namespaces are resolved using the query language of the XML-database such that the information on the interface of the DL-application can be retrieved. At the present state the parameters given by the definition of the interface of the requested service have to be serialized using XML:RPC and have to be encapsulated in form of SOAP messages for the distribution. This means the values of the parameters of the interface have to be mapped in a specific way. Only simple data structures can be mapped automatically. Other mappings have to be done manually, especially when higher structured data types are needed. On the other hand the middleware for the supply of DL-oriented Web services is able to evaluate SOAP messages including XML:RPC to transfer and transform result (sets) of the requested DL-applications to a browser, which is the user interface.

5 Conclusion

DL-applications support the scientific publication process. The researcher´s publications have to be searchable, discoverable, distributable, accessible, and referable. We derived criteria to describe the researchers´ activities so that we can relate the working process to existing applications for the handling of structured resources. The corresponding needs are different forms of search like external and internal search of structured resources, the search of DL-oriented Web services, facilities of discovery of structured resources and DL-oriented Web services, reachability of structured resources and DL-oriented Web services, authentication of researchers, structured resources and DL-oriented Web services, delivery of structured resources, delivery of result sets of DL-oriented Web services and accessibility of metadata, structured resources and DL-oriented Web services. Note that we have to distinguish in all cases between the objects and their levels. The objects (or sets) are structured resources, metadata, and Web services. The initiation of events defines the executions of operations; the operations are searching, harvesting, discovering, accessing and delivering, whereas the operations have to be determined depending upon the sets of objects. There is a process model given by the SOA paradigm for the design of Web services. We presented a specialized process model for DL-oriented Web services, which can be easily supported by the use of existing Web technologies.

6 References

van der Aalst, W., van Hee, K. (2002). *Workflow Management. Models, Methods, and Systems*.The MIT Press, Cambridge, Massachusetts, 2002.

Apache (2002): *Cocoon 2.0*. http://cocoon.apache.org/2.0/
Baresi, L.; Heckel, R.; Thöne, S.; Varró, D. (2003). *Modeling and Validation of Software-Orientes Architectures: Application Vs. Style*. In: Proc. Of the European Software Engineering Conference and ACM SIGSOFT Symposium on Foundations of Software Engineering (ESE/FSE 2003), Helsinki, Finland, ACM Press.
Lavoie, B.; Dale R.; Alexander, M.; Bradley, K.; May, M.; Guenther, R.; Hurley, B.; Lupovici, C.; Rieger, O.; Sergant, D.; van der Werf, T.; Webb, C.; Wendler, R.; Woodyard, D. (2002). *Perservation Metadata and the OAIS Information Model. A Metadata Framework to Support the Perservation of Digital Objects*. A Report by The OCLC/RLG Working Group on Perservation Metadata, RLG Mountain View CA. http://www.oclc.org/resaerch/pmwg.
Meier, W.; Schumann, N.; Heise, S.; Schmiede, R. (2003). *SozioNet: Networking Social Science Resources*. In: Proc.Research and Advanced Technologies for Digital Libraries, 7th European Conference, ECDL 2003, Trondheim, Norway, August 2003.
Raymond, E. (1999). *The Cathedral and the Bazaar – Musings on Linux and Open Source by*
Accidental Revolution, O'Reilly, Beijing, Cologne.
Service Oriented Architectures. An Introduction. See: http://www.developer.com/design/article.php/1010451 and http://www.developer.com/services/article.php/1014371 [28 February 2004]
Stoll, J.; Sonkajärvi, H.; Körnig, S.; Schmiede, R. (2003). *How Should Libraries Respond to New Forms of Publication? Some Reflections on the Expectations of Users and Providers and What Could Be Realized at the Present State of Technical Development.* Digital Resources in the Humanities (DRH 2003), University of Glouchestershire, Chaltenham. Great Britain 26-29 September 2003.
W3C – XML (2000): *Extensible Markup Language (XML)* 1.0 (Second Edition) [06 October 2000] http://www.w3.org/TR/REC-xml
W3C (2003a): *Web Service Description Language (WSDL) Version 1.2;* Part 1: Core Language [11 June 2003] http://www.w3.org/TR/2003/WD-wsdl12-20030611; Part 2: Message Pattern [11 June 2003] http://www.w3.org/TR/2003/WD-wsdl12-patterns-20030611*;* Part 3: Bindings [11 June 2003] http://www.w3.org/TR/2003/WD-wsdl12-bindings-20030611
W3C (2003b): *Web Content Accessibility Guidelines 2.0* [24 June 2003] http://www.w3.org/TR/WD-WCAG-20-20030624

7 Contact Information

Julia Stoll, Stephan Körnig, Rudi Schmiede
Darmstadt University of Technology
Residenzschloß

64289 Darmstadt
Germany

e-mail: jstoll@ifs.tu-darmstadt.de
e-mail: koernig@ifs.tu-darmstadt.de
e-mail: schmiede@ifs.tu-darmstadt.de