Advanced Concepts for Next Generation Portals

Christian Zirpins, Harald Weinreich, Andreas Bartelt, Winfried Lamersdorf
University of Hamburg, Department of Informatics, Distributed Systems Group (VSYS)
{zirpins,weinreich,bartelt,lamersd}@informatik.uni-hamburg.de

Abstract

Over the last two years, Portal sites surfaced as a major trend in electronic commerce. Unlike a lot of other exaggerated trends, Portals have become an important facet of the World Wide Web which is likely to stay. Emerging from search engines and catalogues, they integrate an increasing amount of information and functionality. The paper reflects on Portal characteristics as well as on foundations of their realization and introduces technology needed for the next generation service trading concepts: based on the promising open source project Jetspeed a prototype of the GIBRALTAR service Portal is introduced, capable of integrating, trading and composing complex remote e-services.

1. Introduction

Over the last few years, the Internet, and most of all the World Wide Web, which is constantly expanding in qualitative terms of technology and quantitative terms of size, have become an essential medium of business and society. Simultaneously, an increased need for orientation within a growing universe of multi-dimensional possibilities related to form and function emerged. From a technical viewpoint, this typically leads to integration and abstraction efforts, which are today realized through a fuzzy class of mediating information systems called Portals. This notion of Portals spans a wide range of solutions, quickly evolving in pace with internet speed and ranging from simple resource catalogues to highly integrated brokerage facilities for open electronic services.

The paper is organized as follows: section two characterizes the notion of Web Portal sites, including a classification scheme and an overview of the basic technical building blocks. Section three discusses electronic services in Web-based application context and proposes Portal technology for their desirable integration. Section four introduces concepts for service integration in future Portals on a methodological level, which is done by example of the GIBRALTAR project. Furthermore, it discusses experiences gained from the implementation of a Web-based prototype system. The paper closes with a summary and outlook.

2. Characteristics of Portal sites

The term *Portal site* or *Web Portal* has not been clearly defined yet. As one of the buzzwords of recent years it was used and abused for a wide range of Web-site types. However, some common characteristics can be identified; other aspects like the intended users and the offered services help to define different Portal categories. The next section describes the most common Portal types, followed by a look at basic technologies employed.

2.1. A Portal classification

Portals range from Web catalogues to complex intranet solutions. However, a common characteristic is their role as a starting point, offering a doorway into using Web services. With regard to the aims of the users and the information respectively services offered, a classification of Portal sites can be accomplished [1]. Most Portals today aim at consumers. Here two basic classes can be identified:

Horizontal Portals or *consumer Portals* are Web sites that serve as a universal entry point into the Internet [2]. They usually offer a broad array of resources and services in an effort to convince users to make the site their home page and let them use it as long and often as possible. The typical features of consumer Portals include searching capabilities, Web catalogues, messaging services, news, online shopping and free home pages. The first consumer Portals were hosted by online services, such as AOL. .Meanwhile, most of the traditional search engines have transformed into consumer Portals to stay in competition for customer favor.

Vertical Portals, sometimes called "*Vortals*", constitute the second basic class of Portals offering content and services aimed at a specific domain or community. Vortals can be focused on consumers with particular tasks, people at certain locations or communities with individual interests.

One notable kind of Vortals are "*Enterprise Information Portals*" (EIPs), also called *"Corporate Portals"*. These are Web applications integrating all kind of data and services related to a specific company. In this way, they represent a common entry point for customers and employees [3]. EIPs offer e.g. support information, detailed product catalogues and the functionality of an

online-shop. Two varieties of these concepts are "*Business Portals"* and "*Knowledge Portals"* [4]. Former provide a single gateway to personalized commerce information on a group of companies, allowing one-stop shopping like in a shopping mall. Latter are more focused on business data and information exchange, which are needed to make well-informed business decisions.

Another subclass of Vortals is constituted by *Intranet Portals*. These are business applications dedicated to the personnel of a company, serving all business units and every departmental requirement [5]. Thus they allow a consistent view on the business, which can, however, be personalized to the employee's needs. Intranet Portals give access to enterprise data, provide frequent business applications like attendance recording or settlement of travelling expenses and support cooperation as well as workflows.

Recently a new kind of Portals emerged, aiming at corporations rather than consumers: *B2B* (business to business) or *Industry Portals* [6]. They provide a virtual marketplace where particular industries can benefit from information sharing, and most importantly, the automatic establishing as well as processing of business transactions. Thus, they can act as basis for new economies and be a catalyst in lifting old economies to contemporary competitors. The GIBRALTAR project described in this paper is placed in this field.

2.2. Basic technologies of Portals

In April 1994 the Ph.D. students Jerry Yang and David Filo of the Stanford University started to experiment with a searchable Web directory, categorized by topic. This project developed quickly, found backers and is now known as "Yahoo!" [7]. Starting from such simple link collections Portals have developed to highly complex application engines. Today, most Portal systems either run on high-performance application servers or on servers that support a robust repository combined with high-speed output and delivery system. They integrate different kinds of Portal functionality and support interfaces to various back-end information stores and application engines.

One basic concept of Portal sites is the *portlet*, introduced by Oracle in September 1999 [8]. Portlets are applications within the Portal much the same way as servlets are applications within a Web server. Portlets are used to simultaneously offer different services on one page. From the user's point-of-view, a portlet is a specialized content area that occupies a small window on the Portal page. Portlets can be modified independently, and the user can move them on the Portal screen or maximize and minimize them.

This leads to a key feature of current Portals: the ability to personalize and customize the functionality and appearance of the Portal to personal needs. The user cannot only select the graphical appearance and contents of the Portal home page, he is also often empowered to tailor the portlets themselves, according to his preferences. These settings are stored and subsequently make the Portal "his" personal home page. Some Portal systems like Intranet Portals also allow the administrator to define groups as well as set customized client permissions according to user tasks, location or status.

Many different vendors of commercial solutions already included Portal frameworks within their product range, usually integrating this functionality within their application servers. Two examples are constituted by the BEA WebLogic Commerce Server 2.0, or Oracle's 9iAS server including the Oracle Portal Architecture. As we were looking for a less expensive alternative, we found the open source project *Jetspeed* to be a promising candidate. Jetspeed is a part of the well-known Apache Project [9] and offers a comprehensive framework for building Web-based information Portals.

3. Portal based service brokerage

Simultaneously to the rising of Portals a second major development took place throughout the Internet with the emerging of open electronic services (sometimes called "*e-services"*). Along with numerous promising advantages and capabilities goes a series of severe problems. Integrating the concepts of Portals and e-services could solve some of them.

In the following chapter e-services are examined and put into relation with Portal concepts. First, basic concepts of e-services are introduced. After that, an overview of fundamental system-support mechanisms is given, which are capable of handling complex e-service scenarios and intended to be integrated with Portals.

3.1. Electronic services and markets

Within the evolution of the World Wide Web, one of the major changes has been the shifting of content from passive hypertext documents to active distributed services. Based on globally interconnected systems and encouraged by massive economical investments of the "new economy", a multitude of distributed e-services – focused on end user clients as well as reselling providers of value added services – emerged over the last years, lengthily exceeding a critical mass [10].

The consequentially emerging electronic service markets are forming an environment of complex business processes, which often go beyond the scope of single services thus fostering and often naturally forcing their decomposition into multiple sub-services. For these value added services, corresponding value chains have to be established to facilitate concrete transactions [11].

Though the significant quantity of possible services respectively service combinations leads to markets with a rich variety of offers as well as fair competition, it also results in severe problems for customers. They have to spend a lot of time and effort to gather the information, needed for orientation in the great number of providers and their offers, finding the best business deal. Providers, on the other hand, while benefiting from easy market access, face major difficulties in the attraction of customers and business partners, further aggravated by increasing specialization of services.

Today, Portals, immanently acting as integrators respectively concentrators, appear to be an excellent solution for the outlined conciliation problems of global electronic service markets, as their general concept comprises the provision of a central point of access to aggregated information and service collections. However, to accomplish this, they have to shift from passive catalogues to active brokers, joining together a dynamic, ever changing market of distributed electronic services.

Augmenting today's common understanding of Portal concepts – which, as illustrated above, comprise mostly link collections and a few locally provided support functions – with consequent encouragement of electronic markets, requires advanced methodological support at application level, which equips the user with active brokerage abilities for e-services. Related to this, service requestors should be guided from an orientation phase consisting of the localization of suitable service categories and their combinations all the way to an application phase comprising optimized trading of corresponding service providers, readily available, and their integration into coordinated value chains. Realizing such advanced capabilities in a Portal requires an enabling technological environment, providing system level support for handling e-services.

3.2 An infrastructure for service support

In order to accomplish a system level infrastructure for e-service support, initially a generic typing concept for service categories is needed.

First, the typing of services facilitates their classification, which is needed for abstract service catalogues. These catalogues provide means of orientation within the service offer space, crossing the domain boundaries of single providers.

Subsequently, service typing supplies a foundation, which makes it possible to establish optimized relations between requestors and providers. This is done by using the automated matching abilities of trader mechanisms. Additionally, type information allows the combination and integration of mutually dependent partial-services to coordinated value chains.

In order for Portals to benefit from service typing in this way, the type-system as well as the related system-support-mechanisms have to fulfill some basic requirements.

First of all the type-system has to simultaneously accomplish an abstract customer-related view intended for classification and cataloguing of service-offers as well as an extensive technological view to facilitate the trading and coupling of corresponding code-level service-instances. Related to the type meta-model, this implies the obligation to offer sufficient means for a coexisting synchronized consideration of semantical as well as syntactical aspects.

Additionally it should be possible to specify expandable relations between *service types* and as far as achievable to automatically deduce them. Regarding the desired brokerage-functionality of Portals, conformance and compatibility relations are most important. The conformance-relation makes it possible to consider alternatives for both the selection of service-categories as well as the mediation of corresponding service-providers. The compatibility-relation of types on the other hand assures a sound coupling of their corresponding instances, allowing for a consequently type-save composition of partial-services.

These requirements are only partly addressed by today's common solutions. As an example, the Universal Description, Discovery and Integration (UDDI) approach [12] is, due to its pragmatically reduced service model, not capable of expressing the amount of meta information, needed to deduce sound type relations, usable for programmatically service handling.

In order to soundly handle services in a Portal-site, dedicated system-level support-mechanisms [13] are needed, which first implement the typing-concept and subsequently utilize its expressiveness to realize the mediation of services.

According to this, first of all a *typemanager facility* is needed for a flexible and extendable handling of *service types*. Its functions include registration of new service specifications, *service type* searches respectively browsing and deduction of *service type* relations utilized to accomplish type checking as well as comparison.

Subsequently, a *service trader* provides for the localization of optimum service-suppliers readily available. This functionality comprises the lookup of separate as well as groups of services based on their types. In order to accomplish these tasks, the service trader takes advantage of the typemanager.

Finally the service instances – corresponding to *service types* and provided by service-suppliers – have to be coupled and integrated into the Portal-site. This is the task of a Portal-hosted *integration environment,* which holds local surrogates, remotely connected to corresponding service instances. Moreover, this runtime environment has

to realize a means of coordination for composed partial-services, again taking advantage of the typemanager.

The sketched out concepts are being actively utilized at the University of Hamburg, where a service-mediating Portal in the context of electronic information markets gets realized. The next chapter introduces this research project and gives details of the prototype implementation.

4 The GIBRALTAR Portal

To make the illustrated concepts more concrete, they will be explained by using a Portal which is part of the research project *GIBRALTAR* [14].

Within this project, which belongs to the larger *Global Info* initiative, a comprehensive user-centric integration platform for electronic publication services is being created, leading the way to information Portals of the next generation.

The principal Global Info initiative [15], which is the German counterpart to international digital library projects, aims at creating a general environment of electronic information. To accomplish this, a lot of different application-level services including federated catalogues, retrieval-engines, alerting-facilities or document-stores have to be homogeneously integrated.

According to this, the GIBRALTAR project provides fundamental infrastructure mechanisms comprising facilities for typing, trading, negotiation and integration. Together these system-support-services yield brokerage functionality, which is accessible for all users via the GIBRALTAR Portal site.

To illustrate this Portal, first its utilization is going to be outlined. After that, a prototype system will be shown, giving some details about lessons learned from the implementation.

4.1. The GIBRALTAR Portal application

The GIBRALTAR Portal aims at the support of users in the domain of electronic publishing (e-publishing), which can be divided into different roles including authors, editors, publishers, distributors librarians and readers. The users can be separated furthermore into the orthogonal categories of customers and providers. Customers are using the Portal as a starting point to achieve their domain-specific goals. They are being supported by the Portal with orientation, mediation and application functions. Providers on the other hand use the Portal to present their services exactly to the desired audience. In order for their services to be tradable by the Portal, they are supported in specifying their *service types* and offers. Subsequently, functions will be addressed, which are focused on the role of service requesting customers.

The GIBRALTAR Portal guides its customers to help them achieving their goals. This process is divided into three steps, which assemble a generic procedure to accomplish different objectives in the application domain of e-publishing.

In the first *orientation and mediation* phase, a mediation of task-related services based on individual characteristics and preferences of customers is done, guiding users of different skills to a set of providers, that together accomplish the desired goal. Here, customers find out about the necessary processes, the related services and finally the most favorable providers ready available.

The second *integration* phase of the Portal guides the user in combining sub-services as to establish value chains. These combinations are deduced from the objectives, preferences as well as formerly chosen *service types* and providers, leading to optional or necessary additional services.

Furthermore, coupling and coordination of partial services are supported on system level. This includes establishing of service interoperability and control of process flow, which are deductible from type information. To allow for individual customer influence on task sequences, workflow capabilities are intended to be added in the future.

If concrete services are being taken into account, this conventionally leads to proprietary procedures of use, including repeatedly registrations and different user interfaces. Where registering over and again is just inconvenient, different user interfaces can lead to severe failure due to semantical incompatibility.

To cope with this, the GIBRALTAR Portal offers means of support for *service access and utilization*. This includes initialization, calling, and interactive control of providers in a homogeneous fashion. An important part of this is semi automatic generation of generic Web interfaces (portlets) for categories of services. For further convenience, the phase can be backed by persistent customer profiles, enabling automatic service configuration and single-sign-on capabilities.

All three phases, outlined above, are intended to be self-contained parts, which – while being closely related to each other – can be combined in an individual fashion, related to role and skill of users.

Those individual user characteristics generally have an effect on support functions of a Portal. They can be utilized to optimize the way assistance is carried out. Thus, temporary as well as persistent personalization is a desired feature of the GIBRALTAR Portal, backed by profile gathering and accounting. Besides its influence on the essential phases of assistance, personalization allows for some nice-to-have tools like bookmark lists or individual desktops.

4.2. Prototype implementation

In order to put the GIBRALTAR Portal application into practice, a prototype system was realized, for which an architectural outline will be given now. This overview is followed by some notable details of the implementation.

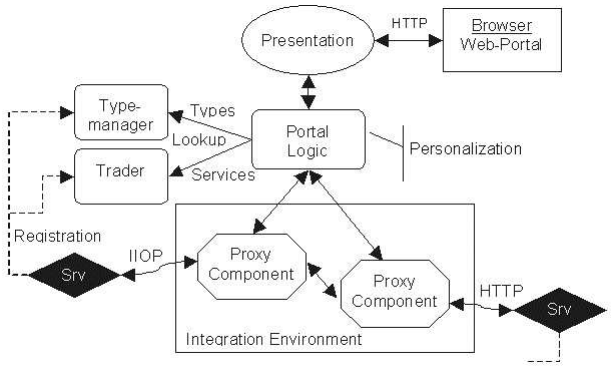

Figure 1. GIBRALTAR broker Portal architecture

Figure 1 illustrates the system architecture of the GIBRALTAR prototype implementation. Fundamental parts of the architecture are given with the *typemanager* and *trader*. Together they realize the system-level functions needed for service mediation. The typemanager implements a sophisticated component-based type system [16] and allows for administration as well as operations on generic service-specifications. The trader, which is based on the typemanager, dynamically maintains a repository of current offers and is capable of matching requests for services in an optimizing fashion.

Services, realized by remote providers are represented in an *integration-environment* by component-oriented service-proxies [17]. These are based on an extendable set of adapters for different communication protocols like IIOP, SOAP or HTTP. The runtime environment is also intended to hold several facilities of coordination and negotiation not in the focus of this document.

System-level components are pulled together by the *Portal Logic*. This central device holds the functionality of the GIBRALTAR Portal application. To provide this functionality, it uses the mechanisms of the surrounding components, i.e., typemanager, trader and integration-environment for orientation, mediation, combination and utilization. Additional functions like some aspects of personalization are intended to be outsourced to external service providers. Finally the Portal has the task to render the application logic into a presentation, forming a Web interface.

The implementation of the Portal prototype made use of the *Jetspeed* framework, being developed as a part of the Apache Jakarta project. Jetspeed has significant advantages like its open source status, a well-designed architecture, a high flexibility and a significant performance provided by an advanced caching logic.

However, during the implementation of basic concepts of GIBRALTAR we could localize some additional demands, which have to be considered for the development of next generation Portals. These demands were neither covered by Jetspeed nor by the Portal packets of commercial application servers.

One main concern is the limited autonomy of portlets. Their concept seems to be too closely tied to their functionality as a part of a Web page. As every portlet represents a different service, it would be more beneficial to make them more independent units, reaching down to system level. A resulting problem was caused by conflicting parameter names. When two portlets used equal names, e.g. for HTML form parameters, these variables could not be distinguished. To solve this problem we introduced *portlet name spaces*: Every portlet automatically adds and utilizes an own unique name space, so even parameters of two instantiations of the same class situated on one page can be discriminated.

Furthermore the GIBRALTAR Portal prototype required the ability of portlets – representing services – to communicate with each other. This can make the output of two portlets on one Portal page dependent, e.g. they operate on the same data or require each other's computations. As Jetspeed generates the output of portlets on one page in a strictly sequential manner, this could lead to erroneous outputs if an early portlet in the sequence depends on data provided by a later portlet. A consistent and correct output requires the computations of all involved portlets to be completed, before their output is generated. We achieved this by introducing *two-phase portlets*, realizing the proven model-view-controller paradigm. In the first phase, parameters are processed and computations are done, while the second phase generates the output of the portlets.

The most sophisticated problem to be solved was caused by time constraints. On the one hand the output of Portal pages has to be as fast as possible, on the other hand some enhanced GIBRALTAR portlets require the time consuming results of complex computations or the unpredictable prolonged replies from services located on remote hosts. As one of the main usability aspects of all Web applications is the instant reply to a user's request a new concept was needed. We developed the idea of live-long partner threads, which we called *worklets*. A worklet is associated to a portlet and can be utilized for any extended operation. The associated portlets display the status of such lengthy computations by using a status bar applet. These applets stay in contact with the worklet by a socket connection. After finishing the computation, the applet requests a page-reload and the result of the worklet is displayed. Hereafter, the worklet can either be disposed or kept for further tasks.

5. Conclusion

On the background of contemporary Internet evolution, which led to the progression from a source of passive textual information to a complex market of active e-services, this paper looked at the current nature of Web Portals, characterizing some of their basic categories and technologies.

While the currently emerging open service markets provide some promising possibilities, they also cause numerous problems: orientation in the growing number of offers becomes increasingly demanding, and so customers have to spend a lot of time and effort to find a well-suited service. In other cases, specialized services of different providers have to be combined to get a desired result or make a reasonable transaction. However, the heterogeneous character of services including their mostly proprietary and therefore inhomogeneous interfaces often complicate these efforts or even make them not worthwhile.

Within the presented GIBRALTAR research project, solution attempts have been proposed, which aim to reduce negative consequences of the increasing amount and diversity of services offered on the Internet. The integration features, which have been presented, incorporate different novel technologies to create a domain specific vertical Portal, allowing an easy access to the service market of electronic publishing.

The presented prototype implementation proved the feasibility of the developed concepts, while showing some of the problems of today's Portal frameworks. These results provide a promising basis for further developments.

6. Acknowledgements

We would like to thank the German Federal Ministry of Education and Research (BMBF) for funding the GIBRALTAR project as part of the Global Info initiative.

Furthermore we want to express our gratitude to Sven Offermann, Stefan Reich, Jens Hahn and Stefan Westerfeld for helping us to develop and implement these concepts.

7. References

[1] M. Schumacher, A. Schwickert: Web-Portale Stand und Entwicklungstendenzen, in: Arbeitspapiere WI, Nr. 4/99, Lehrstuhl f. Allg. BWL und Wirtschaftsinformatik, Johannes Gutenberg-Universität (Mainz), 1999

[2] J. Neil, B. Bass, C. O'Connor, J. Aldort, T. Grimditch: The New Business Portal, Forrester Report February 1999, Forrester Research (Cambridge), 1999

[3] C.C. Shilakes and J. Tylman: Enterprise Information Portals, Merrill Lynch 1998, http://www.sagemaker.com/company/WhitePapers/eip_indepth.pdf

[4] J. Firestone: Defining the Enterprise Information Portal, White Paper, Executive Information Systems, 1999, http://www.dkms.com/EIPDEF.html

[5] J. Walker, T. Schadler, A. Ciadelli, C. Overby: Building an Intranet Portal, Forrester Report Jan. 1999, Forrester Research Inc. (Cambridge), 1999

[6] D. Schneider, G. Schnetkamp: E-Markets B2B-Strategien im Electronic Commerce: Marktplätze, FachPortale, Plattformen. Gabler Verlag, Sept. 2000

[7] Yahoo! Company History, Yahoo Inc., 1999, http://docs.yahoo.com/info/misc/history.html

[8] T. Uimonen: Oracle sees portlets behind corporate Portals, NetworkWorldFusion News, 23.10.1999, http://www.nwfusion.com/news/1999/0923portlet.html

[9] The Jetspeed Portal Framework, A Jakarta Apache Project, Feb. 2001, http://jakarta.apache.org/jetspeed/

[10] M. Merz. Elektronische Dienstemärkte. Springer, 1998

[11] A. Marton, G. Piccinelli and C. Turfin: Service Provision and Composition in Virtual Business Communities, in: Proc. SRDS: 18th Symposium on Reliable Distributed Systems, IEEE Computer Society Press, 1999

[12] Universal Description, Discovery and Integration (UDDI) Initiative , http://www.uddi.org, Mai 2001

[13] K. Geihs, W. Lamersdorf et al. Systemunterstützung für offene verteilte Dienstemärkte, in: Proc. of KIVS'95. Springer Verlag, 1995

[14] VSYS Group, Computer Science at University of Hamburg: The GIBRALTAR Project, 2001, http://vsys-www.informatik.uni-hamburg.de/projects/globalinfo.

[15] Global Info Initiative: Global Info – The German Digital Library Project, http://www.global-info.org

[16] F. Griffel, C. Zirpins and S. Müller-Wilken: Generative Softwarekonstruktion auf Basis typisierter Komponenten, in: Proc. of KiVS'01, Springer-Verlag (Berlin), 2001

[17] F. Griffel: Componentware. dPunkt Verlag, 1998