Dear Reader,

When we were asked to write the editorial of the very first issue of the International Journal of Web Portals, we were more than happy to accept. This is a perfect timing for this new journal, as the Portal market has seen many years of quick progress since its inception in the late 20th century. With emerging technology trends in social computing, web 2.0, mashups and recent developments in Portal related standards, the Portal marketplace continues to be very vibrant. We are confident you will find many valuable insights by reading this journal.

Technology in the Internet and web communities has turned from an ecosystem surrounding a sizeable set of key players defining and leading technology change, to an evolutionary ecosystem consisting of large and small organizations that influence each other (Tim O’Reilly, 2005).

Let us take a moment and have a look at the environment surrounding us and the path that brought us here. In the late 90’s the term Web-Portal (Wikipedia, 2007) was created to describe web sites that were providing entry paths into the Internet for the increasingly proliferated browser usage. Companies like Netscape, Yahoo, AOL, Altavista and many more provided start pages for their user communities leading and guiding them through the Internet. They were either structured web directories, or unstructured full text search engines. This commonly available information was combined with personal services like email, news, chat and file storage services, providing users with their very own, personalized web pages. Beyond this initial focus, there is value that Portals can provide for any business through personalized and centralized access to information and services. Companies started to adopt the Portal concept and related technology for their business-to-employees (B2E Portals), business-to-business (B2B Portals) and business-to-customers (B2C Portals). However, Enterprise requirements for Portal governance were not addressed by public Web Portals that only focused on increasing user and request numbers. Enterprise and business user needs were only addressed by Portals created for enterprise use. Portals of various sizes
were built by these companies, some were used by small user communities sharing a limited set of applications, some were used as the central and only entry path into mission critical applications. Portal technology became an integral part of the enterprise middleware stack. Portal technology is applied to information driven, transactional and process driven sites (Gartner, 2007; IDC, 2007; Patricia Seybold Group, 2007).

Many articles were written about the amount of functionality modern text processing software has to offer, and the percentage of which is actually being used by the average users. Today leading edge Portal software evolved likewise. It is extremely powerful, providing a large variety of out of the box configuration options. For the enterprise space, there are many specific and sophisticated capabilities implemented. It can also be observed, however, that looking from a user perspective, many only use rudimentary customization. This is especially true for internet-based Portals. “Why that?”, one may ask, and the answer is not easy. It might be in the area of usability, meaning it’s still to cumbersome to personalize the experience, or – especially true for non-technical users - they might not even notice the advantages and capabilities a modern Portal offers. Or, as today’s users have a multitude of accounts spread across the internet, it may simply be that they do not use the Portal often enough to justify a need to customize it to their needs. Finding the correct answer here is an interesting area of further investigation and research.

Future approaches may take steps in further standardization of web based user interfaces, or it may be in providing increased ease-of-use, for example by adding more context-sensitivity, generating self-adopting Portals (A. Nauerz, G. Groh, 2008) which show complexity only on demand. After years of debate on the vision of “semantic webs” (Tim Berners-Lee et al, 2001) and initial efforts to establish knowledge bases and semantic databases, the term is seeing a new focus. Technologies like semantic tags, microformats (Microformats, 2007), natural text processing frameworks (OASIS UIMA TC, 2008) and semantic annotation services (Clearforest, 2008) start to enable user interaction services and components to provide semantic inspired value on top of human readable web pages (Intel, 2008). Will user interaction components and frameworks like Portlets and Portal pick up this trend and start to enrich user interactions with semantic services? Context and state data available in a Portal seems to provide a valuable base for building out middleware semantic services for user interactions.

What is the common base that these new services are building on today? In 2003, JSR168 (JSR168, 2008) defined the first version of a common Java API for implementing Portal components. This first version of the standard covers lifecycle, configuration and user interaction models for Portlets within a Java based execution environment, through a Portlet Container. JSR 168 provides Java components with standardized access to information (e.g. language,
markup, encoding) that is required for the construction of consistent web page fragments. The second and related standard Web Services for Remote Portlets 1.0 (OASIS WSRP TC, 2008) is defined by the OASIS (OASIS, 2008) consortium. WSRP defines a web service interface for integrating user interface fragments into web pages which are provided by remote servers. JSR 168 and WSRP 1.0 follow the same conceptual model of assembling independent user interaction components. Both standards and their underlying component model were embraced within the industry, with most commercial and non-commercial Portal implementations providing support for these open standards.

At the time of writing this article, new versions of the two fundamental Portlet standards are completed and agreed to within the community (Portlet Specification 2.0 (JSR286, 2008), WSRP 2.0 (OASIS WSRP TC, 2008). These new Standards extend the capabilities of Portlets to support context sharing and context exchange mechanisms. JSR 286 supports render context sharing by the concept of public render parameters and introduces an event model for Portlets that are executing within the same Portlet container environment. WSRP2.0 provides new service endpoints for event delivery and render context parameters for service invocations.

Within the Web2.0 technology space, lightweight REST based UI fragment models are used. Fragment models referred to as gadgets (Google, 2008) or widgets (Netvibes, 2008; IBM, 2008, Lotus Mashups) are currently vendor proprietary models. Some technology provider alignment initiatives are also building, for example Open AJAX (Open AJAX, 2008). Fundamental to these different fragment models, beyond their support for composition of multiple fragments into a single page, seems to be support for user or page specific configurations and the expressed need for context exchange mechanisms. The base concepts of these fragment models appear quite similar to the focus areas of the Portlet component model. It will be interesting to follow the evolution of these technologies and see how they will influence each other. While their capabilities appear very similar, there is a difference in the adoption approach of these different technologies. It seems Portlets are predominantly leveraged by IT personnel of service and data owners to provide user interaction access to their own systems. On the other hand, fragment models find more and more adoption within technical communities that are predominantly connecting to existing data and services (ProgrammableWeb.com, 2008).

In these new Mashup models, it’s the end users who create applications that are providing solutions to the very specific needs of themselves or their workgroup very quickly, by combining existing services. This ease of use for simple, situational application creation causes a shift of control towards the user. The amount of its adoption, especially in the corporate businesses, and subsequently the way this integrates into the existing centrally-controlled IT systems is one of
the most interesting spaces of evolution in the coming years: How does this new ecosystem influence the way people work with IT systems within enterprises, what will be the success model for the fusion of enterprise IT needs for governance, security, reliability with community driven technology?

Looking at the infrastructure side when trying to predict Portal evolution, some interesting trends will further expand Portal scalability and reliability: The widespread adoption of 64 bit system will finally break the 4GB memory barrier and taking away one potential issue for enterprise scale systems. The trend towards more and more parallelism in processors and systems (clustering) is already exploited by Portal software to a large extent today, but the possibilities of Virtualization will further add new use cases for Portal adoption, allowing e.g. automated and dynamic addition and removal of system based on the load – a feature heavily needed for flexible internet based Portals. So, recent advances in hard- and software technology provide a solid base for a further scale-up of Portals in various dimensions – users, size, speed and reliability.

As this adds yet more complexity to a Portal system, at the same time significant advances in the autonomy of these systems by introducing more and more self-healing technologies will be necessary, in order to be able to manage these modern systems. It will be very interesting to watch whether this increasing autonomy will enable Portal operation and administration to be done in-house also in non-IT centric companies which today mainly operate with the help of service providers, or whether this same complexity will accelerate the trend towards Software-as-a-Service solutions also in the Portal space. The first examples are already in existence, building on the concept of virtual Portals (which allows one installation to provide multiple, isolated Portal sites), and are mostly providing a set of predefined applications augmented with a Portal interface to provide a customizable, configurable interface to a variety of customer companies. Here, another link into the previously mentioned “Mashups” trend can be clearly identified.

In parallel to the Portlet technology evolution, Portal development and Portal project management best practices emerged (IBM, 2004, *Architecting Portal Solutions* is one example). Consulting and research activities focus on the specialization of software development methods to Portal software projects and organizational changes initiated by Portal Projects. The *International Journal of Web Portals* will facilitate the communication, knowledge sharing and networking among Portal, web services, SOA researchers and engineers. Many Portals already found an ideal complement in Service Oriented Architectures, extending the concept of service decoupling and composition to an end-2-end view from the user interface to the backend system.

It is our intent to ensure the best value add for you, our dear reader, by providing you with guidance in this interesting
but complex ecosystem by a mixture of articles ranging from academic to project experiences and best practices.

Stefan Liesche, Ralf Grohmann
May 2008

REFERENCES


Stefan Liesche is the chief architect for WebSphere Portal Accelerators and the WebSphere Portal Platform, defining the technical roadmap for IBM’s Portal Product. Liesche holds a MSc in computer science from University of Hildesheim, Germany. In 1998 he joined IBM and gathered experiences in various technical roles. He worked as Architect for IBM Services in enterprise customer projects at various locations in Germany, where he constructed mission critical applications. Liesche has been working with Portal Technologies for years, in services projects, as well as within IBM’s Software Group product development organization. In 2007, Liesche gave a lecture on Portal technologies at the Friedrich-Schiller-University in Jena, Germany. Liesche is an appointed Senior Technical Staff Member (STSM) of IBM and The Open Group Master Certified IT Architect working in the IBM Deutschland Research and Development Laboratory, Böblingen, Germany.

Ralf Grohmann currently has the responsibility for WebSphere Portal Development in the IBM Lab in Boeblingen, Germany. His team delivers the Portal foundation, which is a major part of the WebSphere Portal Server, and also forms the basis for many other IBM offerings. He had the worldwide Product delivery responsibility for WebSphere Portal Version 6, available since July 2006. Prior to this, he had various other management responsibilities in Lab Based Services, Pervasive Computing, and Software development for the finance industry. In his technical career, he worked as a senior consultant and IBM certified IT Architect in IBM’s global services sector and defined, architected and implemented solutions based on Java. Ralf joined IBM in 1998. He holds a degree in computer science from the university of Karlsruhe, Germany.