

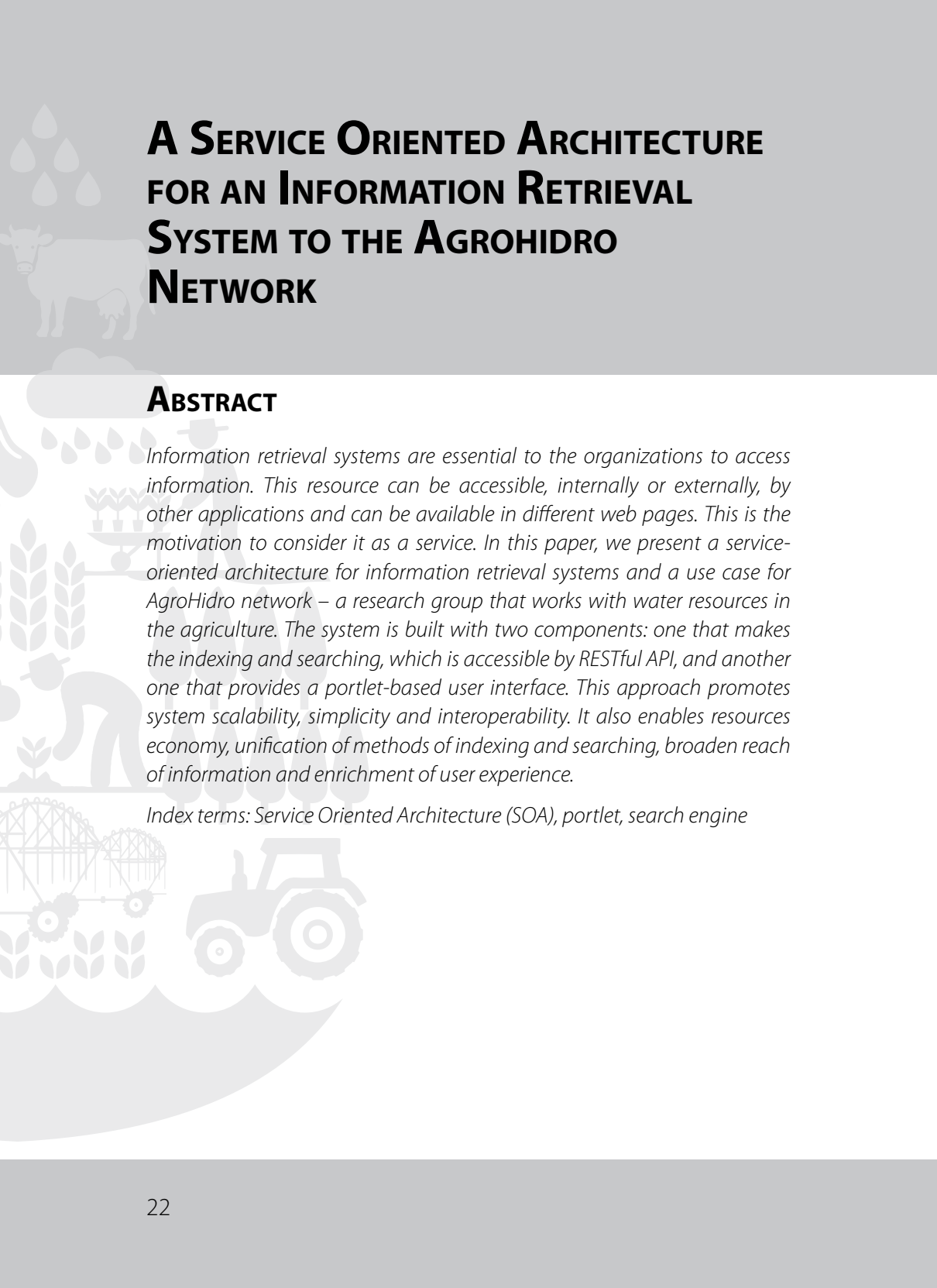
UMA ARQUITETURA ORIENTADA A SERVIÇOS PARA UM SISTEMA DE RECUPERAÇÃO DE INFORMAÇÃO PARA A REDE AGROHIDRO

GLAUBER JOSÉ VAZ

RESUMO

Sistemas de recuperação de informação são fundamentais para o acesso a informações nas organizações. Este recurso pode ser acessível, interna ou externamente, por outras aplicações e estar disponível em diferentes páginas web, o que motiva tratá-lo como um serviço. Neste artigo, apresentamos uma arquitetura de sistema de recuperação de informação orientado a serviço e um caso de uso para a Rede Agrohidro, grupo de pesquisa que trabalha com recursos hídricos na agricultura. O sistema é formado por dois grandes componentes, um, acessível via API RESTful, que faz indexação e busca, e outro, baseado em portlets, que oferece interface de usuário. A abordagem proposta promove escalabilidade, simplicidade e interoperabilidade de sistemas. Também favorece a economia de recursos, a unificação de métodos de indexação e de busca, maior alcance das informações e a melhor experiência dos usuários.

Termos para indexação: Service Oriented Architecture (SOA), portlet, mecanismo de busca.



A SERVICE ORIENTED ARCHITECTURE FOR AN INFORMATION RETRIEVAL SYSTEM TO THE AGROHIDRO NETWORK

ABSTRACT

Information retrieval systems are essential to the organizations to access information. This resource can be accessible, internally or externally, by other applications and can be available in different web pages. This is the motivation to consider it as a service. In this paper, we present a service-oriented architecture for information retrieval systems and a use case for AgroHidro network – a research group that works with water resources in the agriculture. The system is built with two components: one that makes the indexing and searching, which is accessible by RESTful API, and another one that provides a portlet-based user interface. This approach promotes system scalability, simplicity and interoperability. It also enables resources economy, unification of methods of indexing and searching, broaden reach of information and enrichment of user experience.

Index terms: Service Oriented Architecture (SOA), portlet, search engine

INTRODUÇÃO

Sistemas de recuperação de informação são fundamentais para que as organizações acessem seus dados e informações. Oferecidos como serviços, podem compor outras aplicações e podem ser acessíveis por públicos internos ou externos à organização, além de apresentarem inúmeras vantagens proporcionadas por uma abordagem de Service Oriented Architecture (SOA). Este trabalho visa o desenho de uma arquitetura de sistema de recuperação de informação orientado a serviço e posterior implementação de uma aplicação real, que, oferece, além de uma interface de usuário modular, uma interface para acessos realizados por aplicações. Especificamente para a Rede Agrohidro, grupo de pesquisa que trabalha com recursos hídricos na agricultura, um sistema de recuperação foi construído para acesso a produção bibliográfica relacionada a este tema.

MATERIAL E MÉTODOS

Service Oriented Architecture – Arquitetura Orientada a Serviço (SOA) é uma abordagem para abstrair as capacidades de software de uma organização como serviços reusáveis (BLOOMBERG, 2016). O uso de SOA apresenta vários benefícios, como custos menores, melhor consistência de aplicações e maior agilidade. SOA reduz redundância e produz sistemas modulares e interoperáveis que são mais fáceis de se usar e de se manter (GARTNER GROUP, 2016). SOA é um estilo de arquitetura baseado em serviços, que podem ser expostos de forma controlada e a audiências mais amplas através de APIs (HOLLEY et al., 2014). Uma Application Programming Interface (API), segundo Willmott e Balas (2013), especifica como componentes de software devem interagir uns com os outros, e uma Web API pode ser definida como uma interface para um componente de software que pode ser invocado à distância através de uma rede de comunicações usando tecnologias baseadas em padrões. Seu uso foi impulsionado pelo padrão REST associado ao formato JSON. Fielding (2000) define REST como um estilo de arquitetura para sistemas hipermídia distribuídos que fornece um conjunto de restrições

de arquitetura que enfatiza escalabilidade de interações de componentes, generalização de interfaces, implantação independente de componentes e componentes intermediários para reduzir latência de interação, reforçar a segurança e encapsular sistemas legados. JavaScript Object Notation (JSON) é uma formatação leve de troca de dados. Para seres humanos, é fácil de ler e escrever. Para máquinas, é fácil de interpretar e gerar. Está baseada em um subconjunto da linguagem de programação JavaScript (CROCKFORD, 2016).

Podemos entender um sistema de recuperação de informação em dois macrocomponentes: um que faz indexação e busca, e outro que exibe os resultados. Nesta proposta, o primeiro é baseado na tecnologia Elasticsearch e o segundo em portlets.

Elasticsearch é um mecanismo de busca de código aberto construído sobre o Apache Lucene, uma biblioteca para mecanismos de busca em texto reconhecidamente de qualidade e amplamente utilizada. Elasticsearch abstrai a complexidade da Lucene tornando bem mais simples a construção de mecanismos de busca em textos, oferece recursos adicionais e ainda viabiliza a utilização de suas funcionalidades via API Restful (GORMLEY; TONG, 2015).

Para a exibição dos resultados de busca, utilizamos portlets, que, conforme a especificação JSR-286 (HEPPER, 2008), corresponde a uma aplicação web que fornece um fragmento de conteúdo a ser incluído como parte de uma página de portal e que pode variar de acordo com o usuário e as configurações do portlet, fornecendo assim uma interface de usuário para serviços baseada em componentes. Os portlets possibilitam a integração consistente dos serviços e das interfaces de usuário. Há várias tecnologias de portal que implementam esta especificação. Uma delas é o Liferay Portal, considerada uma das melhores alternativas para a construção de portais (MURPHY et al., 2015) e utilizada para construir o sistema tratado neste trabalho. Vaz (2015) mostra que a implementação de um mecanismo de busca para a Rede Agrohidro com a tecnologia Liferay Portal viabiliza a construção de uma ferramenta de qualidade, possibilitando buscas em ambientes restritos e também adicionando conhecimentos específicos para melhorar seus resultados.

RESULTADOS E DISCUSSÃO

O mecanismo de busca proposto neste trabalho conta com dois componentes, um responsável pela indexação e busca, representado na Figura 1, pelo servidor Elasticsearch, e outro responsável pela interface com o usuário, composto pelos portlets A, B, C e D exibidos na página web e seus respectivos controladores.

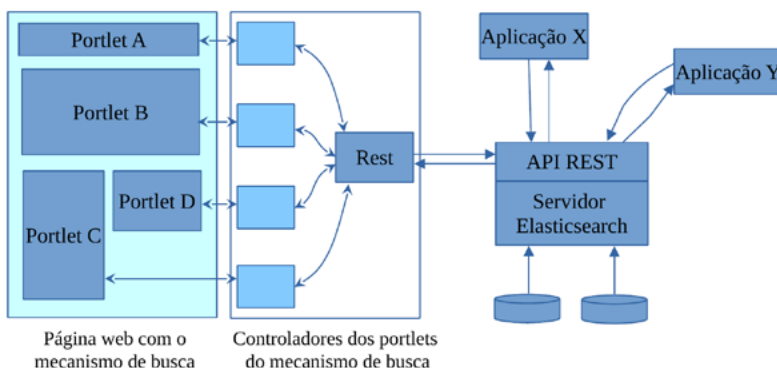


Figura 1. Arquitetura do sistema de recuperação de informação.

Ilustra-se ainda, na Figura 1, que outras aplicações, como X e Y, também podem fazer uso dos recursos oferecidos pelo servidor, que pode conter vários índices construídos a partir de diferentes fontes de dados. Com essa abordagem, não é necessário haver, na organização, índices replicados referentes às mesmas fontes de dados. Além de economizar recursos, unificam-se os métodos de indexação e busca, de maneira que os resultados sempre serão os mesmos para uma mesma consulta, independentemente de qual aplicação a utiliza. A comunicação entre as aplicações e o servidor é realizada através de requisições REST e as respostas obtidas são recebidas no formato JSON, padrão que facilita o intercâmbio de dados entre aplicações.

Do ponto de vista de interface de usuário, um mecanismo de busca pode ter inúmeros componentes, principalmente porque as possibilidades de visualização dos resultados de uma busca são ilimitadas, assim

como as análises que se desejam fazer com os dados recuperados e as alternativas de interação com o usuário para refinamento de uma busca. Apesar de esta aplicação poder ser implementada em um único portlet, que constitui a opção padrão para o Liferay Portal, por exemplo, a proposta desta arquitetura é baseada no uso de vários portlets, de maneira que outros portlets possam ser agregados para enriquecer a experiência dos usuários. A especificação Java de portlets (HEPPER, 2008) permite coordenação na camada de interface de usuário de diferentes maneiras para prover uma melhor integração entre os diferentes serviços.

O modelo de portlet segue o padrão Model-View-Controller (MVC), que estabelece a separação das camadas de controle e de geração de visão, representadas respectivamente na Figura 1, pelos controladores dos portlets e pela exibição dos portlets na página web. Cada portlet tem sua interface gerada por um controlador. E todos estes controladores acessam um mesmo elemento que envia as requisições REST para o servidor Elasticsearch, recebe as respostas em JSON e as analisa para consumo dos primeiros.

Esta arquitetura foi utilizada para implementar um mecanismo de busca específico para a Rede Agrohidro, composto inicialmente por quatro portlets: (a) um que contém o campo de texto, entrada para a busca; (b) um que exhibe documentos que constituem a resposta à consulta do usuário; (c) um que exhibe definições para os termos do domínio que aparecem nos resultados; e (d) outro que exhibe em um mapa regiões que são referenciadas em determinado documento anotado com coordenadas geográficas. Para atender a estes portlets, dois índices foram criados no componente de indexação e busca. No primeiro, foram indexados documentos da produção bibliográfica da Embrapa e documentos externos relacionados ao tema de interesse: recursos hídricos. No outro, foram indexados termos relacionados a este tema com suas respectivas definições. Embora tenham sido usados para uma aplicação específica, a arquitetura proposta e os portlets desenvolvidos são facilmente aplicáveis a outros domínios de interesse.

CONCLUSÕES

Apresentamos uma arquitetura orientada a serviço para um mecanismo de busca, baseada em dois grandes componentes, um que faz indexação e busca, e outro que oferece interface de usuário. A comunicação entre estes dois componentes e entre os elementos que os constituem é realizada de acordo com padrões e mecanismos bem estabelecidos, que possibilitam maior simplicidade, interoperabilidade e escalabilidade. Embora tenha sido base de um sistema específico para a Rede Agrohidro, esta arquitetura pode ser utilizada em diferentes contextos. O componente de indexação e busca pode ser consumido por parte de uma ou várias aplicações, internas ou externas, via API, viabilizando-se maior alcance, economia de recursos e unificação dos métodos de indexação e busca. Já os componentes de interface de usuário, representados pelos portlets, podem ser aproveitados em diferentes páginas web e podem ser estendidos com novos componentes a fim de enriquecer a experiência dos usuários.

REFERÊNCIAS

BLOOMBERG, J. Architecting the composable enterprise. In: MASON, R. **First, Break IT:** how composable services are changing the role of the CIO. pp. 36-43. Disponível em: <<https://www.mulesoft.com/lp/ebook/api/first-break-it>>. Acesso em: 19 fev. 2016.

CROCKFORD, D. **Introducing JSON.** Disponível em: <<http://www.json.org/>>. Acesso em: 24 mar. 2016.

FIELDING, R. T. **Architectural styles and the design of network-based software architectures.** 2000. Dissertation (Doctor of Philosophy) -- University of California, Irvine. Disponível em: <https://www.ics.uci.edu/~fielding/pubs/dissertation/rest_arch_style.htm>. Acesso em: 24 mar. 2016.

GARTNER GROUP. **Gartner IT Glossary:** Service-Oriented Architecture (SOA). Disponível em: <<http://www.gartner.com/it-glossary/service-oriented-architecture-soa>>. Acesso em: 21 mar. 2016.

GORMLEY, C.; TONG, Z. **Elasticsearch:** the definitive guide. 2015. Disponível em: <<https://www.elastic.co/guide/en/elasticsearch/guide/current/index.html>>. Acesso em: 24 mar. 2016.

HEPPER, S. **JSR 286**: portlet specification 2.0. Java Community Process. 2008. Disponível em: <<http://download.oracle.com/otndocs/jcp/portlet-2.0-fr-oth-Jspec/>>. Acesso em: 24 mar. 2016.

HOLLEY, K.; ANTOUN, S.; ARSANJANI, A.; BROWN, W. A. B.; COSTAS, J. F.; COZZI, C.; GOYAL, P.; IYENGAR, S.; JAMJOOM, H.; JENSEN, C.; LAREDO, J.; MADDISON, J.; NARAIN, R.; NATARAJAN, A.; PETRIUC, J.; RAMACHANDRAN, K.; RAVISHANKAR, R.; REINITZ, R.; VAIDYA, S.; VUKOVIC, M. **The Power of the API Economy**: stimulate innovation, increase productivity, develop new channels, and reach new markets. Armonk: Redbooks IBM, 2014. Disponível em: <<http://www.redbooks.ibm.com/redpapers/pdfs/redp5096.pdf>>. Acesso em: 24 mar. 2016.

MURPHY, J.; PHIFER, G.; TAY, G.; REVANG, M. **Magic quadrant for horizontal portals**. Setembro, 2015.

VAZ, G. J. Plataforma computacional para um mecanismo de busca de qualidade para a Rede AgroHidro. In: SEMINÁRIO DA REDE AGROHIDRO, 3., 2015, Corumbá, MS. **Água na agricultura**: desafios frente as mudanças climáticas e de uso da terra: resumos. Brasília, DF: Embrapa, 2015. p. 52. Disponível em: <<http://ainfo.cnptia.embrapa.br/digital/bitstream/item/134800/1/Agrohidro-plataforma-Vaz.pdf>>. Acesso em: 18 jul. 16.

WILLMOTT, S.; BALAS, G. **Winning in the API Economy**: using software and APIs to transform your business, drive revenues, broaden distribution and unleash innovation. [S.l.]: 3Scale, 2013. Disponível em <<https://www.3scale.net/wp-content/uploads/2013/10/Winning-in-the-API-Economy-eBook-3scale.pdf>>.