
Integração de Sistemas

Relatório do 1º projeto

Bruno José Borges Madureira - 2011161942

Fábio Alexandre Martins Moura - 2013118534



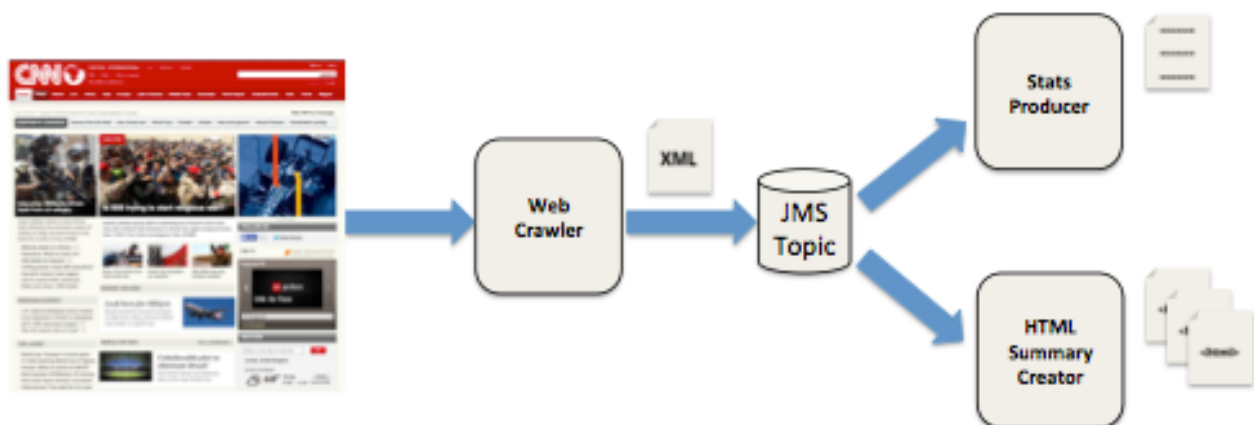
Introdução

O objetivo deste trabalho consistiu em integrar sistemas sem que eles estivessem construídos de forma a serem interligados. O *site* da “CNN” não disponibiliza nenhuma API para que programas/sistemas tenham acesso aos seus conteúdos, dificultando assim a integração com outros sistemas. Nestes casos, um método viável para integrar um sistema deste género é através da técnica de “*Screen Scraping*”. Esta técnica consiste em fazer *parsing* à informação mostrada num terminal para que possa ser usada num sistema diferente.

O enunciado pedia que um programa fizesse *parsing* à página e enviasse o resultado para dois programas independentes. Um deles é responsável por gerar *html* com o conteúdo do site e o outro é responsável por guardar informação sobre as notícias das últimas 12 horas.

Para fazer o que é pedido é necessário dominar a tecnologia XML e usar as ferramentas Jboss 8.1 (WildFly) e Jsoup.

Arquitetura usada



Como ilustrado nesta imagem retirada do enunciado, o Web Crawler é responsável por fazer *parsing* ao conteúdo do *site* e enviá-lo aos outros dois programas. Antes de enviar, o Web Crawler transforma todas as classes que contêm os dados do site num ficheiro *xml* que é enviado em formato de *String* numa JMS *message*, que é publicada num JMS Topic.

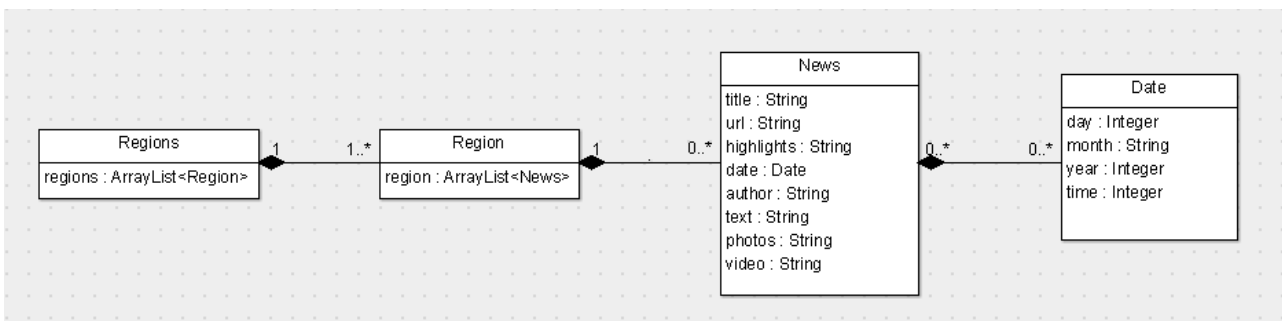
Tanto o *stats producer* como o HTML Summary Creator estão sempre ligados a espera de mensagens do JMS Topic.

O Html Summary Creator recebe o *xml* numa *string*. De seguida altera essa *string* adicionando uma linha extra que irá ligar este html a um ficheiro de xsl. Por fim esta *string* é guardada num ficheiro. Com a linha que foi adicionada, dá-se ao browser que abrir aquele *xml* indicações de “estilo”, que permitem uma visualização de dados mais fácil para nós, humanos.

O Stats Producer é responsável por apresentar apenas as notícias com menos de doze horas. O seu modo de funcionamento é simples: este encontra-se à escuta de mensagens do JMS Topic, à espera de uma *string xml*, e quando finalmente recebe uma faz unmarshall dos dados para objectos em java. Após isso analisa todas as notícias que recebeu comparando a hora atual com a hora da notícia, se a diferença for menor que doze horas então apresenta o título da mesma no visor juntamente com a descrição de à quanto tempo foi publicada e por fim guarda uma cópia, em *xml*, com as notícias que cumprem o requisito.

Modelo de dados

A imagem seguinte mostra o modelo de dados da aplicação, para tal foi preciso primeiro identificar os atributos de cada notícia no site da CNN, construir um *xml* de exemplo com dados de notícias e usar as ferramentas Trang e xjc para gerar as respetivas classes.



Tecnologias usadas

Foi principalmente usado *xml* e tecnologias adjacentes, o *xml* foi usado para guardar os dados das notícias. Para serem criadas classes que representassem o conteúdo do *xml* foi usada a ferramenta “Trang” que criou um ficheiro *xsd*. Para, a partir do *xsd*, gerar as

classes foi necessário usar outra ferramenta, o *xjc*. Tanto o ficheiro *xsd* como as classes foram alteradas à mão para representarem o problema de forma mais concreta.

Para fazer a conversão entre *xml* e classes usámos a biblioteca JAXB.

Para fazer a ligação entre os 3 programas usámos o Jsoup 8.1 (Wilfly), e para fazer trocas de mensagens persistentes foi necessário utilizar tópicos em vez de simples filas.

Exceções tratadas

Tal como pedido no enunciado, caso o Crawler não consiga ligar-se ao Topic ele fica em *loop* a tentar conectar-se. Caso o utilizador desligue o programa sem que ele tenha enviado com sucesso, mas tenha feito o parse da página, é guardado um ficheiro *xml* com o conteúdo a enviar. Na próxima vez que for ligado o programa é enviado esse ficheiro, não sendo necessário fazer parse à página.

Divisão do trabalho

	Horas	Parte
Bruno Madureia	5 horas de estudo 15 horas de implementação	Web Crawler Html Creator Relatório
Fábio Moura	4 horas de estudo 13 horas de implementação	Configuração Jboss Stats Producer Relatório Transformações de estilo/ aspecto ao xsd

Bibliografia

<http://jsoup.org/>

<http://www.w3schools.com/xml>

<http://www.w3schools.com/schema>

<http://www.w3schools.com/xpath>

<http://jaxb.java.net/tutorial/index.html>

<http://www.thaiopensource.com/relaxng/trang.html>

<http://docs.oracle.com/javaee/7/api>

<http://www.digilife.be/quickreferences/pt/introducing%20the%20java%20message%20service.pdf>

<http://eai---course.blogspot.pt>

<http://krishnasblog.com/2012/12/02/jsoup-nice-way-to-do-html-parsing-in-java/>