

---

**E-Business Technologies – Lab – Winter Term 2014/15**

**Assignment 3: XML Processing with Java  
– Implementation of a Simple E-Business  
System for Wholesalers –**

**\*\*\* 40 Points \*\*\***

---

## **1 Overall Scenario**

Assume you are a wholesaler for ICT products and you have to implement an E-Business system with product data management functionality. This system should be able to import product catalogs from your suppliers. Furthermore, the system should be able to export your product catalog to your customers. Fig. 2 depicts the overall scenario.

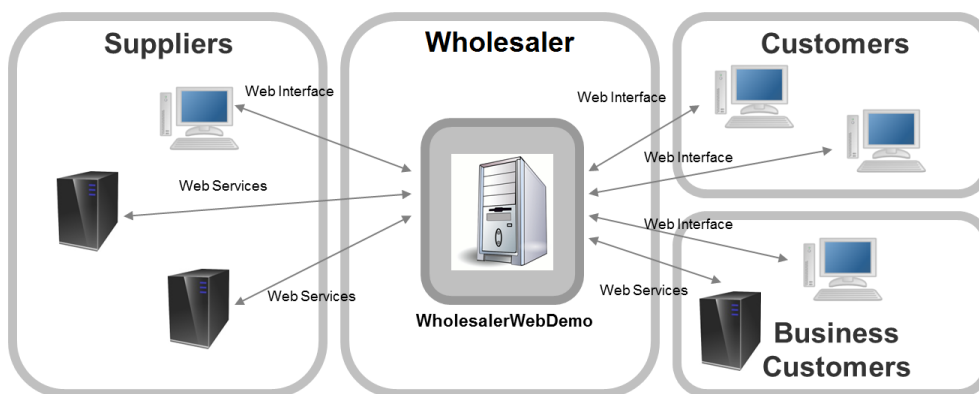


Figure 1: Overall Scenario.

## **2 Initial Situation**

You already have a basic E-Business system running (the *WholesalerWebDemo*) which is based on the *E-Business Software Framework* (EBSF). The EBSF implements the

data management layer (based on a DBS and an OR mapping) and a simple business logic layer for E-Business applications. It also offers an application programming interface (API) for the business logic layer.<sup>1</sup>

### 3 Assignment

Design and implement the following *import / export* functionality for your E-Business system meeting the following functional requirements.

- Import product catalogs (*simple BMEcat XML-format*) into your existing E-Business system.
- Export articles from your E-Business system into a product catalog (*simple BMEcat XML-format*).
- Export articles from your E-Business system as a *simple XHTML* document.
- The export functionality has to offer *two options*:
  - Export of the whole product catalog (i.e., all articles).
  - Export of articles only where the short description matches a given search string (i.e., substring matching, not exact matching).
- Export and import functionality should be available via a *Web GUI*.

Non-functional requirements:

- The language of the Web GUI has to be *English*.
- Use the *Document Object Model* (DOM) as *API* to implement the import and export functionality.
- Use the *XML validation package* from Java to validate the XML document against the available simple BMEcat XML format etc.
- Using the *XML XPath package* from Java may ease DOM processing for the product catalogue import.
- Use *XSLT* together with the *XML transformation packages* from Java to create the (valid) XHTML output.
- You only have to use the *Java 7 APIs* (JAXP) for all XML processing (no JDOM, dom4j, etc.).
- Comment your code *sufficiently* (classes, attributes, methods, etc.) in *English*. You can use JavaDoc if you wish.

---

<sup>1</sup>You can find detailed information on *Moodle*.

## 4 Helpful Hints

All necessary resources, e.g., installation instruction, the simple BMEcat XML schema, a sample import XML product catalog (`articles_kn_media_store.xml`), etc. are available on Moodle.

- Create a supplier in your database first to enable product catalog import for this supplier.
- Design and implement your import and export functionality in a modular way, so you are able to (re-)use it in several contexts, e.g., within a Web GUI or to implement Web service interfaces later on.
- The upload function via the Web GUI *could* be realized, e.g., with the Apache File Uploader<sup>2</sup> (packages: `commons-fileupload-x.jar` and `commons-io-x.jar` - where 'x' stands for the latest version).
- It is not necessary to use temporary files on the server.
- If you decide to use the *WholesalerWebDemo* application as a basis for your solution, use the following two packages (within your *WholesalerWebDemo*):
  - `de.htwg_konstanz.ebut.main` → for your import-/export code,
  - `de.htwg_konstanz.ebut.action` → for your *IAction* implementations.

Do not forget to announce your action Java package to the `ControllerServlet.java` class! Use *no* additional servlet.

- It is recommended to use Checkstyle<sup>3</sup> to ensure code quality standards.
- Use the W3C Markup Validation Services<sup>4</sup> to validate your XHTML output.

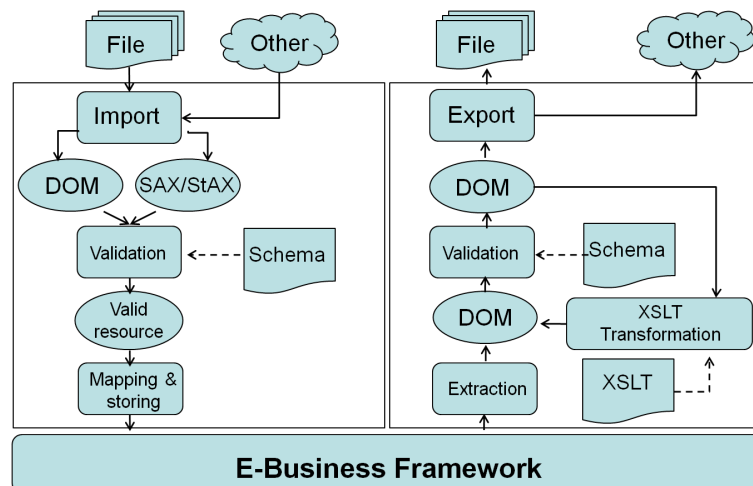


Figure 2: Possible "architecture" for catalog import and export.

A possible "architecture" for your catalog import and export is depicted in Fig. 2.

<sup>2</sup>[commons.apache.org/proper/commons-fileupload/](http://commons.apache.org/proper/commons-fileupload/)

<sup>3</sup>[checkstyle.sourceforge.net/](http://checkstyle.sourceforge.net/)

<sup>4</sup>[validator.w3.org/](http://validator.w3.org/)

## 5 Acceptance

For the successful acceptance of your solution for assignment 3, a *checklist* is defined. It will be used for your demonstration and to assess your solution. The following topics are included, among others, in the checklist:

- Catalog import/export with DOM, validation, export of XHTML, full/selective export, etc.
- Upload/download via Web GUI.
- Update semantics for catalog import.
- User experience (incl. failure handling, error messages, warnings, etc.).
- Your explanation of your own source code incl. comments included in your source code.

Deadline for demonstration of your assignment solution is: **December 16, 2014.**

**Good luck!**