



УЧЕБНА ПРОГРАМА

Име на курса:	Архитектура ориентирана към услуги (SOA) и съвременни стандарти за моделиране на бизнес процеси (BPM)
Код на курса:	IB-05
Професионално направление:	Софтуерно инженерство
Брой модули:	1
Продължителност на всеки модул:	40 учебни¹ (30 астр.) часа
Обща продължителност на курса:	40 учебни (30 астр.) часа
Форма на обучение:	Интензивен /Съботно- Неделен

София, 2012 година

Copyright © 2003-2012 IPT – Intellectual Products & Technologies Ltd. All rights reserved.

¹ Учебните часове са по 45 минути.



УЧЕБНА ПРОГРАМА

на курса: **Архитектура ориентирана към услуги (SOA) и съвременни стандарти за моделиране на бизнес процеси (BPM)**

УЧЕБЕН ПЛАН

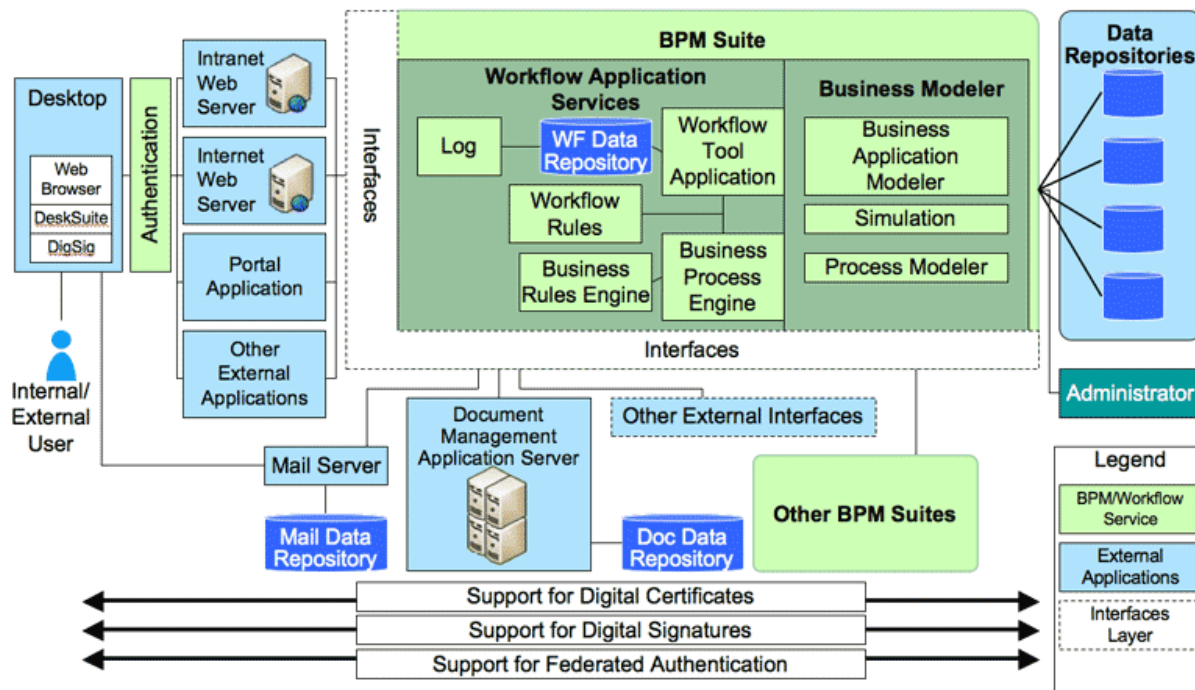
Име на модул	1. Архитектура ориентирана към услуги (SOA) и съвременни стандарти за моделиране на бизнес процеси (BPM)
Лекции, уч. часа	20
Практически занятия, уч. часа	20
Общо, уч. часа	40
Форма на оценка	ТЕСТ
Сертификат	IPT

Преподавател:

Траян Илиев

IPT – Intellectual Products & Technologies Ltd.

E-mail: tiliev@iproduct.org



Пример за Business Process Management (BPM) Service. Този шаблон показва как инструментите за управление на бизнес процеси (BPM) могат да бъдат използвани за реализация на бизнес процеси чрез оркестрация и хореография на дейности изпълнявани от хора и софтуерни системи.²

² Източник: National Institute of Health (2007). Business Process Management (BPM) Service Pattern – <http://enterprisearchitecture.nih.gov/ArchLib/AT/TA/WorkflowServicePattern.htm>



Анотация:

Курсът *Архитектура ориентирана към услуги (SOA) и съвременни стандарти за моделиране на бизнес процеси (BPM)* запознава с множеството взаимно-свързани и развиващи се технологии и стандарти, които дават възможност за моделиране на бизнес процеси с помощта на бизнес-ориентирана графична нотация, тяхната автоматизация с помощта на архитектури базирани на уеб услуги по един многократно използваем, гъвкав и независим от IT платформата начин, както в рамките на една бизнес организация, така и между различните участници в цялостна бизнес екосистема.

Курсът започва с преглед на основните стандарти и базираните върху тях технологии за реализация на уеб услуги:

- *eXtensible Markup Language (XML), XML Schema* – основа за реализация на платформено независими бизнес комуникации;
- *Simple Object Access Protocol (SOAP)* – W3C стандарт позволяващ обмяна на информация между уеб услуги реализирани на различни IT платформи в Remote Procedure Call (RPC) и документен стил;
- *Web Services Description Language (WSDL)* - W3C стандарт създаден за цялостно описание на начина по който могат да бъдат извиквани уеб услугите от множество клиенти, включително други услуги;
- *Representational State Transfer (REST)* – един нов начин за използване на съществуващия HTTP протокол за реализация на „олекотени“ откъм XML уеб услуги.

Следва разглеждане на по-широкия контекст, в който тези стандарти за реализация на услуги биха могли да допринесат за осигуряване на конкурентни предимства за бизнеса и за справяне с предизвикателствата: глобализация, икономически натиск за постигане на по-висока ефективност в условия на криза, аутсорсинг на бизнес процеси, регулаторно съответствие, нови технологии, липса на кохерентна бизнес-информационна стратегия, многообразие от стандарти. Архитектурата ориентирана към услуги (*Services-Oriented Architecture – SOA*) включва:

- разпределени ресурси пресичащи границите на отделния бизнес;
- хора и системи взаимодействащи помежду си, също пресичайки организационните граници;
- разпределени сигурност, управление на тези ресурси;
- взаимодействието между хора и системи чрез обмяна на съобщения, с изискване за надеждност подходяща за целите на използване.



От холистична перспектива една SOA базирана система е мрежа от независими услуги, машини и хора, които оперират, взаимодействат, използват и управляват тези услуги, а също и доставчиците на оборудване и персонал за тези хора и услуги. При толкова голяма система е ясно, че никой не контролира или управлява цялата екосистема – въпреки че има въввлечени много участници, всеки от тях има известен контрол и влияние върху общността.

Service Component Architecture – SCA е практическият начин за реализиране на SOA чрез дефиниране на изпълним модел за изграждане на приложения ориентирани към услуги като композиция (мрежа) от отделни компоненти (уеб услуги). Тази архитектура включва *Service Data Objects – SDO* като унифициран модела на работа с използваните от услугите данни, независимо от техния източник или получател.

Третата основна част на курса включва детайлно разглеждане и добиване на практически опит в бизнес процес моделирането чрез използване на два взаимно допълващи се стандарта:

- *OMG Business Process Model and Notation (BPMN 2.0)* – актуална спецификация на *Object Management Group (OMG)* от януари 2011 дефинираща графична нотация за моделиране на бизнес процеси в рамките на бизнес функцията или организацията (*private*), наречен още работен поток (*workflow*), между процеса и външни участници или системи (*public*) и между различните организации (*global*), включваща мета-модел и файлов формат за обмяна на описания на бизнес процеси между различни софтуерни системи, и позволяваща както *оркестрация* в рамките на организацията, така и *хореография* на бизнес процеси реализирани от различни организации;
- *OASIS Web Services Business Process Execution Language (WS-BPEL)* – възприет от индустрията стандарт за реализация на изпълними бизнес процеси чрез интегриране на уеб услуги с множество реални продукти от различни доставчици, които го поддържат.

Практическите занятия по време на курса ще включват разглеждане на примери за използване на стандартите и самостоятелна реализация на уеб услуги и клиентски приложения в среди Eclipse и NetBeans, реализация на композитни уеб услуги с Apache Tuscany (SCA), графично моделиране на бизнес процеси с BPMN и оркестрация с WS-BPEL.

От участниците се очаква да притежават практически опит в разработката на софтуер с Java™³ и XML.

³ Oracle®, Java™, JSP™, and Servlet™ are trademarks or registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.



СЪДЪРЖАНИЕ НА УЧЕБНАТА ПРОГРАМА

А. Лекции.

Модул 1: Архитектура ориентирана към услуги (SOA) и съвременни стандарти за моделиране на бизнес процеси (BPM)

1. Кратък преглед на базовите стандарти за обмяна на бизнес информация: *eXtensible Markup Language (XML)*, *XML Schema*, *Simple Object Access Protocol (SOAP)*, *Remote Procedure Call (RPC)* и документен стил на *SOAP*. *Web Services Interoperability Organization (WS-I)* профили. Основни елементи и структура на *SOAP* съобщение. Обработка на грешки със *SOAP*. Модел за обработка на *SOAP*. Използване на различни протоколи за пренос на *SOAP*. (2 ч.)
2. *Web Services Description Language (WSDL)* – предназначение и структура. Основни блокове от дефиницията на веб услуга в *WSDL 1.1*: `<types>`, `<message>`, `<portType>`, `<binding>` и `<service>` и техните елементи. Новости в *WSDL 2.0*: `interface`, `service endpoint`. Поддръжка на множество крайни точки от веб една услуга. Автоматично генериране на *WSDL* описания на веб услуги. Автоматично генериране на клиентски код (`web service reference`) за достъп до веб услуга по *WSDL* описание. Реализация на класически веб услуги на *Java™* с *JAX-WS API*. Основни анотции: `@WebService`, `@WebMethod`, `@WebParam`, `@WebResult`, `@WebServiceRef`, `@WebServiceRefs`, `@Addressing`, `@Action`, `@FaultAction`, `@WebFault`, `@WebServiceClient`, `@WebEndpoint`. Примери. (2 ч.)
3. Оптимизация на *SOAP* преноса – *XML-binary Optimized Packaging (XOP)*, *Message Transmission Optimization Mechanism (MTOM)*, *Fastinfoset*. Надеждност и сигурност при реализацията на веб услуги – *Web Services Addressing*, *Web Services Secure Conversation*, *Web Services Policy*, *Web Services Reliable Messaging*, *Web Services Trust*, *Web Services Secure Conversation*, *Web Services Security Policy*. Сравнение между най-разпространените рамки (`web services stacks`) за реализация на веб услуги – *Glassfish Metro*, *Apache Axis2*, *Apache CXF*. (3 ч.)
4. *Representational State Transfer (RESTful) web services* – олекотени, ресурсно ориентирани веб услуги, особено удобни за клиентски достъп чрез *JavaScript* и *AJAX*. Достъп до ресурса и предаване на параметри чрез *URI*. Използване на основните методи на *HTTP* протокола (`POST`, `GET`, `PUT` и `DELETE`) за модификация, достъп, създаване и изтриване на ресурси по техни представяния. Алтернативи при сериализация на ресурсите – *XML*, *JavaScript Object Notation (JSON)*, *plain text*. Реализация на *RESTful web services* на *Java™* с *Jersey (JSR-311: JAX-RS - Java API for RESTful Web Service)* – директна експозиция на *POJOs (Plain Old Java Objects)* като веб ресурси чрез прилагане на набор от стандартни



анотации и интерфейси. Анотации: @Path, @PathParam, @QueryParam, @FormParam, @HeaderParam, @CookieParam, @MatrixParam, @DefaultValue, @Encoded, @Context, @Produces, @Consumes, @HttpMethod, @GET, @POST, @PUT, @DELETE и др.. *Web Application Description Language (WADL)* – олекотена алтернатива на WSDL за описание на уеб услуги. Автоматична сериализация на обекти в XML и JSON формат с Java™ Architecture for XML Binding (JAXB). Разработка на AJAX клиенти за уеб услуги в стил REST. Реализация на декларативна сигурност при услуги в стил REST. Примери. (4 ч.)

5. *Service Oriented Architecture (SOA)* – обща идея и визия за развитието на ИТ за удовлетворяване на променящите се нужди на бизнеса. SOA референтна архитектура. *SOA Governance*. Разглеждане на стандарта *OASIS Reference Architecture for Service Oriented Architecture Version 1.0*. Управление на жизнения цикъл на услугите в SOA – добри практики, шаблони, коментари върху стандартите. Интеграция с помощта на SOA и перспективи за развитие. (2 ч.)
6. *Service Component Architecture – SCA*. Реализация на SOA чрез SCA. Уеб-услугите като многократно използвани компоненти. SCA сценарии и спецификации. OASIS SCA Tcs. Асемблиране на платформено независими уеб услуги чрез SCA. Управление качеството (QoS) на уеб услугите чрез SCA политики – SCA Policy Framework V1.0. SCA стандарти за реализация за различни платформи – Java, C, C++, BPEL. Проекти с отворен код – *Apache Tuscan* (2 ч.)
7. *OMG Business Process Modeling and Notation (BPMN)* – графична нотация за моделиране на бизнес процеси в рамките на бизнес функцията или организацията (private – workflow, public, global). Основни елементи на графичната нотация – дейности (activity), събития (events), входове (gateways), процеси и участници във взаимодействието (pools & swimlanes), данни (data), транзакции и документация (групи и анотации). Оркестрация и хореография на бизнес процеси. Елементи на BPMN модела. BPMN -> BPEL mapping. (2 ч.)
8. *OASIS Web Services Business Process Execution Language (WS-BPEL)* – възприет от индустрията стандарт за реализация и интеграция на изпълними бизнес процеси. Основни елементи на спецификацията. Платформена поддръжка на BPEL. Примери за използване. (2 ч.)
9. Финален тест. (1 ч.)



Б. Практически занятия

Практическите занятия за всеки от модулите използват проблемно-ориентирана методология за обучение. Разглеждането на примерите и разработката на собствени компоненти и приложения се осъществява в интегрирани среди за разработка *Eclipse* (<http://www.eclipse.org/>) и *NetBeans* (<http://www.netbeans.org/>), в комбинация с приложни/ уеб сървъри *GlassFish v3* и *Apache Tomcat 7*, които предлагат всички необходими средства за проектиране, разработка, внедряване и тестване на уеб услуги и клиентски приложения за тях. Използват се специализирани средства за графично моделиране на бизнес процеси с BPMN и WS-BPEL.

Обучението се провежда в малки групи – до 10 човека. Към участниците в курса се прилага индивидуален подход. По време на упражненията ще има възможност за дискутиране на въпроси от практиката на участниците.

ОСНОВНА ЛИТЕРАТУРА И ИНТЕРНЕТ РЕСУРСИ

1. Хънтър, Д., Програмиране с XML. Софтпрес 2001
2. Закас, Н, Пийк, Д., Фосет, Д., Професианлно програмиране с AJAX. Алекс Софт, 2006.
3. World Wide Web Consortium (W3C) homepage – <http://www.w3.org/>
4. Object Management Group OMG homepage – <http://www.omg.org/>
5. SOA Blueprint homepage – <http://soablueprint.com/>
6. OASIS OpenCSA homepage – <http://www.oasis-opencsa.org>
7. Oracle® / Sun Java™ Technologies webpage – <http://java.sun.com/>
8. Glassfish Metro Web Services Stack – <http://metro.java.net/>
9. Glassfish Jersey RESTful Web Services Stack – <http://jersey.java.net/>
10. Apache CXF Open Source Web Services – <http://cxf.apache.org/>
11. Apache Axis2 Web Services Stack – <http://axis.apache.org/axis2/java/core/>
12. Eclipse Application Development Framework – <http://www.eclipse.org/>
13. NetBeans IDE – <http://www.netbeans.org/>
14. GlassFish Application Server – <https://glassfish.dev.java.net/>
15. Apache Tomcat Web Server – <http://tomcat.apache.org/>