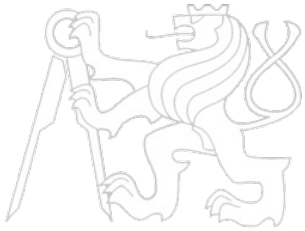
Middleware and Web Services

Motivation and Course Overview

doc. Ing. Tomáš Vitvar, Ph.D.
tomas@vitvar.com • @TomasVitvar • http://vitvar.com



Czech Technical University in Prague
Faculty of Information Technologies • Software and Web Engineering • http://vitvar.com/courses/mdw

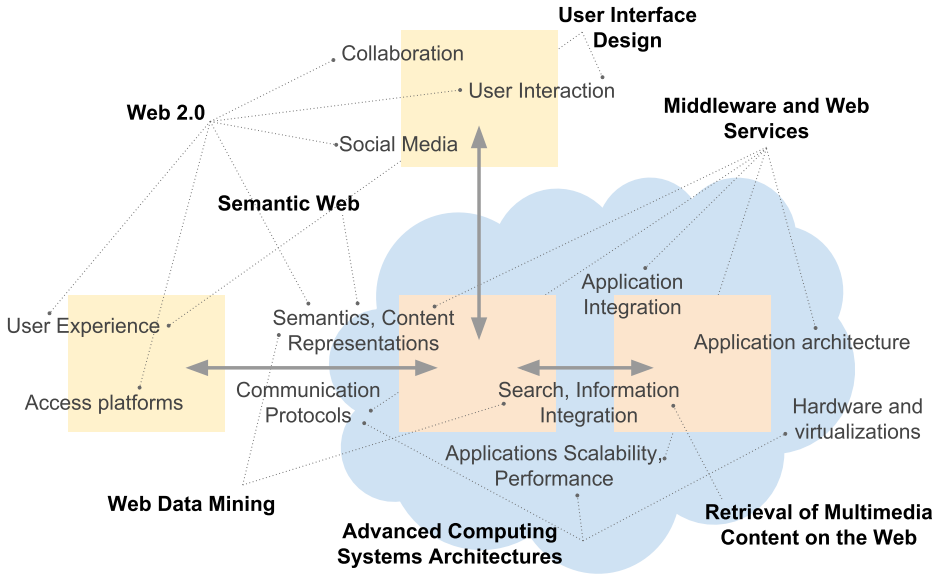
Modified: Sun Sep 30 2018, 21:19:05
Humla v0.3

Hellos

- Tomáš Vitvar – lectures (Czech)
  - *Web engineering study programme chair at CTU FIT*
  - *Technology Architect Director at Oracle*
  - *Research*
    → *Web Intelligence, Semantic Web, Linked Data, Web Services*
  - *Industry*
    → *Large scale integration architecture projects (Vodafone UK, IKEA IT Sweden, Turkish Telecom)*
    → *Design, governance, troubleshooting, performance tuning*
- Jaroslav Kuchař – labs (Czech)
  - *Research assistant at CTU FIT, Seznam.cz*
  - *Web usage mining, big data analytics, Web services*
- Milan Dojčinovski – lectures and labs (English)
  - *research assistant at CTU FIT, Ph.D. candidate*
  - *Semantic Web, Linked Data, NLP, Web services*

Web Engineering Curricula – bird's view

- Web 2.0 engineering is...
  – *far beyond PHP and HTML!*



Overview

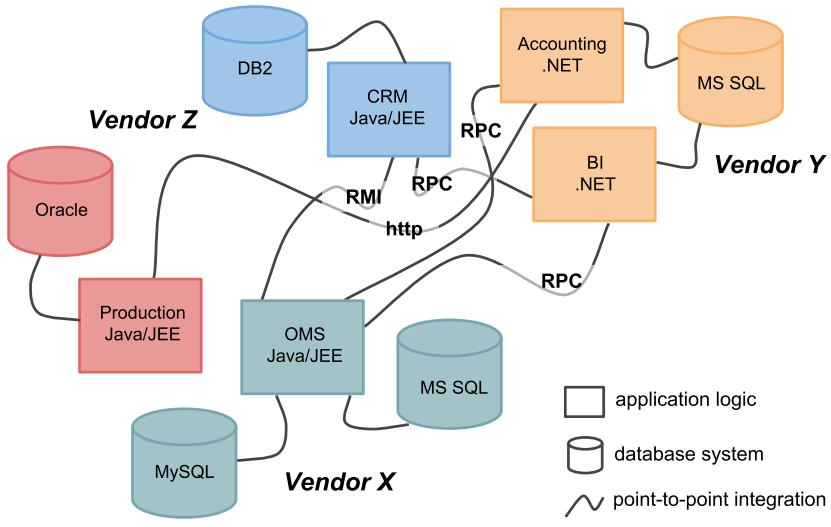
- Course at a Glance
  – *Motivation and Scope*
  – *Requirements and Organization*
- Assessment
- Communication and Resources

Motivation in Brief

- Systems rely on complex infrastructures
  - *A lot of data and many processes, internal and external*
  - *As people communicate, underlying systems must too*
  - *But:*
    - → *variety of data formants, technologies, protocols*
    - → *variety of architectures, client-server, peer-to-peer, ...*
- Good performance
  - *frequent changes in applications' loads, peek hours*
  - *scalability – effective load balancing*
  - *low costs – cheaper to outsource?*
- Rapid changes in applications' functionality
  - *modular development*
  - *reuse of application functionality*
  - *low costs – do it now and quickly!*

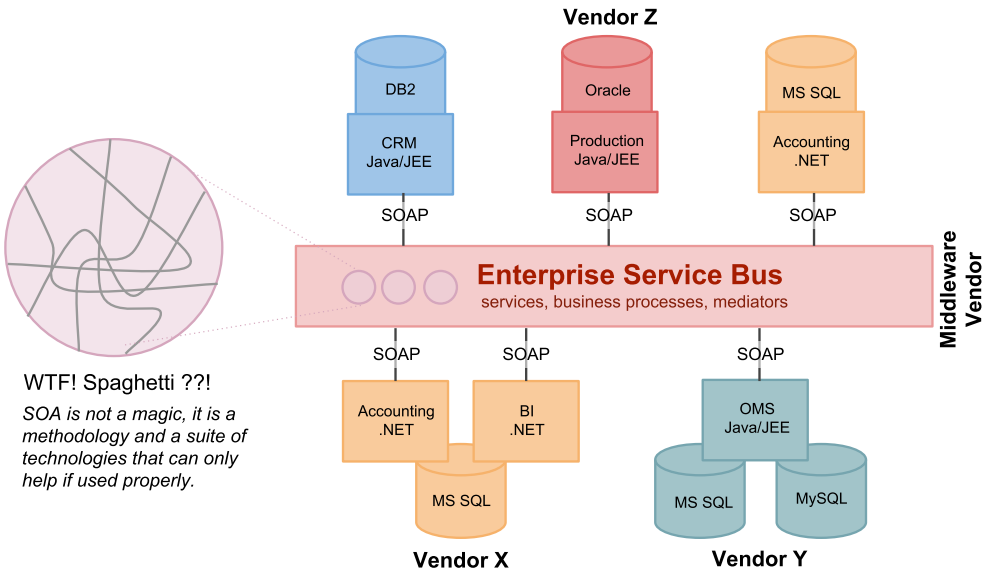
Spaghetti Architecture

- Need for the integration
  - *One-to-one integration*
  - *Hard to maintain, vendor interoperability problem*



SOA Architecture

- Integration organized
  - *Enterprise Service Bus, to be used wisely*



Scope

- Architectural and conceptual basis
  - *What is architecture – enterprise, processes, data, software*
  - *Service Oriented Architecture, Service Concepts, Middleware*
- Web Service technologies
  - *Details of HTTP and REST, SOAP*
  - *Web Service Description Language (WSDL)*
- Infrastructure and Middleware
  - *Performance, Scalability*
  - *Application server*
- Cloud
  - *Microservices, Docker*

Overview

- Course at a Glance
  - *Motivation and Scope*
  - *Requirements and Organization*
- Assessment
- Communication and Resources

Prerequisites

- Object-oriented programming
  - *Principles*
    - → *class, object, inheritance, encapsulation, ...*
      - → *basis for service concepts*
- Java
  - *All code examples will be in Java*
  - *All lab work will be in Java*
- Web Architecture
  - *Basics of XML, XPath, HTTP, URI*
- Other
  - *Networking*
  - *Regular expressions, basics of Linux*

Organization of Lectures

- 12 Lectures
  - *Czech: Mon 9:15-10:45, TK:BS*
  - *English: TBA*
- Plan
  1. *01.10.2018 – Motivation and Course Overview (html)*
  2. *08.10.2018 – Introduction to Architectures (html)*
  3. *15.10.2018 – Service Architecture and Technologies 1 (html)*
  4. *22.10.2018 – Service Architecture and Technologies 2 (html)*
  5. *29.10.2018 – Service Architecture and Technologies 3 (html)*
  6. *05.11.2018 – Application Server 1 (html)*
  7. *12.11.2018 – Application Server 2 (html)*
  8. *19.11.2018 – Messaging Systems (html)*
  9. *26.11.2018 – High Availability and Performance 1 (html)*
  10. *03.12.2018 – High Availability and Performance 2 (html)*
  11. *10.12.2018 – Cloud Architectures (html)*
  12. *17.12.2018 – Microservices and Docker (html)*

Organization of Labs

- Individual work (no teams!)
- Labs every second week
- Number of labs: 6
  1. *Introduction - Setup, Simple Web Application*
  2. *REST – Basics, development of a REST service*
  3. *REST advanced*
  4. *SOAP Web services*
  5. *Messaging Services - JMS*
  6. *Web Logic Metrics and Load Balancing*

Methodology for Lab Work

- No app development, not directly related assignments
  - *assignment every second week*
  - *be prepared for the lab!*
  - *work alone, ask others for advices*
  - ***Results:***
    → *5 completed tasks*
    → *you will submit results to gitlbab@FIT*

Overview

- Course at a Glance
- Assessment
- Communication and Resources

Assessment

- Labs
  - *Presence is mandatory*
    → *You can miss up to 1 lab without sending regrets*
  - *Every task gives you the maximum of* 5 *points*
  - $5 \cdot 5 = 25$ *points*
  - *Activity in labs gives you the maximum of* 15 *points*
  - *Total maximum points:* $p_p = 40$, ***to pass***: $p_p \geq 20$
- Final exam
  - *Written exam: 3 exercises, 1 hour*
    → *each gives you a max. of* 20 *points, the total* $p_t = 60$ *points*
    → *To pass, you need to have at least* 50% *from each exercise!*
  - *Final score:*
    → $p_p + p_t$ 100 *maximum points*

Final Marks

Mark	Points	In words
A	100–90	výborně
B	89–80	velmi dobře
C	79–70	dobře
D	69–60	uspokojivě
E	59–50	dostatečně
F	49–0	nedostatečně

Source: http://www.cvut.cz/pracoviste/pravni-odbor/dokumenty/studijni-predpisy/studijnirad.pdf

Overview

- Course at a Glance
- Assessment
- Communication and Resources

Communication

- Language
  - *Text: English (slides, tweets, posts, instructions, etc.)*
  - *Voice: Czech and English (English version of the course)*
- Direct
  - *you can always contact me directly at tomas@vitvar.com or @TomasVitvar*

Overview of Resources

- Overview of resources

Item	URL
Course slides	http://mdw.vitvar.com
Courses@FIT	https://courses.fit.cvut.cz/MI-MDW/
Lab projects	https://gitlab.fit.cvut.cz/
Assessment	https://grades.fit.cvut.cz/courses/MIE-MDW.16/

- Books
  - *Jiří Voříšek: Strategické řízení informačního systému a systémová integrace. Management Press, 1997.*
  - *L. Richardson, S. Ruby: RESTful Web Services: Web services for the real world, O'Reilly Media, May 2007, ISBN 9780596529260.*
  - *Jon Mountjoy, Avinash Chugh: WebLogic: The Definitive Guide. O'Reilly Media, Inc., 2004*
  - *Thomas Erl: Service-Oriented Architecture: Concepts, Technology, and Design. Prentice Hall, Aug 2, 2005.*

About Slides

- Humla – Open Source HTML5 Presentation System
  - *every slide has a unique URL*
  - *all figures linked with Google drawings*
  - *auto-generated PDFs (1 and 2 slides per page) using travis-ci*
  - *running local (with local nodejs-based http server), and in github pages*
  - *Suggest edits or correct errors by pull requests at mdw github repo*
- Keys
  - 1 *default browsing mode*
  - 2 *slideshow mode (automatically scales to fullscreen)*
  - 3 *grid (overview) mode*
  - 4 *print mode, 2 slides per page*
  - ← *slide left*
  - → *slide right*