DAD – Distributed Applications Development
Cristian Toma
D.I.C.E/D.E.I.C – Department of Economic Informatics & Cybernetics
www.dice.ase.ro
cristian.toma@ie.ase.ro
sections details
Agenda for DAD

1. Lectures Structure
2. Sections
3. Exchange Ideas
DAD Lectures Structure

DAD Administrative issues, Mission, Target Group Profile
1.1 DAD Lectures Structure

Main issues:

Didactic Activities: Lectures 50% + Lab / Seminar 50%

14 meetings 14 meetings

Evaluation: PC Exam – 60% / Seminars tests & assignments – 40%

E-Framework: VMs – VM-Ware Virtual Machines with:

- Linux Ubuntu 12 LTS + JDK 6 + Eclipse Indigo + Netbeans + Apache Tomcat + JBOSS + GCC
- Linux Ubuntu 8 + JDK 6 + Apache Hadoop 0.18
- Red Hat Linux + GCC + JDK6 + Globus Toolkit + Condor


Prerequisites: Fundamentals of Java 6 SE + C/C++ ‘11 + Networking + Linux/Windows OS | Optional – Ruby & Python

Mission: Technological transfer from university to the students of practical and theoretical issues related with distributed applications development.
1.2 Target Group Profile

DAD – Distributed Applications Development

DAD needs students with C/C++, C#, Java, Networking, OS Knowledge of Fundamentals
Sections – OOP, Networking, Web Dev, Core Middleware Dev, Distributed Solutions Dev

DAD Sections & References
It’s not just about the programming, but providing smart solutions

DAD Sections & References
What about the **DAD as it is @ Harvard/MIT?**

*Could you provide a solution for finding out the biggest mark in the class?*

Do we have unicast, multicast, broadcast messages or client-server, P2P / hybrid paradigms?...GREAT...Please upload the solution in Java / C# / C/C++ / Python / Ruby till next week 23:50 in e-learning platform – SAKAI...I’m NOT kidding...
2.1 DAD Sections

**Section I** – Summary of JSE for Distributed and Parallel Computing

**Section II** – Summary of Network Protocols Programming in JSE for Distributed Systems

**Section III** – Summary of Web Development in JEE

**Section IV** – Core Distributed MiddleWare Programming in JEE

**Section V** – Distributed Systems for Distributed & Parallel Computing
2.1 DAD Sections

I. Summary of JSE for Distributed and Parallel Computing

1. Java Annotations & Java Reflection
2. Java Generics & Java Collection Framework
4. Exceptions + Singleton classes & factory methods
5. JVM issues, processes & Multi-threading in JSE and C/C++ Linux vs. IPC and VM threading features
6. Parallelism Intro – Win7/Linux – Multi-threading, OpenMP, OpenMPI vs. OpenCL – Intel and nVidia providers or native nVidia CUDA

II. Summary of Network Protocols Programming in JSE for Distributed Systems

1. TCP – Transmission Control Protocol - RFC 793
2. UDP – User Datagram Protocol - RFC 768
3. HTTP – HyperText Transport Protocol - RFC 2616
7. RMI/RPC – Remote Method - RFC 1050 & RFC 1057 / RFC 2713
8. ARP/RARP, ICMP, LDAP, DNS, DHCP

III. Summary of Web Development in JEE

1. JNDI – Java Naming Directory Interface
2. XML DOM & SAX / JAXB vs. JSON – optional
3. JDBC – Java Database Connectivity – optional
4. Java Servlet – Intro
5. JSP – Java Server Pages & Taglibs
6. Java Beans & Mail
7. HTTP traffic analysis for JSP & Servlet
8. MVC: Spring vs Struts vs JSF – Optional
2.1 DAD Sections

IV. Core Distributed Middleware Programming in JEE

1. RMI – Remote Method Invocation
2. Middleware based on components/agents – CORBA
3. Web Services – SOAP – Simple Object Application Protocol, WSDL + SOA – Service Oriented Architectures vs. XML-RPC, REST services with JSON or XML-JAXB results
5. EJB – Enterprise Java Beans
6. HA – High-availability / Fail-over Clusters – JBOSS / Apache

V. Distributed Systems for Distributed & Parallel Computing – Case Studies

1. P2P Programming – JXTA - Java
2. Apache Hadoop - Java
3. Globus Toolkit
4. Condor + Globus Toolkit – Condor & Condor-G
6. ZeroC Ice – The Internet Communications Engine (Ice) is a modern distributed computing platform – combination of “CORBA style” with GRID - IceGrid
2.2 Mission & Goals Items for Basis using Java

- RMI, JNDI & CORBA
- Java Servlet & JSP
- Java Beans
- XML & Java Web Services
- JMS – java Messaging Service & EJB – Enterprise Java Beans

1. JSE Programming
- Java Annotations & Reflection
- Generics & Java Collection Framework
- I/O Stream & Java Libraries
- Singleton classes & Factory Methods
- JNI – Java Native Interface
- Java Multithreading & Linux C/C++ Threads Analogy
- Net Programming – Socket

2. Networking
- TCP & UDP
- SNMP
- SMTP, POP3, IMAP4
- FTP
- HTTP

3. Web Servers & Clusters Admin
- Apache Tomcat 7.0
- JBOSS

4. JEE Development
2.3 References


Section I – Summary of JSE for Distributed and Parallel Computing
1. [http://java.sun.com](http://java.sun.com)
4. [http://docs.oracle.com/javase/tutorial/](http://docs.oracle.com/javase/tutorial/)

Section II – Summary of Network Protocols Programming in JSE for Distributed Systems
1. [http://docs.oracle.com/javase/tutorial/networking/TOC.html](http://docs.oracle.com/javase/tutorial/networking/TOC.html)
2.3 References

Section III – Summary of Web Development in JEE

Section IV – Core Distributed Middleware Programming in JEE
1. http://docs.oracle.com/javase/tutorial/
RMI:
CORBA:
3. http://docs.oracle.com/javase/1.4.2/docs/guide/idl/GShome.html
XML & Web Services:
JMS & EJB:
2.3 References

Section V – Distributed Systems for Distributed & Parallel Computing
http://download.java.net/jxta/jxta-jxse/2.5/

High Performance Computing Training – HPC combined with HTC:
https://computing.llnl.gov/?set=training&page=index#training_materials
HTC = High Throughput Computing
HPC = High Performance Computing

Extra-ReadMe

HA Clusters & Cloud Computing – Ubuntu Cloud Infrastructure
http://en.wikipedia.org/wiki/High-availability_cluster
Hypervisor-KVM + OpenStack - IaaS Cloud:
https://help.ubuntu.com/community/UbuntuCloudInfrastructure
http://www.openstack.org/
http://www.openstack.org/software/start/

Embedded Distributed Systems
http://www.cl.cam.ac.uk/freshers/raspberrypi/tutorials/distributed-computing/ - demo in Python
JSE 25% + JEE 30% + C/C++ 15% + OS Admin 10% + Distributed Solutions 20%
Recommended Languages, OS & Technologies

OS & Virtualization

Programming Languages

interpreted Languages

HPC – High Performance Computing / Parallel Computing
Frameworks & Languages – C/C++ “flavors”
Recommended Languages & Technologies for HTC

**HTC – High Throughput Computing Frameworks based on C/C++/Java**

- **ZeroC ICE** – Java, C++, C# .NET
- **JXTA** – Java P2P
- **hadoop** – Java Map-Reduce
- **the globus toolkit** – Java & C/C++ for GRID
- **Condor** – C/C++ for GRID & HTC

**Internet Communications Engine + GRID**
Recommended Open IaaS Clouds

Open IaaS – Infrastructure as a Service Cloud


http://www.openstack.org
Section Conclusions

DAD – Distributed Applications Development

Technological Transfer from UNI2Student

Main Technologies

- IPC Linux + Multi-threading
- Java Standard Edition
- Java Enterprise Edition
- Core Distributed Middleware
  - RMI
  - CORBA
  - SOA – Web Services
  - JMS, EJB
- Distributed Systems for Parallel & Distributed Computing – Case Studies:
  - Apache Hadoop
  - Condor
Communicate & Exchange Ideas
Show and tell our **KNOWLEDGE**

Share and realize **IT&C Technological Transfer**

CREATE together **Distributed Application Development Entry-Level Support - AWARENESS!**
1. DAD - Is what you expected?

2. How many hours per week are you going to invest in order to achieve DAD goals?

3. How many of you are working in IT field – SW Dev., Admin., Designers?

4. What bachelor programs are you graduated from?

5. How many students get the payment scholarship from the companies vs. how many are/aren’t paying the studies?

6. In what disciplines did we collaborate together?
Thanks!