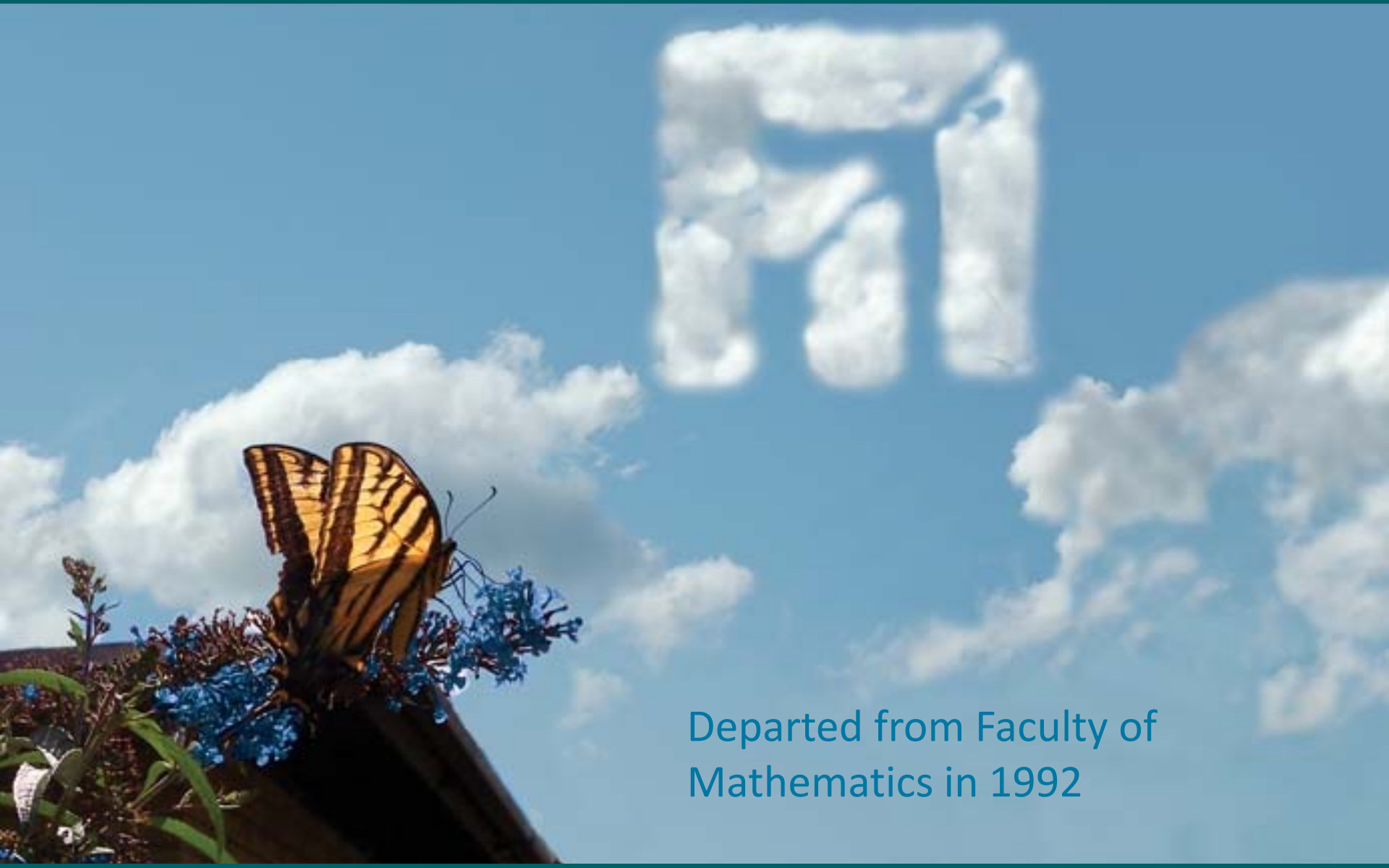
Service oriented computing at FII-UAIC

Dorel Lucanu

Alexandru Ioan Cuza University

Faculty of Computer Science



Departed from Faculty of Mathematics in 1992

www.info.uaic.ro

Research areas at FII-UAIC

- Algorithms and modern heuristics
- Bioinformatics
- Cryptography and information security
- Computational linguistics (NLP)
- Formal Methods in Software Engineering
- Models of computations
- Web technologies and Semantic Web

Service oriented NLP

group leader: **Dan Cristea**

dcristea@info.uaic.ro

NLP Web Services

- Tokenization and lemmatization
- Morpho-syntactic tagging
- Discourse parsing
- Anaphora resolution
- Summarization
- Semantic constituents detection
- Textual entailment
- Question answering

- soon available on FII servers
- all these are developed within CLARIN project

NLP WS Integration

- ALPE platform (Automated Linguistic Processing Environment)
  - complex chains of web services
  - local execution
  - distributed execution using GECC (General Environment for Cluster Computing)
- Similar platforms
  - WS at RACAI: http://www.racai.ro/webservices/
  - WebLicht: http://weblicht.sfs.uni-tuebingen.de/englisch/weblicht.shtml

Service oriented Web computing

group leader: **Sabin Buraga**

Projects 1/2

- **GRAI**: semantic description of Web services, using WSMO (Web Service Modeling Ontologies) and related approaches, in the context of Grid systems
- **TELEMON**: implementation and deployment of Web services using SOA (Service Oriented Architecture) paradigm, in the context of e-health applications
- **UpCity**, a prototype awarded with 1st prize at Imagine Cup 2009, Software Design section), e-government

Projects 2/2

- a XWiki-based Web site and a SharePoint solution for the intranet; several examples of such e-services are FII Admis, FII Licentiat, FII
- **hWiki** aims to use/define micro-formats for expressing certain information within a Wiki platform in order to semantically enhance the Web content
- **tuBiG** represents a Java-based layered infrastructure to create the proper setting for complex interactions between Grid components (e.g., agents or Web services)

Using of the MOP paradigm in SOA-based applications

group leader: **Dorel Lucanu**

Context

- a compositional modeling and validation framework for dynamically adaptable software-intensive systems, composed of distributed black-box components and services
  - formal models of evolvable architectures manifested as connectors, together with models of transformation to enact such a evolution
  - modeling, validation, and run-time monitoring of the QoS properties of the connectors as an integral part of the architecture of a system

MOP

- the three main aspects of MOP (Rosu et al)
  - observe a run of the system
  - check it against the desired properties, and
  - react/report if needed
- the monitors are automatically generated from specs
- MOP in SOA applications
  - allows adaptable executions that reacts to changes occurring during the execution of a complex system (e.g., some components become unavailable or change their predicted QoS), by reconfiguring the execution in order to ensure that the QoS remains optimal

Thanks!