

TECHNOLOGY TRANSFER PRESENTS

VLADIMIR BACVANSKI

**ARCHITECTURAL
THINKING**

**How to be
a good Architect**

MAY 22-24, 2017

CLOUD THINKING

**Designing and
Implementing
Systems for the Cloud**

MAY 25-26, 2017

RESIDENZA DI RIPETTA - VIA DI RIPETTA, 231
ROME (ITALY)



info@technologytransfer.it
www.technologytransfer.it

ABOUT THIS SEMINAR

The Architectural Thinking course teaches the skills and the mindset required to build robust, scalable and maintainable architectures. The course teaches a methodology and a set of techniques that show how to ensure that all aspects of a technical architecture are covered and documented.

We cover a set of principles, criteria, architectural structures and patterns essential in building software and enterprise architectures. The course includes the recent developments that cover scalability, cloud, Microservices, Big Data, and NoSQL impact on software architecture.

Numerous discussion and design exercises provide for engaging class.

AUDIENCE

- Developers
- Designers
- Architects
- Consultants
- Technical managers

PREREQUISITES

Experience with software development.

OBJECTIVES

- Upon completion, attendees will be able to:
- Recognize and understand the key qualities of good software architectures
- Be able to recognize, understand and use architectural patterns
- Effectively document architectures
- Be able to identify the various customers of IT architecture and work with architectural views
- Apply an effective methodology for evolving architectures
- Create architectures that withstand the challenges of modern applications
- Understand the impact of cloud, NoSQL, Microservices and container technologies

OUTLINE

1. Introduction

- What is Software Architecture?
- Architectural categories
- What makes a good architecture?
- Cohesion and coupling
- Software processes and the architecture business cycle
- Where do architectures come from?
- Typical architectural goals
- Enterprise Architecture

2. The Architectural Development Process

- A reference process
- Examples of popular processes
- Rational Unified Process (RUP)
- TOGAF Lifecycle
- Agile and Architecture

3. Architectural views

- The Stakeholders
- What are views?
- The Zachman Framework
- IEEE 1471-2000
- How to select views
- Commonly used views

4. A Notation for Describing Software Architecture

- Why notation?
- UML and beyond
- Other notations
- A pragmatic approach

5. Requirements and Architecture

- The requirements view
- Ensuring continuity in architecture
- Documentation of requirements
- Validation of architecture against functional requirements

6. Model Driven Engineering

- Model-Driven Engineering
- Productivity through automation
- Domain Specific Languages (DSL)

7. Architectural Assets

- What is an Asset?
- Reuse perspective
- Asset management

8. Quality Attributes: Modern Nonfunctional Requirements

- What are Quality Attributes?
- Attribute taxonomy
- Examples of Taxonomy
- Faults vs. failures
- Business qualities
- General scenario
- Example, Availability
- Architectural styles and patterns

9. Architectural Patterns

- Categories of patterns
- Layers
- Model View Controller Family of Patterns
- Publish subscribe
- Broker
- Blackboard
- Service Oriented Architecture
- Microservices

10. New Patterns for Scalability, NoSQL, Big Data, and Cloud

- Patterns for Scalability
- Sharding
- NoSQL Stores
- MapReduce and Hadoop
- The impact of Cloud
- Microservices and Cloud
- Dealing with Failures
- Containers and their Impact: docker and Related Systems

ABOUT THIS SEMINAR

Cloud Thinking is a fast paced, two-day course that introduces Architects and Developers to the new way of thinking required to build great Cloud applications.

We briefly introduce Cloud Computing, its benefits, challenges, and then focus on new technologies that make Cloud development different. The main objective is to teach design of applications for the Cloud as discrete, simple, well-formed units of work that scale well in the Cloud. We discuss just recently emerged Cloud specific design and architectural patterns. We explore the impact of Cloud on data stores and the use of NoSQL stores in Cloud applications.

We conclude with an overview of Docker, a container technology commonly used in Cloud applications.

AUDIENCE

- Developers
- Architects
- Data architects
- Technical managers
- Customer Quality Engineers

OBJECTIVES

Upon completion, attendees will be able to:

- Architect scalable applications on Cloud
- Select among design options for Cloud applications
- Design applications that consist of modular, well-formed units that scale well on Cloud
- Recognize and understand the value and challenges of Cloud Computing
- Be able to recognize, understand and use cloud architectures and patterns
- Understand containers and docker

1. A rapid intro to the Cloud

- What is Cloud and why do we care?
- Understanding Cloud
- How to move to the Cloud
- Cloud infrastructure

2. Cloud infrastructure impact

- Latency
- Collocation
- Resource pooling
- Elasticity
- Load balancing
- Cloud bursting
- Hypervisor clustering
- Redundant storage
- Content Delivery Network
- Multitenancy
- Multi-site deployment
- Service relocation
- Bare metal provisioning
- Logging
- Concurrency

3. Patterns for Cloud software development

- Architecting applications for the Cloud
- Writing well defined units of work that scale
- Services
- Service granularity
- Microservices
- Combating downsides of services
- Horizontal scaling
- Workflow distribution
- Dynamic and auto-scaling
- Busy signal
- Dealing with node failures
- Working with untrusted clients

4. Essential cloud security

- Threat agents
- Threats
- Defending against threats in the Cloud

5. Data in the Cloud

- Files storage in the Cloud
- Relational storage
- Sharding
- NoSQL stores
- CAP theorem and its effects
- ACID vs. BASE
- The impact of eventual consistency on users
- NoSQL case study: MongoDB
- NoSQL case study: Cassandra
- Big Data in the Cloud
- Hadoop in the Cloud
- Spark and the next generation of Big Data in the Cloud

6. Docker and the Cloud

- Containers and Docker
- Docker images
- Registries
- Networking
- Docker Compose
- Kubernetes
- Docker Datacenter: An Introduction
- Docker in the Cloud

INFORMATION

<p>PARTICIPATION FEE</p> <p>Architectural Thinking € 1700</p> <p>Cloud Thinking € 1300</p> <p>Special price for the delegates who attend both seminars: € 2800</p> <p>The fee includes all seminar documentation, luncheon and coffee breaks.</p> <p>VENUE</p> <p>Residenza di Ripetta Via di Ripetta, 231 Rome (Italy)</p>	<p>SEMINAR TIMETABLE</p> <p>9.30 am - 1.00 pm 2.00 pm - 5.00 pm</p> <p>HOW TO REGISTER</p> <p>You must send the registration form with the receipt of the payment to: TECHNOLOGY TRANSFER S.r.l. Piazza Cavour, 3 - 00193 Rome (Italy) Fax +39-06-6871102</p> <p>within May 8, 2017</p> <p>PAYMENT</p> <p>Wire transfer to: Technology Transfer S.r.l. Banca: Cariparma Agenzia 1 di Roma IBAN Code: IT 03 W 06230 03202 000057031348 BIC/SWIFT: CRPPIT2P546</p>	<p>GENERAL CONDITIONS</p> <p>DISCOUNT</p> <p>The participants who will register 30 days before the seminar are entitled to a 5% discount.</p> <p>If a company registers 5 participants to the same seminar, it will pay only for 4. Those who benefit of this discount are not entitled to other discounts for the same seminar.</p> <p>CANCELLATION POLICY</p> <p>A full refund is given for any cancellation received more than 15 days before the seminar starts. Cancellations less than 15 days prior the event are liable for 50% of the fee. Cancellations less than one week prior to the event date will be liable for the full fee.</p> <p>CANCELLATION LIABILITY</p> <p>In the case of cancellation of an event for any reason, Technology Transfer's liability is limited to the return of the registration fee only.</p>
--	--	---

VLADIMIR BACVANSKI

ARCHITECTURAL THINKING

Rome May 22-24, 2017
Residenza di Ripetta - Via di Ripetta, 231
Registration fee: € 1700

CLOUD THINKING

Rome May 25-26, 2017
Residenza di Ripetta - Via di Ripetta, 231
Registration fee: € 1300

BOTH SEMINARS

Special price for the delegates who attend both seminars: € 2800

If anyone registered is unable to attend, or in case of cancellation of the seminar, the general conditions mentioned before are applicable.

first name

surname

job title

organisation

address

postcode

city

country

telephone

fax

e-mail



Stamp and signature

Send your registration form with the receipt of the payment to:
Technology Transfer S.r.l.
Piazza Cavour, 3 - 00193 Rome (Italy)
Tel. +39-06-6832227 - Fax +39-06-6871102
info@technologytransfer.it
www.technologytransfer.it



SPEAKER

Software has always been **Dr. Vladimir Bacvanski** passion. Developing software faster, better, and scaling it to handle the largest data volumes arriving with great speed is his world. Realizing that great software requires great minds, Dr. Bacvanski also creates teams of exceptional developers and creators who can confidently take on the most challenging software problems. He is founder of SciSpike. His technical interests these days are centered around two main areas: architectures for high performance modular software and Big Data systems. This includes systems built with Node.js, Scala, Akka, commonly deployed in the Cloud. On the Big Data side these are Hadoop and its descendants such as Spark and Flink. He has over twenty years of experience with software technologies in areas such as architecture and design of mission critical and distributed enterprise systems, rule-based systems and languages, modeling tools, real-time systems, agent systems, and database technologies. Dr. Bacvanski also has extensive experience in software architecture and requirements analysis, and has helped many companies to select, transition, and apply new software technologies. He is published worldwide and is a frequent speaker, session chair, and workshop organizer at leading industry events. He is also the recipient of a number of prestigious academic scholarships and grants. Dr. Bacvanski is known for liking to explain how complex stuff works, and is a sought presenter and keynote speaker at leading industry events. O'Reilly recently published his course on the architecture of Big Data and NoSQL systems. For his contributions to the information management community, IBM awarded him the title of the IBM Information Management Champion for the sixth year running on 2015.

Mr. Bacvanski received a Doctoral degree in Computer Science from Aachen University of Technology (RWTH Aachen) in Germany.