IMPORTANT DATES
Deadline for papers submission: July 17, 2015
Notification of acceptance: August 21, 2015
Online pre-proceedings: September 19, 2015
Workshop dates: September 28, 2015

ORGANIZATION
Workshop organizers and chairs
Federico Ciccaci (contact), Mälardalen University
Etienne Borde, Telecom ParisTech
Patrizio Pelliccione, Univ. Gothenburg| Chalmers

Program Committee
Marco Autili, Italy
Steffen Becker, Germany
Jan Carlson, Sweden
Antonio Cicchetti, Sweden
Diarmuid Corcoran, Sweden
Ivica Crnkovic, Sweden
Guglielmo De Angelis, Italy
David Garlan, USA
Sébastien Gerard, France
Jeff Gray, USA
Lucia Happe, Germany
Anne Koziolek, Germany
Patricia López Martínez, Spain
Julio Medina, Spain
Raffaela Mirandola, Italy
Marco Panunzio, France
Alfonso Pierantonio, Italy
Pascal Poizat, France
Ansgar Radermacher, France
Mehrad Saadatmand, Sweden
Lionel Seinturier, France
Séverine Sentilles, Sweden
Massimo Tivoli, Italy
Tullio Vardanega, Italy

SCOPE
Model-Driven Engineering (MDE) and Component-Based Software Engineering (CBSE) have been shown to effectively reduce software development complexity by (i) shifting the focus from source code to models and (ii) building software systems as composition of new and existing components, respectively. Moreover, the interplay of MDE and CBSE approaches is gaining recognition as a very promising means to boost the development of software systems by reducing costs and risks and shorten time-to-market. While several attempts to effectively combine MDE and CBSE have been documented, there are still unsolved clashes arising when exploiting interplay of MDE and CBSE, mostly due to mismatches in the related terminology as well as to differences in their basic essence. As satellite event of MODELS’15, the goal of ModComp’15 is to gather researchers and practitioners to share opinions, propose solutions to open challenges and generally explore the frontiers of intertwining between MDE and CBSE.

TOPICS
Solicited topics include, but are not limited to:
- Partial model reuse: once individual components are modelled (interfaces and behavior), these models should be reusable in the different usage contexts of these components;
- Model composition: building a system model by composition of pre-existing models of individual components;
- Model versioning: in order to handle model evolution when for instance one component is upgraded to a newer version;
- Modelling component interaction and component behaviours: clear separation of internal behaviour and externally visible interaction capabilities, e.g. by interface protocols;
- Model extraction for componentization of legacy systems: when legacy systems are componentized, generation of architectural and behavioural models from, e.g., implementation artefacts is needed in order to get full support from model-based activities such as analysis, e.g. if those components are reused in a new context. Along with implementation artefacts, other kinds of information regarding any observation of the system at runtime, such as, e.g., log files, system execution traces, might need to be considered for reverse componentization;
- Component interoperability: in order to enable the automated construction of semantic matching and mapping between different modelling notations (e.g., component models) with emphasis on precise syntactic, protocol and operational descriptions of components;
- Management and elicitation of model interdependencies: to infer and support automated reasoning on the possible interdependencies between the different software models exploited throughout the software life cycle;
- Model (co)evolution: tackling challenges in metamodels (e.g., component models) evolution and model co-evolution which are amplified by the high degree of interchangeability typical of CBSE;
- Model transformations for analysis and code generation in presence of third-party components: exploring how model-driven techniques may deal with third-party components, especially concerning the preservation of system properties (both functional and extra-functional) along the involved model manipulations;
- Metamodel modularity: reasoning on issues related to composability of (i) metamodels and (ii) views in terms of metamodel portions;
- Composition of MDE artefacts: analysis results, model transformations, and/or model viewpoints could take advantage of CBSE advancements in order to ease their reuse, and composition;
- Enforcement of incrementality: models and model manipulations to support incremental verification and validation of component-based systems;
- Case studies & applications: best practices applied to real world applications, lessons learned, success/failure stories in intertwining MDE and CBSE.

WORKSHOP FORMAT AND PAPERS SUBMISSION
The workshop is organized as a full-day event with papers presentation and group discussions.
ModComp’15 welcomes research papers, experience papers and tool presentations.
Contributions can be:
- Research papers (maximum 6 pages, including figures, appendices AND references)
- Experience reports (maximum 6 pages, including figures, appendices AND references)
- Position papers (maximum 2 pages, including figures, appendices AND references)

Contributions should represent original and previously unpublished ideas that are currently not under review in any conference or journal; the workshop’s language is English.
Each submitted paper undergoes a formal peer review process by a minimum of 3 Program Committee members. Submitted papers should include authors’ names, affiliations and contact information.
Contributions should be uploaded electronically via EasyChair according to the instructions on the workshop’s website. Papers must be in the IEEEtran format (link to templates on the workshop’s website).
Accepted papers will be published in a pre-conference edition of CEUR, which is indexed by DBLP.

At least one author of each accepted paper must register and attend the workshop in order for the paper to be published.

SPECIAL ISSUE OF JOURNAL OF OBJECT TECHNOLOGY (JOT)
The best papers from the workshop will be invited for submission in extended form to a special issue of the Journal of Object Technology (JOT). The extended papers will undergo a new reviewing process.