

| | |
|-------------------------|---|
| 1. Record Nr. | UNINA9910485008703321 |
| Titolo | Architectures for Adaptive Software Systems : 5th International Conference on the Quality of Software Architectures, QoSA 2009, East Stroudsburg, PA, USA, June 24-26, 2009 Proceedings // edited by Raffaella Mirandola, Ian Gorton, Christine Hofmeister |
| Pubbl/distr/stampa | Berlin, Heidelberg : , : Springer Berlin Heidelberg : , : Imprint : Springer, , 2009 |
| ISBN | 3-642-02351-7 |
| Edizione | [1st ed. 2009.] |
| Descrizione fisica | 1 online resource (X, 213 p.) |
| Collana | Programming and Software Engineering, , 2945-9168 ; ; 5581 |
| Altri autori (Persone) | GortonIan HofmeisterChristine MirandolaRaffaella |
| Disciplina | 005.1 |
| Soggetti | Software engineering Compilers (Computer programs) Computer programming Computer science Algorithms Software Engineering Compilers and Interpreters Programming Techniques Computer Science Logic and Foundations of Programming |
| Lingua di pubblicazione | Inglese |
| Formato | Materiale a stampa |
| Livello bibliografico | Monografia |
| Note generali | Bibliographic Level Mode of Issuance: Monograph |
| Nota di bibliografia | Includes bibliographical references and index. |
| Nota di contenuto | Model-Driven Quality Analysis -- A Model-Based Framework to Design and Debug Safe Component-Based Autonomic Systems -- Applying Model Transformations to Optimizing Real-Time QoS Configurations in DRE Systems -- Automated Architecture Consistency Checking for Model Driven Software Development -- Architectural Performance Prediction -- Improved Feedback for Architectural Performance Prediction Using Software Cartography Visualizations -- Predicting Performance Properties for Open Systems with KAMI -- Compositional Prediction of Timed Behaviour for Process Control Architecture -- Timed Simulation of Extended AADL-Based Architecture Specifications |

with Timed Abstract State Machines -- Architectural Knowledge -- Achieving Agility through Architecture Visibility -- Successful Architectural Knowledge Sharing: Beware of Emotions -- Toward a Catalogue of Architectural Bad Smells -- Case Studies and Experience Reports -- On the Consolidation of Data-Centers with Performance Constraints -- Evolving Industrial Software Architectures into a Software Product Line: A Case Study -- Adaptive Application Composition in Quantum Chemistry.

Sommario/riassunto

Much of a software architect's life is spent designing software systems to meet a set of quality requirements. General software quality attributes include scalability, security, performance or reliability. Quality attribute requirements are part of an application's non-functional requirements, which capture the many facets of how the functional requirements of an application are achieved. Understanding, modeling and continually evaluating quality attributes throughout a project lifecycle are all complex engineering tasks which continue to challenge the software engineering scientific community. While we search for improved approaches, methods, formalisms and tools that are usable in practice and can scale to large systems, the complexity of the applications that the software industry is challenged to build is ever increasing. Thus, as a research community, there is little opportunity for us to rest on our laurels, as our innovations that address new aspects of system complexity must be deployed and validated. To this end the 5th International Conference on the Quality of Software Architectures (QoSA) 2009 focused on architectures for adaptive software systems. Modern software systems must often reconfigure their structure and behavior to respond to continuous changes in requirements and in their execution environment. In these settings, quality models are helpful at an architectural level to guide systematic model-driven software development strategies by evaluating the impact of competing architectural choices.
