1. Record Nr. UNIORUON00082115

Autore	HABNAKA, Abd al-Rahman
Titolo	al-Taqafah al-islamiyyah : al-Mustawa al-awwal (101) / Abd al-Rahman Habnaka, Muhammad al-Gazzali ; Muragi ah Muhammad Ibrahim Ali, Husayn Hamid Hassan
Pubbl/distr/stampa	Gedda, : Gami at al-malik Abd al- Aziz, 1994
Descrizione fisica	268 p. ; 24 cm
Altri autori (Persone)	al-GAZZALI, Muhammad
Disciplina	297
Soggetti	Islam - Arabia Saudita
Lingua di pubblicazione	Arabo
Formato	Materiale a stampa
Livello bibliografico	Monografia

2. Record Nr. UNINA9910483380903321

Titolo	Quality of Software Architectures : Second International Conference on Quality of Software Architectures, QoSA 2006, Västeras, Schweden, June 27-29, 2006, Revised Papers / / edited by Christine Hofmeister, Ivica Crnkovic, Ralf H. Reussner
Pubbl/distr/stampa	Berlin, Heidelberg : , : Springer Berlin Heidelberg : , : Imprint : Springer, , 2006
ISBN	3-540-48820-0
Edizione	[1st ed. 2006.]
Descrizione fisica	1 online resource (X, 218 p.)
Collana	Programming and Software Engineering, , 2945-9168 ; ; 4214
Altri autori (Persone)	HofmeisterChristine CrnkovicIvica ReussnerRalf
Disciplina	004.2/2
Soggetti	Software engineering Electronic digital computers - Evaluation Computer science Computer networks Computers and civilization Software Engineering System Performance and Evaluation Computer Science Logic and Foundations of Programming Computer Communication Networks

	Computers and Society
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	Abstracts of the Keynotes -- Expanding the Scope of Software Product Families: Problems and Alternative Approaches -- Composing with Style – Components Meet Architecture -- Abstracts of the Tutorials -- Documentation Principles and Practices That You Can Live with -- Model-Based Software Development with Eclipse -- Software Architecture Analysis and Evaluation -- Architecture Evaluation: Selecting Alternatives -- MEMS: A Method for Evaluating Middleware Architectures -- Evaluating Alternative COTS Assemblies from Imperfect Component Information -- Managing and Applying Architectural Knowledge -- Building Up and Reasoning About Architectural Knowledge -- Managing Architectural Design Decisions for Safety-Critical Software Systems -- Architectural Evaluation: Performance Prediction -- Runtime Prediction of Queued Behaviour -- Model Transformation in Software Performance Engineering -- Processes for Supporting Architecture Quality -- Traveling Architects – A New Way of Herding Cats -- A Practical Architecture-Centric Analysis Process -- Models for Architecture Evaluation -- Embedded Systems Architecture: Evaluation and Analysis -- Parameter Dependent Performance Specifications of Software Components -- Architectural Evaluation -- Applying the ATAM to an Architecture for Decentralized Control of a Transportation System -- Towards an Integration of Standard Component-Based Safety Evaluation Techniques with SaveCCM.
Sommario/riassunto	Although the quality of a system’s software architecture is one of the critical factors in its overall quality, the architecture is simply a means to an end, the end being the implemented system. Thus the ultimate measure of the quality of the software architecture lies in the implemented system, in how well it satis?es the system and project requirements and constraints and whether it can be maintained and evolved successfully. In order to treat design as a science rather thananart,weneedtobeabletoaddressthequalityofthesoftwarearchitecture directly, not simply as it is re?ected in the implemented system. Therefore, QoSA is concerned with software architecture quality directly by addressing the problems of: – Designing software architectures of good quality – De?ning, measuring, evaluating architecture quality – Managing architecture quality, tying it upstream to requirements and do- stream to implementation, and preserving architecture quality throughout the lifetime of the system Cross-cutting these problems is the question of the nature of software archit- ture. Software architecture organizes a system, partitioning it into elements and de? ning relationships among the elements. For this we often use multiple views, each with a di?erent organizing principle.