

1. Record Nr.	UNINA9910300155903321
Autore	Mauro Jacopo
Titolo	Constraints Meet Concurrency // by Jacopo Mauro
Pubbl/distr/stampa	Paris : , : Atlantis Press : , : Imprint : Atlantis Press, , 2014
ISBN	94-6239-067-3
Edizione	[1st ed. 2014.]
Descrizione fisica	1 online resource (151 pages) : illustrations
Collana	Atlantis Studies in Computing, , 2212-8565 ; ; 5
Disciplina	005.11
Soggetti	Compilers (Computer programs) Machine theory Compilers and Interpreters Formal Languages and Automata Theory
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 at the end of each chapters.
Nota di contenuto	Introduction -- Constraints -- Concurrency -- Constraint Handling Rules -- Non Turing powerful fragments of CHR -- Expressive power of priorities in CHR -- Constraints in Clouds -- A Classification-based Approach to Manage a Solver Portfolio -- Broadcast messages in Jolie -- Interruptible request responses in Jolie -- Conclusions -- References -- Proofs.
Sommario/riassunto	This book describes the benefits that emerge when the fields of constraint programming and concurrency meet. On the one hand, constraints can be used in concurrency theory to increase the conciseness and the expressive power of concurrent languages from a pragmatic point of view. On the other hand, problems modeled by using constraints can be solved faster and more efficiently using a concurrent system. Both directions are explored providing two separate lines of development. Firstly the expressive power of a concurrent language is studied, namely Constraint Handling Rules, that supports constraints as a primitive construct. The features of this language which make it Turing powerful are shown. Then a framework is proposed to solve constraint problems that is intended to be deployed on a concurrent system. For the development of this framework the concurrent language Jolie following the Service Oriented paradigm is used. Based on this experience, an extension to Service Oriented

Languages is also proposed in order to overcome some of their limitations and to improve the development of concurrent applications.

---