

SpringerBriefs in Computer Science

Series Editors

Stan Zdonik
Peng Ning
Shashi Shekhar
Jonathan Katz
Xindong Wu
Lakhmi C. Jain
David Padua
Xuemin Shen
Borko Furht
V. S. Subrahmanian
Martial Hebert
Katsushi Ikeuchi
Bruno Siciliano

For further volumes:
<http://www.springer.com/series/10028>

Clarissa Cassales Marquezan
Lisandro Zambenedetti Granville

Self-* and P2P for Network Management

Design Principles and Case Studies

Clarissa Cassales Marquezan
Paluno
University of Duisburg-Essen
Essen
Germany

Lisandro Zambenedetti Granville
Institute of Informatics
Federal University of Rio Grande do Sul
Porto Alegre, RS
Brazil

ISSN 2191-5768
ISBN 978-1-4471-4200-3
DOI 10.1007/978-1-4471-4201-0
Springer London Heidelberg New York Dordrecht

ISSN 2191-5776 (electronic)
ISBN 978-1-4471-4201-0 (eBook)

Library of Congress Control Number: 2012940225

© The Author(s) 2012

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed. Exempted from this legal reservation are brief excerpts in connection with reviews or scholarly analysis or material supplied specifically for the purpose of being entered and executed on a computer system, for exclusive use by the purchaser of the work. Duplication of this publication or parts thereof is permitted only under the provisions of the Copyright Law of the Publisher's location, in its current version, and permission for use must always be obtained from Springer. Permissions for use may be obtained through RightsLink at the Copyright Clearance Center. Violations are liable to prosecution under the respective Copyright Law.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

While the advice and information in this book are believed to be true and accurate at the date of publication, neither the authors nor the editors nor the publisher can accept any legal responsibility for any errors or omissions that may be made. The publisher makes no warranty, express or implied, with respect to the material contained herein.

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

To our families and friends

Foreword

It is commonly accepted that the main obstacle currently facing the IT industry is a looming software complexity crisis. The need to integrate several heterogeneous environments to one or more computing systems, and beyond a single company's boundaries, introduces new levels of complexity. The management of communications systems is in an even worse situation; a highly diverse set of hardware and software must interact in even more heterogeneous environments. Communication systems are becoming massive and complex enough to render almost impossible the job of installing, configuring, optimizing, maintaining, and merging them in a timely manner. Is it not crazy to say that not far in the future it will be unfeasible to make decisive responses to demands, failures, or changes on time.

Under these assumptions Self-managed or self-* systems have emerged as a promising solution embraced not only by the academy (self-*, distributed systems are a lot of fun) but also by the industry (remember IBM's Autonomic Computing). Self-* embeds the complexity into the infrastructure and promises to manage it with little or no need for recognition or effort on the part of the user. Users would then be allowed to focus on their tasks, and not on the means of accomplishing them, which results in greater productivity and enhanced work experience. It is said that self-* entities will cooperate to maintain and adjust their operation in front of changing topologies, components, workloads, demands, and external conditions, even in the event of hardware or software failures.

One of the key words in the paragraph above is cooperation and, more often than not, cooperation in application-layer systems lately means a peer-to-peer organization. Despite the fact that the classic network management systems were and still are eminently centralized, self-management and autonomic communications researchers understood very soon that peer-to-peer techniques were the natural manner of organizing systems based on the idea of autonomous agents that have to self-organize, share information, and make coordinated decisions.

It is not common to find researchers with a knowledge and a view so broad that they are able to work at the same time, well, in the same research work at least, from two opposite ends of the complex and rich problem of automating the very

human task of managing a network. Clarissa and Lisandro, the authors of the work presented in this book do that, and outstandingly well.

In one chapter, this book will introduce you to the world of self-* systems; in another it will explain much of what you need to know about peer-to-peer techniques applied to network management, but what is more interesting is that the authors use all this knowledge in a test-case driven investigation methodology that impedes them to avoid issues and imposes rigorousness to their work.

Montevideo, March 2012

Javier Baliosian

Preface

The need for sophisticated network management solutions have become more and more important due to the complexity of the managed environment and the expectations of the end users. The employment of self-* properties or autonomic computing and Peer-to-Peer (P2P) for network management became promising alternatives to provide such sophisticated solutions. We observed that in general the proposed autonomic or self-* solutions for network management were designed in a very centralized fashion and focused on abstract issues of management. In contrast, P2P was being used in network management as the underlying infrastructure to enhance the connectivity capabilities of network managers, whose designs follow the traditional network management approaches, such as Management by Delegation.

Interestingly, these two trends were developed in parallel to each other, without (or with minor) direct connection between them. This situation caught our attention and we started to wonder how we could combine these two trends in order to enable a more sophisticated and truly distributed design of network management solutions able to deal with the complexity of current management. Our wonders gave space for the proposition of the Self-* P2P design for network management solutions.

Specifically, this book describes the insights, definitions, and guidelines to support the readers to understand how one should think and what one should consider when designing a Self-*P2P solution. In an incremental fashion, we move from abstract concepts into the tangible application of those concepts. In this way, the readers are presented with a description of all the steps to design a Self-* P2P solution. We start with the abstract concepts needed to change the mind-set for developing new self-managed, fully distributed, and cooperative network management solutions. In the next step, we exemplify how one should identify scenarios that can benefit from a Self-* P2P solution. Then, the readers are presented with a detailed description of self-* P2P solutions for two case studies considering distinct scenarios.

The focus of this book is not the experimental part of the developed solutions. Instead, we focus on the description of the conception phase of each solution.

Especially, we dedicate two sections and one entire chapter to analyzing the design decisions which were taken during the development of the case studies. By doing this, we expect to help the readers to deeply understand the changes in the way of thinking network management solutions by exploring the potentials of using distributed, parallel, and concurrent models which are combined in the Self-* P2P design. Particularly, readers involved with the development of dynamic, heterogeneous, large-scale network management solutions are the ones that could benefit the most from the insights and solutions described in this book.

The ideas and principles discussed in this book are in fact the result of more than 4 years of investigation. During this time, different people contributed directly and indirectly to the accomplishments described in this book. Therefore, we would like to thank all of you for the support and contributions.

Dortmund, Germany, March 2012
Porto Alegre, Brazil, March 2012

Clarissa Cassales Marquezan
Lisandro Zambenedetti Granville

Contents

| | |
|--|----|
| 1 Introduction | 1 |
| 1.1 Latest Network Management Trends | 1 |
| 1.2 Investigating a New Alternative | 3 |
| References | 4 |
| 2 State of the Art | 5 |
| 2.1 Background | 5 |
| 2.1.1 Network Management Approaches | 6 |
| 2.1.2 Autonomic Computing and Self-* Properties | 8 |
| 2.1.3 Peer-to-Peer | 10 |
| 2.2 Autonomic Computing and Network Management | 11 |
| 2.3 Employment of P2P on Network Management | 13 |
| 2.4 Autonomic/Self-*, Peer-to-Peer, and Network Management | 15 |
| 2.5 Summary | 17 |
| References | 18 |
| 3 Principles of the Self-* P2P Design | 27 |
| 3.1 Leading Conditions and Fundamental Questions Towards the Self-* P2P Alternative | 27 |
| 3.2 Characterization of Networks and Management Requirements | 29 |
| 3.3 Definition and Delimitation of Terms and Concepts | 31 |
| 3.4 Integration Requirements for Designing the Self-* P2P Solutions | 34 |
| 3.5 Relationship of Integration Requirements and Attributes of Concurrent Models | 35 |
| 3.6 Selection of the Case Studies | 38 |
| 3.7 Summary | 40 |
| References | 41 |

- 4 Case Study I: Reliability of Monitoring Platforms** 43
 - 4.1 Self-Healing P2P-Based Approach 43
 - 4.1.1 Supported Types of Failures 44
 - 4.1.2 Architecture and Concepts 45
 - 4.1.3 Failure Detection 48
 - 4.1.4 Service Instance Activation and Policies 49
 - 4.2 Development of the Case Study 52
 - 4.2.1 NAC Monitoring System 52
 - 4.2.2 Extending ManP2P Platform 53
 - 4.2.3 Implementation 55
 - 4.3 Experimental Evaluation 58
 - 4.3.1 Measurement Process 59
 - 4.3.2 Summary of Experimental Results 60
 - 4.4 Discussion About Designed Approach 61
 - 4.4.1 Compliance to Management Requirements 61
 - 4.4.2 Achievement of Integration Requirements 63
 - 4.4.3 Potentialities and Shortcomings 65
 - 4.5 Final Remarks on the Case Study 66
 - 4.6 Summary 67
 - References 68

- 5 Case Study II: Resource Management of Network Virtualization** 69
 - 5.1 Self-Organizing P2P Approach 69
 - 5.2 Network Virtualization Model 72
 - 5.3 Development of the Case Study 75
 - 5.3.1 Self-Organizing Control Loop 76
 - 5.3.2 Receiving Candidate Heuristic 80
 - 5.3.3 Moving Candidate Heuristic 81
 - 5.3.4 Implementation 81
 - 5.4 Experimental Evaluation 84
 - 5.4.1 Testbed 85
 - 5.4.2 Summary of Simulation Results 86
 - 5.5 Discussion About Designed Approach 86
 - 5.5.1 Compliance to Management Requirements 86
 - 5.5.2 Achievement of Integration Requirements 88
 - 5.5.3 Potentialities and Shortcomings 89
 - 5.6 Final Remarks on the Case Study 90
 - 5.7 Summary 91
 - References 92

- 6 Results Discussion** 93
 - 6.1 Analyzing the Design of the Integration Requirements 93
 - 6.2 Delineating Dimensions of the Self-* P2P Approach 95

| | |
|---|------------|
| Contents | xiii |
| 6.3 Identifying Self-* P2P Dimensions in the Case Studies | 97 |
| 6.4 Answering Fundamental Questions | 100 |
| 6.5 Summary | 101 |
| 7 Conclusions | 103 |