Preface

The book “Agile and Lean Service-Oriented Development: Foundations, Theory, and Practice” is a collection of original research studies and experience reports covering a cross-section of software engineering issues: firstly, agile and lean software development methods and secondly, service oriented computing. Typical topics in this cross-sectional area seek to exploit the synergy that can be generated from combining knowledge from both fields. These topics include organizational adoption of agile/lean software development for service oriented development, test-driven service oriented systems, service oriented architectures driven by agile/lean principles, evolving requirements in service oriented systems, refactoring of service oriented systems, migration to service oriented systems using agile approaches, and agile/lean cloud computing. The book offers insights into the issues, challenges, patterns, and solutions for agile and lean service-oriented development, on the basis of rigorous scientific research, sound empirical evidence, and industrial experiences and lessons learnt. It will be of broad interest to agile and service research communities and practitioners alike.

WHEN AGILE AND LEAN METHODS MEET SERVICE-ORIENTATION

Modern software development projects are enacted in increasingly turbulent business environments typified by unpredictable markets, changing customer requirements, pressures of ever shorter time-to-deliver, and rapidly advancing information technologies. Two notable approaches developed in recent years - service oriented computing and agile and lean software development - present promising approaches for contemporary software development projects to deal effectively with these challenges.

Service Oriented Engineering (SOE) is an emerging software development paradigm that copes with today’s organization requirements, such as shorter time to market, continuous change of business partners, integration of information systems within the organization and across organisational boundaries, distributed business and computing, and agility. Service oriented computing standards, architectures, principles and technologies are used to build applications that are currently found on the web, or mobile devices. New programming models have emerged based on service oriented principles such as cloud computing. The development of information systems based on service oriented computing suggests changes to traditional software engineering. It holds the promise of open, distributed, dynamic and adaptable systems that provide business solutions based on concepts such as loose coupling, integration and reusability.

Agile methods are a family of software development methods promoted through the Agile Manifesto, which specifies a set of agile values and principles. Agility itself is defined as “the ability to create and respond to change.” Communities have formed around specific development methods, such as Scrum
and eXtreme Programming (XP). Lean software development is claimed as “the next wave of software process.” An increasing number of software organisations and IT departments have been adopting or are planning to adopt lean thinking and lean techniques. The agile and lean software development movement challenges the values that were held dear in conventional development methods, such as detailed upfront plans, precise prediction and rigid control strategies, preferring to emphasize the frequent delivery of working software, collaboration and communication, and the importance of people over process.

Research and practice in SOE and agile and lean methods areas have been advanced independently over the past several years. It is intriguing to explore the potential synergy that comes from a combination of both areas. For example, some may argue that SOE is based on design and architecture principles that cannot be matched with agile software development. However, as distinct as they seem to be, SOE and agile methods may have a lot in common and can inform each other. For example, agility is one of the key attributes of service oriented computing, and agile research and practice can shed light on how to achieve agility in service oriented development. Vice versa, research and practice conducted on service oriented architecture may provide valuable guidelines for agile teams to build truly agile systems that are easy to modify or extend to accommodate frequent change requests. This book is one of the first major attempts to explore this intriguing and promising cross-sectional area.

OBJECTIVES

The mission of the book is to provide a roadmap for advancing the research and practice of software engineering through the potential provided by the combination of service oriented engineering and agile and lean methods. The overall objectives of this book are: (1) to explore the foundations on which service oriented engineering and agile and lean software development can be combined; (2) to build the conceptual basis and empirical evidences for the combination of the two approaches; and (3) to provide tools, best practices and guidelines for agile and lean service oriented development in practice.

CONTRIBUTION

There is an increasing interest in investigating the values and venues of combing service orientated computing and agile and lean methods. However, very few publications have addressed this potential, and even less publications have done it in a comprehensive and systematic manner. Most of the available sources that leverage these two areas are anecdotal in nature, e.g., personal blogs. We believe that our book is a valuable contribution to the body of knowledge of both service oriented engineering and agile and lean development. It can help in establishing foundations for the combination of the two and point out directions for future software engineering research and practice.

The book advances theory by considering the intersection of agile and lean development methods and service oriented engineering. In particular it contributes to a better understanding of how adopting agile software development techniques, methodologies and practices can affect service oriented engineering and organizations that are migrating their systems into service based ones or developing them.

It also constitutes a contribution to practice. Senior IT managers who are responsible for application development and the IT infrastructure need to be aware of the potential for synergy and conflict of agile/lean and SOE. Project managers who are involved in agile development in a context of service oriented
architecture will find frameworks and lessons from practice that will be valuable in understanding the architectural implications of the closer relationship between development methods and IT infrastructure. Software developers will also find this book helpful; those with an understanding of agile and SOE and how they can work together to achieve more effective business outcomes will likely be in great demand in the future.

AUDIENCE

The intended audience of the book is first of all researchers from either software engineering or information systems disciplines whose research interests lie in agile and lean methods, service computing/service science, or both. The book will provide them with a roadmap for researching on these topics and equip them with necessary theoretical devices to conduct their studies. Secondly, software development practitioners who want to explore the potential of both agile and lean methods and service oriented computing may also find this book useful. The book will provide them with informed understanding and solutions of how to get the best out of the two worlds. Lastly, the book can also be useful for university students who specialise in software engineering or information systems. It will help them to obtain up-to-date and in-depth knowledge on agile and lean methods, service computing/service science and the overlap of the two areas.

CONTENT

The chapters of the book are divided into three sections.

Section 1: Foundations

This section has two invited introduction chapters that lay the foundation for the book and for the audience. The first chapter mainly focuses on agile and lean software development, examining the history of the agile movement as well as providing basic concepts. It touches briefly on service orientation. The second chapter takes service orientation as its primary subject, introducing relevant foundations concerning evolution and adaptation of service-oriented systems. It concludes by highlighting the need for agility in both adaptation and evolution of service-oriented systems.

Section 2: Theory

This is the main and largest section and contains ten chapters based on original research work. The topics of these chapters cover diverse, yet inter-related topics, which can be grouped into the following themes: (1) service oriented software processes (Chapter 3, 4, and 5), focusing on how agile practices are used to build service based applications; (2) agile software development and service compositions (Chapter 6, 7, and 8), describing how agile software development techniques enable service compositions and vice versa; (3) security architectural styles and agile development (Chapter 9 and 10), presenting how agile and lean approaches can benefit software architectures of secure service based applications; and
(4) testing services (Chapter 11 and 12), discussing how agile practices can be applied in service testing. The research methods used to produce the scientific results reported in these chapters range from systematic literature review, case study, and experiment to design research. Together they present the state-of-the-art of research in the cross-sectional area of agile and lean service development.

**Section 3: Practice**

As the section title suggests, the two chapters in this section focus more on real world experiences of agile and lean service oriented development. While Chapter 13 is a detailed and vivid recount of the agile and lean service development journey of a software development team, Chapter 14 adopts a more methodological stance reporting the experiences of test-driven decomposition of legacy systems into services. Both chapters provide valuable lessons learnt from the trenches in an easily digestible manner for practitioners.

In summary, the chapters contained in the book collectively provide a comprehensive understanding of the agile and lean service development area from different perspectives. They not only represent the knowledge of today, but also highlight the significant and promising topics that need further scientific and practical endeavours from both research communities and software practitioners.

*Xiaofeng Wang*
Free University of Bozen/Bolzano, Italy

*Nour Ali*
Lero, the Irish Software Engineering Research Centre, Ireland

*Isidro Ramos*
Universidad Politécnica de Valencia, Spain

*Richard Vidgen*
Hull University Business School, UK