

Preface

A distributed system is a collection of autonomous units able to make decisions locally. Through cooperation, these distributed units can solve global computational problems together. This book is dedicated to one of the most fundamental processes in operating a distributed system, referred to as the *rendezvous* process. Rendezvous takes place when the external ports belonging to two neighboring units become connected, and they can communicate and exchange information through this connection.

There are five dimensions in which to define an instance of rendezvous. First of all, the autonomous units can run an algorithm that is *symmetric* or one that is *asymmetric*. In some distributed systems, the autonomous units all play different roles and so they may run different algorithms, which are also called asymmetric algorithms, while in other systems, all units are of the same type and run the same symmetric algorithm. Second, the distributed units may or may not start at the same time, which correspond to the *synchronous* or *asynchronous* scenario, respectively. Third, some of the external ports in a unit may be occupied by services unrelated to rendezvous, and so different units may have different sets of available ports. It is the *symmetric port setting* if all units have the same set of available ports, otherwise *asymmetric port setting*. Fourth, the distributed units are *anonymous* if they appear to be indistinguishable, or *non-anonymous* if they can be distinguished by their labels or identifiers. Labeling applies to external ports also. In some distributed systems, the external ports have the same labels across all units, which is referred to as *non-oblivious port labeling*. These global labels of the ports simplify the rendezvous problem as one can easily capitalize on these labels in designing the rendezvous solution. Alternatively, *oblivious port labeling* assumes there is no global labeling rule, and the units label their external ports locally. Obviously, rendezvous is harder to achieve when using the latter labeling which however is more practical. In this book, we present rendezvous algorithms for all the combinations of these five dimensions.

This book is divided into five parts, which are further broken into 20 chapters. We start with an introduction of distributed rendezvous theory, which includes distributed system preliminaries, distributed computing, and rendezvous theory.

Next, we present different kinds of rendezvous algorithms for the blind rendezvous problem in Part II, where the autonomous units' ports have the same labels, i.e., non-oblivious port labeling. Then, in Part III, we introduce the oblivious blind rendezvous problem where the ports are labeled locally by the units and we present distributed rendezvous algorithms for a range of rendezvous settings. In Part IV, we introduce several rendezvous applications and discuss the method of extending the rendezvous algorithms for distributed systems to these applications. Finally, we summarize the rendezvous results and mention some future work in Part V.

This book can be treated as a handbook of solutions to the rendezvous problem in distributed systems. Rendezvous as a fundamental process underpins the construction of many important functions in distributed systems and networks. Other than theories and algorithms, this book also covers applications in which rendezvous has a valuable role to play. These applications are just a small sample of many potential applications that can benefit from an efficient rendezvous process. This book offers in particular an in-depth treatment of the blind rendezvous and oblivious blind rendezvous problems and their solutions. Rendezvous should be of interest to readers from other research fields such as robotics, wireless sensor networks, and game theory as the need for rendezvous arises naturally in many scenarios in these different fields.

"If I had my life to live over, I'd have fewer meetings and more rendezvous" (Robert Breault). Indeed, rendezvous is more than just a usual meeting which might not amount to anything; there is a purpose behind every rendezvous which is to enable the parties involved to establish a relationship and engage in an activity that will benefit both. Rendezvous makes things happen!

Should you have any questions or suggestions, please contact the authors via e-mail to zqgu@hku.hk, amywang@zju.edu.cn, qshua@hust.edu.cn, or fcmlau@cs.hku.hk.

Beijing, China
March 2017

Zhaoquan Gu
Yuxuan Wang
Qiang-Sheng Hua
Francis C.M. Lau

Rendezvous in Distributed Systems

Theory, Algorithms and Applications

Gu, Z.; Wang, Y.; Hua, Q.-S.; Lau, F.

2017, XIX, 262 p. 99 illus., 22 illus. in color., Hardcover

ISBN: 978-981-10-3679-8