Software intensive systems are increasingly expected to deal with changing user needs and dynamic operating condition at run time. Examples are context dependable interfacing with users, life reconfigurations, resource variability, and the need to deal with particular failure modes. Endowing systems with these kinds of capabilities poses severe challenges to software engineers and requires the development of new techniques, practices, and tools that build upon sound engineering principles.

Self-organization is an approach to engineer decentralized, distributed and resource-limited systems that are capable of dynamically adapting to changing conditions and requirements without external intervention. This useful system property is often reflected in functions such as self-configuration, self-optimization, and self-healing. Engineering approaches to self-organizing systems often rely on global functionality to emerge from local and autonomous decisions of individual agents that communicate through a shared coordination medium. Models for environment-mediated coordination are often inspired by biological, physical and other naturally occurring systems. Typical examples are gradient fields and digital pheromones that guide agents in their local context and as such facilitate the coordination of a community of agents. Since environment-mediated coordination has shown to result in manageable solutions with very adaptable qualities, it is a promising paradigm to deal with the increasingly complexity and dynamism of distributed applications.

Although we have some initial understanding how to engineer such systems, many research issues are still open. When designing a system that is based only on local interactions and the emergent properties resulting from these interactions, it is a difficult research problem on the one hand to obtain the required global behavior of the system and on the other hand to avoid undesired global properties. A particular issue in the design of self-organizing systems is determining the suitable complexity of the individual agents required to achieve the desired emergent functions. Typically, agents in self-organizing systems are less complex in their sensing, reasoning, and acting capabilities than agents in traditional multi-agent systems that follow a deliberate organizing approach. But, depending on the application domain, the functional requirements, and the sheer number of agents available in a particular setting, individual agent complexity may vary. As agent complexity increases, self-organization may become harder to achieve and to prove. ECOSOA addresses the approach of environment-mediated coordination among self-organizing agents that off-loads some of the agent complexity into the processes of the coordination medium (e.g., truth maintenance through pheromone evaporation). Off-loading agent complexity into the coordination medium simplifies agent design, implementation, and evaluation and thus increases the likelihood of a successful application development.

ECOSOA 2008 received 16 submissions. After a thorough review process, four papers were accepted as full papers and another four papers were accepted as short papers.