3 Problem Statement

This chapter defines the core research questions guiding the study and outlines the research strategy based on Design Science Research (DSR) principles. It details the methodology for developing and evaluating the ServerlessIntent framework, aimed at optimizing the orchestration of BaaS functions across cloud platforms.

3.1 Research Questions

The research questions tailored to answer the outlined research gap, are as follows:

- **RQ1:** Which parameters are important for BaaS services related to *ServerlessIntent*?
- **RQ2:** How can we overcome limitations of individual providers?
- **RQ3:** Considering RQ1 and RQ2, to what extent can we automatize the creation and orchestration of BaaS-enabled serverless functions?

3.2 Research Strategy

In this section, we outline the research strategy, drawing on Design Science Research (DSR) principles as described by Dresch et al. (2015). DSR is well-suited for the exploration and creation of innovative technological solutions in information systems. DSR is fundamentally a problem-solving process that involves the iterative development and evaluation of artifacts designed to address specific organizational or societal needs. In the context of information systems, these artifacts can be models, methods, constructs, or frameworks. The primary objective of DSR is not just to understand the world but to actively change it by introducing novel and effective artifacts.

The choice of DSR for this thesis is driven by several factors that align with the goals and challenges of developing the ServerlessIntent framework:

- **Innovation and Utility:** DSR emphasizes creating innovative solutions that have practical utility. The ServerlessIntent framework aims to innovate by abstracting and orchestrating heterogeneous BaaS services to enable complex cloud-based functionalities without vendor lock-in, aligning perfectly with DSR's focus on utility and innovation.
- **Iterative Development:** DSR supports an iterative development process where feedback from each phase informs improvements in the next. This iterative cycle is crucial for the ServerlessIntent framework, as it involves complex integration of multiple BaaS services and requires constant refinement to meet diverse cloud computing demands efficiently.