Bridging the ASV capability gap with cloud native microservices

Neville Danny[1], Carr Blaine [1], Davis Matthew[1]

[1] Spatialnetic Solutions Inc., Canada
danny.neville@spatialnetics.com

The number of deployed Autonomous Surface Vehicles (ASVs) has steadily grown over the last decade. These platforms are being applied to a wide range of missions and are operated by a diverse set of operators. Although holding significant promise, many ASV operators are still experiencing a gap between expected capability vs the invested resources. This gap is most significant in the areas of autonomous capability, reliability, and connectivity. At the core of these challenges is a wide range of software systems, including legacy applications, IT infrastructure and custom functionality. This complexity slows down the adoption of new technology, increases the cost of integrating hardware, and requires additional human resources to support.

Spatialnetics has been developing lightweight cloud native microservices to address these challenges, starting with improved ability to automate hydrographic workflows. Microservices are an architectural style which have been proven to be effective in reducing the complexity of distributed systems. They are used to develop a workflow as a suite of small services, each running independently. The goals are to reduce complexity with focused, relatively small pieces of functionality that do one thing well, as well as increasing resilience by allowing parts of a workflow to continue operating even with component failure. An additional objective is improved flexibility with the ability to independently scale, evolve, and deploy the microservices across autonomous platforms (edge), field computers, cloud resources, and in office desktops.

The Spatialnetic microservices have been successfully deployed both on ASV platforms and in virtualized cloud environments. Working either independently, alongside hydrographic acquisition software, or complementing a middleware suite such as the Robotic Operating System (ROS), the microservices have automated key parts of the processing workflow and operated as smart data bridges between incompatible components. This presentation describes these implementations and results.