Conclusion: OPC UA SaaS Deployment Questions

1. Is it necessary to instantiate a separate OPC UA server per customer?

Yes. While a single shared server is possible, per-customer instances are strongly recommended. This ensures data isolation, avoids noisy-neighbor issues, and simplifies compliance with security and confidentiality requirements.

2. How to implement automated instance provisioning for each customer to support scalability and isolation?

Automated provisioning can be achieved with Kubernetes and Infrastructure-as-Code (e.g., Helm, Terraform, GitOps). Each customer gets their own namespace, dedicated certificates, and isolated storage. This makes onboarding new customers fast and consistent, while maintaining strict separation.

3. How does horizontal scalability affect cost (infrastructure, licensing, and operations)?

Infrastructure: More instances mean more compute/storage, but elastic scaling reduces waste. Licensing: Commercial OPC UA stacks may require per-instance fees, while open-source stacks avoid this. Operations: Automation keeps DevOps overhead low even as customer count grows. Overall costs grow linearly with customers but remain predictable.

4. What are competitors doing, and how can Rayan Platform stand out?

Competitors like TagoIO, Prosys, and AVEVA rely on traditional subscription or usage-based pricing. Rayan Platform can stand out by combining per-customer OPC UA instances (ensuring data isolation) with automated scaling and flexible pricing (subscription + usage hybrid). This provides both security and cost efficiency as differentiators.