



Product Build | Application Modernisation

Employee Wellbeing Provider

Digital Employee Wellbeing, Healthcare

Our client provides technical employee wellbeing solutions for physical, mental, and financial health.

The client approached gravity9 and spoke with Founder Noel Ady to discuss a new module to their existing Mental Health solution that would enable communication between domain microservices and CRM (Customer Relationship Management) in a multi-cloud environment. Additionally, we would provide additional security, improve system performance, and provide domain separation.

The initial solution uncovered additional requirements, which gravity9 designed and built into the second iteration before conducting a final optimisation pass

to streamline further, displaying an evolutionary architecture approach.

The client was delighted with the overall result, especially gravity9's ability to adapt and implement event-driven architecture. gravity9 achieved a remarkable outcome that met all the client's expectations, demonstrating expertise in designing and deploying event-driven architecture and delivering a solution that supports asynchronous communication and facilitates evolutionary changes in response to the dynamic business environment.



Review of Challenges

The client approached gravity9 to discuss building a new product module, where our role would be in enabling communication between the app domains of their Mental Health solution and their CRM.

"Until this point, development teams needed to account for CRM internal data structure nuances and for complex CRM communication issues in order to deliver business functionality. This drained time, resources, and expertise, tying developers into unnecessary challenges."

gravity9 carried out a software architecture assessment to evaluate the quality and suitability of the system's design and structure. The system's components, interactions, dependencies, constraints, trade-offs, and risks were studied to identify strengths, weaknesses, improvement opportunities, and potential future threats.

Our Solution and Approach

The most significant and pressing issue concerned how apps could effectively communicate with the CRM.

"Evolutionary architecture is not about predicting the future, but about embracing change."

gravity9 set up synchronous and asynchronous communication between the Mental Health app domains, leveraging event-driven architecture and enabling support for third-party API integrations.

gravity9 ensured efficient performance by channelling information calls via a domain layer, creating a façade API, and enabling caching (both in memory and materialised). This ensured rapid call completion times via the façade API and significant performance optimisation during high-load request periods.

Utilised Technology Stack

Cloud: Multi-Cloud

QA: SpecFlow, Living Documentation

Development: .NET Core

The Façade API does not just provide faster communication; it also helps implement a more decoupled design. This aids flexibility within the architecture, with further backend data sources easily added in the future without multiple teams editing multiple API calls or creating new ones. The CRM is liable to change as updates are made over time, so the API layer helps isolate these changes. Change Data Capture (CDC) was also introduced to the Façade API, which identifies and captures changes made to a data source and then relays those changes to any downstream processes and systems.

The API layer improves the system's security as it is not a direct call to the

backend system. A standards-based security mechanism provides granular protection for read-and-write execution.

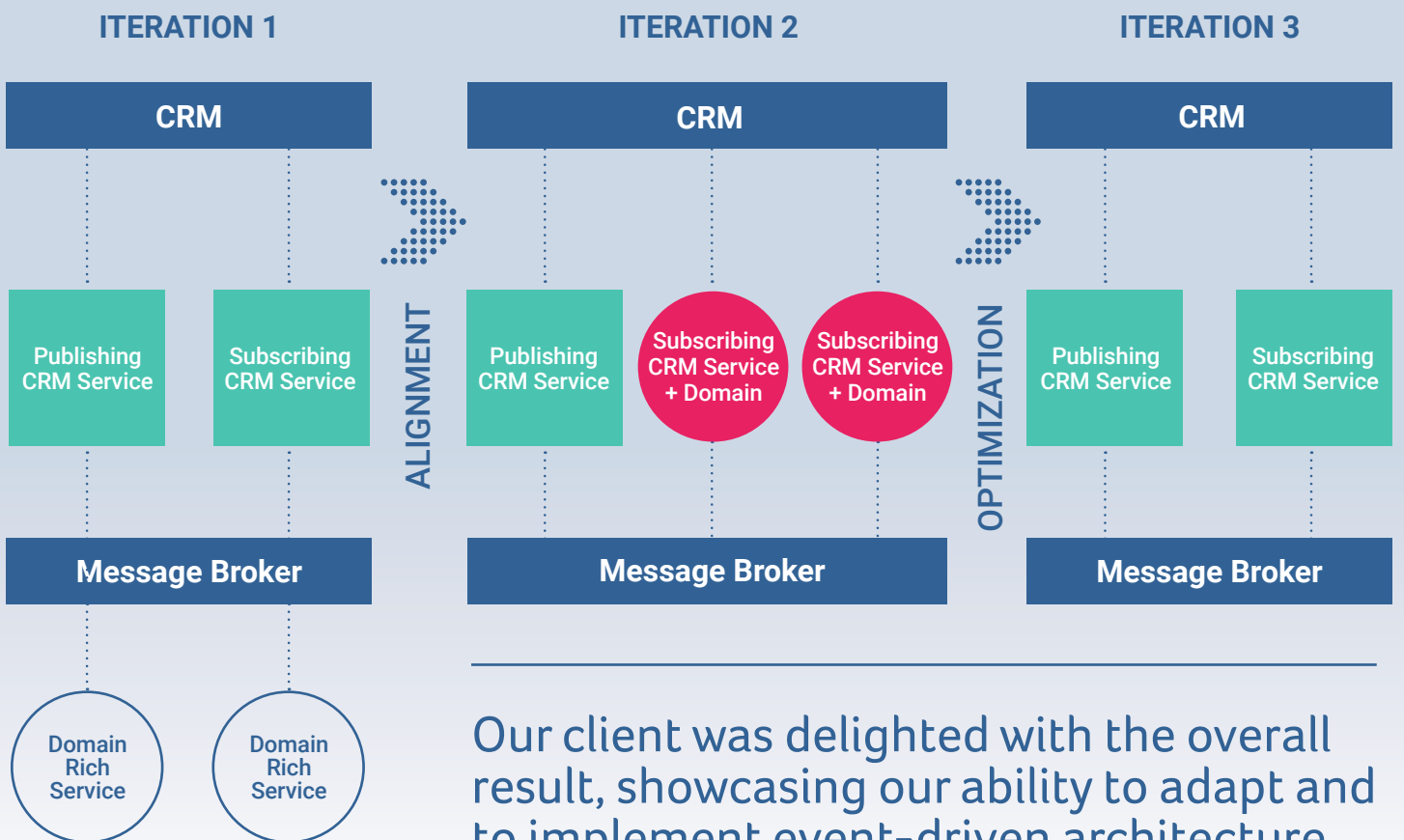
To improve development quality, stability, security, and system performance - automated integration testing frameworks were created and supported for both back-end and front-end within the Mental Health app, and detailed logging, tracing and instrumentation tools were also implemented.

The process of implementing this solution was not a straight line from requirements gathering to final delivery. gravity9 utilised the evolutionary architecture approach to identify further requirements after the initial

consultation. These were implemented in a subsequent update to architectural and system design. This second solution better addressed the client's new needs, but a final update pass was conducted to ensure the finished solution was functional and optimised.



Our evolutionary architecture process



Our client was delighted with the overall result, showcasing our ability to adapt and to implement event-driven architecture.

Subsequent Outcomes

The client selected gravity9 to provide architectural improvements to their systems to address communication, security, and scalability issues.

A key point of this case study is that it neatly shows not every project is a straightforward line from commencement to delivery; sometimes requirements change, or new ones appear and – thanks to gravity9's expertise in applying evolutionary architecture, to the solution could be revised and then optimised to fit the client needs.

Incredibly positive feedback was received from the client, and their application teams can now focus on delivering business features instead of being sidetracked. The implemented solution significantly improves communication across the architecture as well, ensuring a more secure environment that can be scaled and more easily modified to suit changing structures. At the time of writing, this puts gravity9 in an excellent position to provide support and assist with integrating further systems within existing infrastructure.

**We improved development
quality, stability, security,
and system performance**

Visit our Insights page to find more articles around emerging technology trends, the IT industry, interviews and much more!