The CommonTime Infinity Server has been designed to support the complex integration and communication needs of the UK’s largest public sector bodies.

Although all modern, mobile communication systems require a robust, scalable server-side environment in order to handle an increasing number of connected users & devices - nowhere is this more apparent than in the organisations which provide crucial public services.

The Infinity Server - written in Erlang, implemented with Ejabberd and deployable via Docker - is a vital element of CommonTime’s next-generation messaging solutions. This can be hosted in the cloud or on-premise depending on client requirements.

The key benefits of this setup include a clustered structure that supports automatic load-balancing and failover, adherence to all SSL/ TSL best practices and the option between a web-based & command line interface for administration.

In addition to supporting high volumes of traffic, we have developed a server environment which is extensible via pluggable modules, bots and third party data sources. This architecture is vital to our vision of messaging applications as a platform upon which to build.

The result of this is that applications utilising the Infinity Server are able to integrate into recognised clinical systems - including electronic patient records, patient administration systems and any other software users would access on a regular basis.

Ejabberd - the technology behind our Infinity Server - has a rich 12 year history and supports a range of services for organisations such as the BBC, Nokia, Kodak, Facebook, Snapchat and WhatsApp.

The Advantages of Erlang & Elixir

The CommonTime Infinity Server is predominantly written in Erlang - a stable, general purpose programming language & runtime environment with over 20 years of history. Initially developed by Ericsson, Erlang has been used in production by a number of high-profile organisations including; Amazon, Yahoo, LastFM and Github. However, most importantly, Erlang supports a high volume of concurrent activities, task distribution and fault-tolerance.

Elixir is a much more modern language which compiles down to the Erlang VM, providing a powerful macro capabilities and a rich tooling ecosystem. Developed for building scalable, yet maintainable, applications - Elixir code runs inside of lightweight threads of execution. Because of this structure, Elixir is ideal for running thousands of concurrent processes; a vital element of any communication system.

Every decision made in the design and development of the Infinity Server has been in order to create a powerful, future-proof platform that can handle a limitless volume of messages and integrations. With Erlang and Elixir, our team have found this - all within a secure, controlled environment.