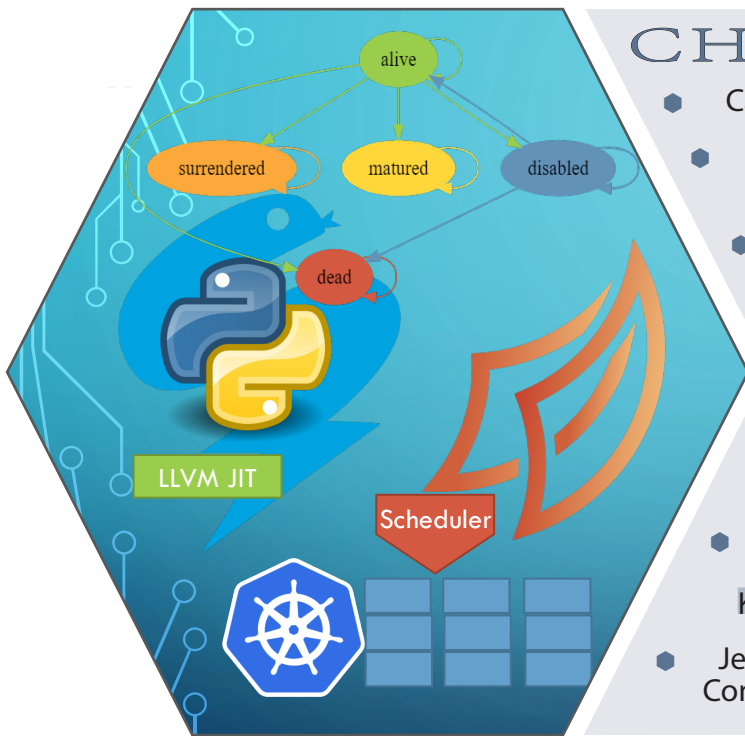


Highly performant Python code with JIT compilation and parallel/distributed computing.

We have successfully implemented a framework to allow the Life team of an Insurance Company to perform the implementation and testing of life actuarial models, as required by regulators, in a highly streamlined and cost-effective manner.



Current performance is comparable to highly optimized commercial solutions written in C++



CHALLENGES

- Calculations in pure Python are slow
- Complex state models based on time inhomogeneous Markov chains
- Scaling up to 1000s of CPUs
- Reduce risk of failure and improve the quality of the delivery

SOLUTION

- Python framework makes use of Numba to accelerate calculations
- Dask provides advanced parallelism for analytics. Using Dask for distributed calculations in Kubernetes cluster enables to scale up to 1000s of CPUs
- Jenkins and Bitbucket play a key role in the Continuous Integration pipeline

BENEFITS

- **70% cost saving:** Replacing expensive proprietary solutions with open source software and the ability to scale computational resources on demand will allow up to 70% cost saving after an initial transition period.
- CI set-up contributes to minimizing the number of bugs and regressions and gives confidence when modifying the codebase.



"Continuous Integration is very rare in actuarial modeling. Designing and implementing the CI pipeline was one of my key contributions to the project success. I also enjoyed the extensive usage of open source, cutting-edge technologies."

