

Peter Kirkham

Experienced developer/tech lead/team lead on projects creating software to support engineering modelling, condition monitoring, alerting and simulation, in research and development and production environments, in energy and building management industries.

Key Skills:

- Shipping useful software
- Object oriented software design
- Technical/engineering computing
- Recent Python, C#, TypeScript development; historically further languages
- PostgreSQL
- REST/HATEOAS/OpenAPI

Employment History:

August 2022-November 2025: Software Engineer, BitBloom.tech,
remote/Glasgow

BitBloom creates software for advanced analytics and performance monitoring of wind farms.

I was engaged as a full stack developer with backend Python (Tornado, asyncpg, PostgreSQL, HDF5, pytest, Pandas) and front-end TypeScript (React, Grommet, playwright) in an nx monorepo using GitHub and Azure CICD pipeline; deployed on AWS EC2, batch and S3.

This required flexible systems capable of batch processing of large amounts of data on the cloud to determine trends and anomalies, but only needed to support a few users, so the architecture reflected the requirements using a simple Tornado backend and AWS batch for the heavier processing. Data was fetched from the turbines' SCADA or first tier supervision systems via a mixture of database access, file transfer or custom APIs.

As a small team the role was varied and informal, talking to the in-house analysts, establishing requirements for new features, dealing with day-to-day emergent issues and outages, guiding the UX designer on technical issues, developing the software (most of the time), evolving the architecture, code reviews, keeping the CICD running, updating packages and dependencies, internal feature demos, and deployments to the cloud.

The backend used a code generator to create repositories to persist data classes to PostgreSQL. I took over development of this, rearchitected to use command/query separation and to move business logic out of the HTTP handlers and into more easily testable units. Separation was added between HTTP handlers that parsed and processed the API and commands which

performed the actions and generated events that could flexibly trigger further asynchronous or long running operations. I also spent some time optimising the front end plots using WebGL, as none of the libraries we tried proved capable of handling the number of data points some users wished to see.

Unfortunately BitBloom failed to meet its sales targets and is stopping new feature development and downsizing its development to a single junior developer, so my role no longer exists.

March 2013 to June 2022: Technical Leader, IES, remote/Glasgow

IES creates software for building energy use optimisation. Their traditional base is simulation of buildings prior to construction, but are moving into monitoring and model-based control of buildings in use.

My role was as the technical & team lead on the iSCAN project to create a time series database suitable for long term monitoring and calibration of thousands of buildings over many years, as well as handling tens of millions of streaming data points per day. I was also the line manager and mentor of the iSCAN development team of eight people. As of 2022, the main deployment was receiving around half a billion data points a year, with 4½ billion in its history. It pulls data in many formats from building and industrial control systems and provides visualisation, analytics and rule based alerts on time series to support optimisation for energy efficiency.

This was implemented using a C# MVC application, HTML5/JavaScript front end, REST (HATEOAS) API, a Python 3 client library for scripting the API, on Azure services with PostgreSQL and Azure blob storage. We were moving the front end to React and backend to .net 6 when most of the team got moved to another project.

July 2010 to March 2013: Software Engineer (contract to April 2011, then permanent), Qualitrol Diagnostic Monitoring Systems, Glasgow

Qualitrol Diagnostic Monitoring Systems create hardware and software systems to monitor gas insulated electric switchgear for faults, detecting faults using machine learning of time/frequency profiles of partial electric discharges.

I was recruited as a contractor and I took the lead developing a rule engine (on a principle similar to a Rete T-box) for activating alarms on maintenance events, and also took the lead in designing the interface architecture and protocols between the components of the condition monitoring system. I developed an efficient serialisation means for extensible messaging between the components, interfaced to bespoke hardware and worked developing UI components and championing consistent design. I developed the tools for configuring the system and live debugging and diagnostic tools. I made several improvements to the use of continuous integration and testing in the team. When faced with a tricky bug in concurrent code, I developed a simple state enumeration tool based on applied pi-calculus which found a path to an invalid system state and allowed the bug to be fixed. I also provided holiday cover for the expert services group who provide reports to customers diagnosing the faults in their electrical systems. I led a project to create a WinEC7 version of the condition monitoring system, which meant I was managing a couple of engineers on a day-to-day basis.

October 2009 to June 2010: R & D Management Systems Engineer (contract), Areva T&D, remote/Stafford

A series of fixed term contracts supporting multi-site use of EA SysML with ClearCase. Discussing data integrity issues with end-user engineers, creating a model of the processes using UML activity diagrams and subjecting the model to formal analysis to identify scenarios where updates are lost due to sequencing of actions between sites (using problem-specific model checker EA plugin written in C#).

May 2008 to December 2008: Software Engineer (contract), EADS Astrium, Stevenage

Fixed term contract creating software for satellite communications monitoring and control.

I was recruited to work with RS232 and SNMP protocols, connecting communications equipment to a SCADA system according to a systems definition in UML, but completed that task quicker than the other engineers, using a Python script to generate the code from the UML. So instead of working with the protocols directly, I was tasked with creating a UML profile and programming plugins for Enterprise Architect in Python to automate the process of generating the monitoring software components, so that all the engineers would benefit from a more streamlined process.

As the managers liked the UI for the tools I'd provided for supporting the engineering process, I was then tasked with replacing the MFC framework for the front-end of the SCADA system (itself ported from Xmotif) with a Java Swing version. This included creating a parser expression grammar parser for the existing DSL used to configure the system.

February 2008 to March 2008: Consultant (contract), QinetiQ, Malvern

Fixed term contract creating test strategy and procedures for software controlled radio equipment, and helping integration of an embedded computer system with bespoke RF electronics components.

November 2005 to November 2007: Senior Software Engineer (permanent), General Dynamics UK, Newbridge, Gwent

GDUK is the prime contractor for the Bowman digital radio system, used for communication by the UK military. I was a senior software engineer in the BLaDE group, creating a simulation environment for autonomous forces simulation - representing the units on a battlefield, their actions and messages. This allows different scenarios to be simulated to trial communications and doctrine without running a full military exercise. The simulation system was used for different wargaming uses, including supplying a virtual battlegroup to augment live field trials of Bowman, operator training, mission planning as well as trades studies of potential algorithms and capability enhancements.

I joined the group about a year into the project, when the team had seven members, and it grew to about a dozen when I left. I had responsibility for green-field design and implementation of the message hub which supported the distributed service discovery and pushing simulation state to the clients (around 10K messages per second), and for one of the user interface frameworks - providing 2D map and military icon (APP-6A) interaction, the primary interface exploited by the users of the system.

I also took on responsibility for evolving the design of the extant event-driven simulation engine, and of the TCP/IP communication and XML service libraries. I also evolved the composition mechanism used in the model from a mechanism requiring explicit wiring up of many ports of subcomponents, to one based on traits that allowed domain modellers to just drag and drop capabilities into their models without having to worry about implementation details.

I added to the software engineering quality of the team by introducing automated unit testing, the use of Subversion source control, Bugzilla bug tracking, and the Moin Moin wiki for documentation. The simulation engine and message hub was written in C++, and the 2D front end in XUL/JavaScript, with supporting model transformation and checking tools in XSLT2. The service framework uses XMPP pub-sub and discovery mechanisms to allow loose coupled communication between processes and machines. A third party provided bridging between the message hub and external HLA simulations.

November 2001 to November 2005: Lead Software Engineer, Technical Computing, BAE Systems (Warton).

The technical computing group handled a wide range of development work, and I would get assigned difficult projects that required high calibre technical skills to turn around. Working as an in-house bespoke software resource, projects typically involved teams of up to half-a-dozen engineers from the group, and lasted a few months. The work was primarily doing hands-on design and implementation in C++ and Java, but also had a research component.

I was the joint technical authority for the technical computing group, having responsibility for conducting and reviewing object-oriented design in Java across the group of around twenty engineers (the other authority covered Fortran, we shared C++). My responsibilities included advising management on tool and technology selection, mentoring junior engineers, and sourcing or creating infrastructure to enable the engineering and operational analysis capabilities BAE requires.

The work involved varied aspects of technical computing in an aerospace engineering firm - simulations for emergent properties such as combat capability and survivability, and analytic tools for budgeting weight, performance or radar signature. The simulations were mainly discrete event driven 5Dof models with state-machine AI using SEP or drag/lift table based platform models. The simulations were primarily applied to air-to-air combat operational analysis, but also touched on mission planning. I also produced some research studies - an algorithm design for UAV collision avoidance, resulting in a patent, and feasibility studies on creating qualitative bond graph models for model based system health monitoring from SysML descriptions, and on moving expensive computations from legacy hardware to idle time on the company's PCs using grid computing tool-kits.

I was technical lead on several projects, having responsibility for the engineering aspects from requirements capture, cost estimation, application and framework architecture design, test,

implementation, through to final delivery and deployment. Projects typically had two or three engineers and lasted four to six months. The technical lead role required that I developed a good relationship with our in-house clients, liaised with subject matter experts, and mentored junior engineers. The development work primarily used C++ and Java for engines, and Java for front end using Swing or JOGL/OpenGL thick clients or thin web pages as appropriate to the degree of visualisation required. There was some maintenance and call graph and module dependency analysis of existing Fortran 77 code.

September 1993 to June 2001: Research Associate, High Integrity Systems Engineering, Department of Computer Science, University of York.

Primarily working as the architect of the Safety Argument Manager (SAM), a system for the representation and analysis of graph based notations showing the dependencies between arguments about the safety of complex engineering systems, and workflow and white-board systems for conceptual engineering use. Many HISE papers were based on research facilitated by the tools I created. SAM was written entirely in MS VC++, using MFC. It incorporated an interpreter for a scripting language for extensions, and tools to define graphical notation schemas and map their data models to SQL persistence using ODBC. The workflow tools were web-based, using Tomcat and Cocoon.

Although I've left the technology from this era behind, the High Integrity Systems mentality of risk and issue management and system resilience has continued to be a strong influence on systems I went on to design and support.

July 1991 to December 1991: Junior Software Engineer, Rotork Instruments, Luton

Creating a tool for reliability estimation by building block diagrams of petrochemical plant control systems from a library of logical components populated with transfer function and FMECA data. The tool was written in PASCAL on MS DOS.

Education:

- 1992-1993 MSc, Software Techniques for Computer Aided Engineering, Cranfield University.
- 1988-1991 MA, Electrical and Information Sciences Tripos Cambridge University, (got first in part 1 but spent too much of time playing with Lisp for my final year project so went down to a second)
- School: Luton sixth-form college, A levels in maths, further maths, physics, chemistry, all grade 'A' . Icknield High school, Luton, 12 O-levels .

- 2005 M208 Pure Maths Open University module - distinction. (Only did the one module of out of interest in pure maths as I'd mainly been using applied since university)

Miscellaneous

- Full Name: Peter Mark Kirkham
- Postal Address: 2 Kirk Wynd, Stirling, FK7 9AY
- Telephone: 07884 321 747
- Email: pete-2025@tincancamera.com
- British Citizen.
- Full driving licence.