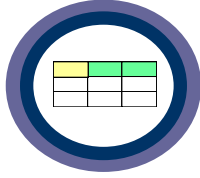


Axe is a test automation platform that greatly increases productivity and drastically reduces maintenance overheads for test automation. It uses simple Microsoft Excel spreadsheets as a means to define test scenarios and code generation techniques to rapidly generate robust, self-documented automation code. Axe is unique and proven in enabling automation in an Agile environment.



Testers design tests in Microsoft® Excel. No code, no scripts, just business logic and data in a simple modular format.



With one click code and documentation are automatically generated by Axe.



Execute tests in a range of market leading test automation tools. Axe leverages the strengths of the tools without the complexity.

## KEY BENEFITS

### Proven return on investment on automated testing

Leverage investment of automation tools by cutting implementation costs by up to 75% and ongoing maintenance by up to 86% making automation affordable for your projects now.

### Easy to use by all levels of testing staff

Facilitate automation using your existing non-technical staff so that automation is incorporated easily into your mainstream testing making it easier to manage and improve job satisfaction.

### Quick and easy to implement

Implement automation with minimal impact on your testing timescale so that you're building assets for the future at the same time as you undertake your testing.

### Automate testing throughout the development lifecycle

Apply automation to tests for both UI and non-UI parts of the application. Component layers can be tested early to improve the overall quality of the development deliverables.

### Future proof your automation investment

Axe tests work with all leading automation tools so you achieve greater flexibility on selection of software suppliers and offshore partners.

## STRATEGIC BENEFITS

Skills used to build tests are transferable to other automation tools, just like other testing skills, giving you greater testing resource flexibility for your projects.

Specialist automation technicians spend minimal time supporting a particular testing project, and can support more than one project at a time. So, this expensive resource can be deployed more cost effectively within your organisation.

The use of standard technology (Microsoft® Excel) to implement Axe provides you greater flexibility on where you locate and resource your test activity.

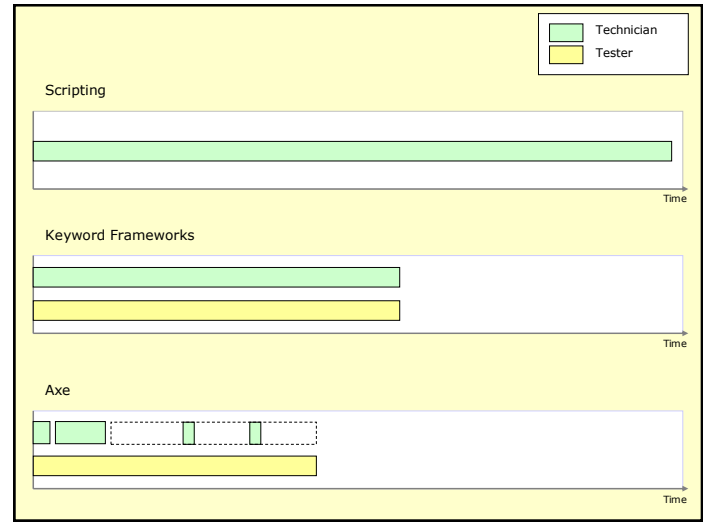
## Reduced Technical Resource

Using the Axe framework changes the shape of how automation projects are resourced, reducing costs. This is shown in the diagram opposite.

A traditional script based approach relies on technical expertise throughout the project lifecycle to hand craft tests in the code-like scripting languages used by tools.

The emergence of early "Keyword frameworks" split the work between Testers and Technicians. A Technician is still required to write script modules, but Testers define the sequence of modules and the required data for tests.

Axe empowers Testers to define all elements of a test, business workflow and data. A Technician is only required to setup and support a project, lowering costs.

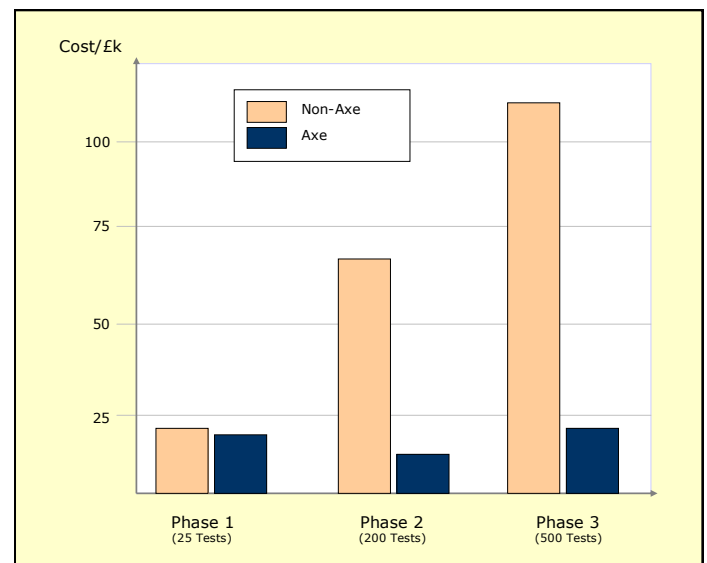


## Reduced Cost of Implementation

Axe significantly reduces the cost of implementation of automation, giving much quicker Return on Investment. The chart opposite shows typical figures for the Axe approach side by side with a script based approach for a project over 3 phases:

- 1 Initial implementation and a 25 test confidence pack.
- 2 Implementation of 200 automated regression tests.
- 3 Scaling up to a regression pack of 500 tests.

By reducing the required Technical expertise and allowing testers to define tests significantly quicker and more efficiently in Excel costs can be less than a quarter with Axe.



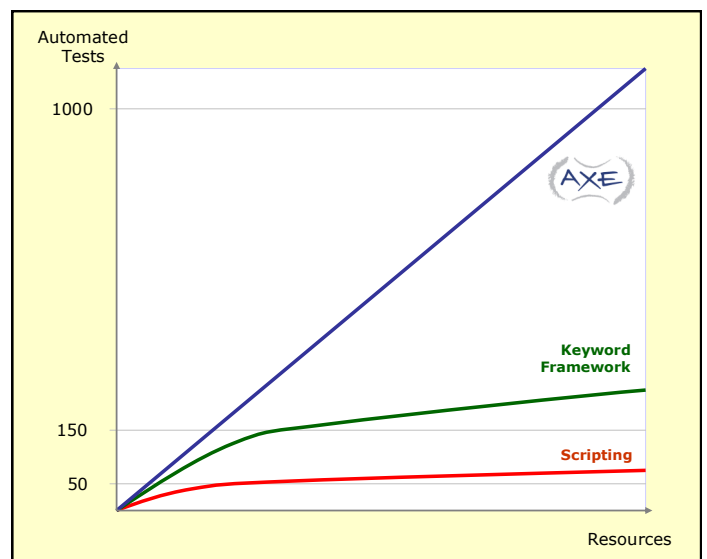
## Highly Scalable

Projects based on the scripting and the use of keyword frameworks rarely scale beyond the figures shown here.

Scripting is effectively software development and prone to all the issues this presents. To scale any software development enterprise a suitable infrastructure needs to be in place. Even simple things like source control are overlooked, leading to the figures shown.

Keyword frameworks still employ scripting but in a more structured way, the problems of scaling are still present but the structure helps improve on scripting alone.

Axe has regularly been used on projects where many hundreds and even thousands of tests are automated. By removing the need to write code-like scripting languages the issue of scalability becomes a simple matter of organising spreadsheets.



tel: +44(0) 118 903 6101  
fax: +44(0) 118 903 6100

email: [info@odintech.com](mailto:info@odintech.com)  
web: [www.odintech.com](http://www.odintech.com)

Atlantic House - Imperial Way  
Reading - RG2 0TD - UK