ABGi

WHITE PAPER

AI, Low Code and UK R&D Tax Relief: What Still Qualifies as R&D?

CONTENTS

1. What's changed? 04

2. AI projects: what is and is not likely to be R&D? 06

3. Low code and no code: when does work cross into R&D? 08

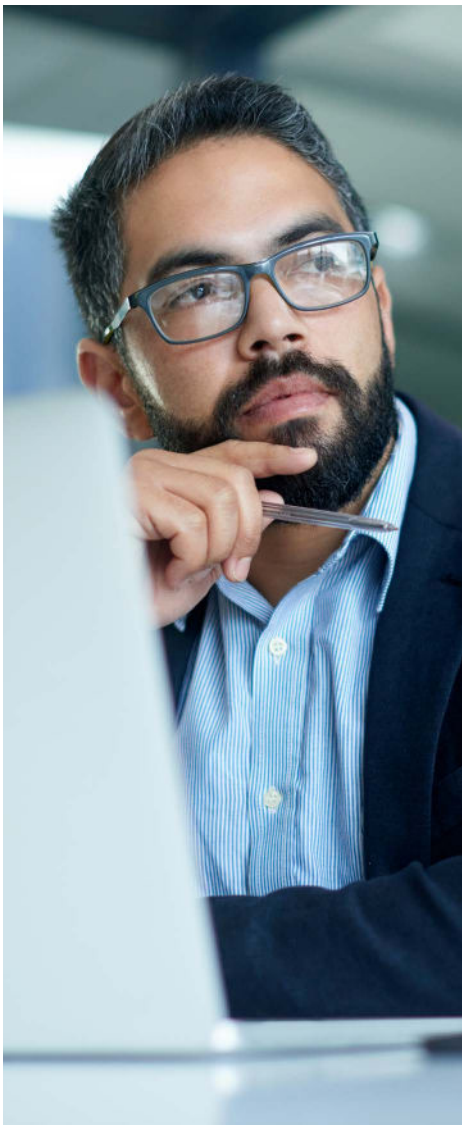
4. Project scenarios examples 09

5. HMRC scrutiny, AI use and why evidence matters more than ever 10

6. What this means for your next claim 11

AI, Low Code and UK R&D Tax Relief: What Still Qualifies as R&D?

Software teams have changed the way they build products. AI models, coding copilots and low code platforms mean work that once took months can sometimes be done in days.



Yet the statutory definition of R&D for UK tax purposes has not changed. The bar remains an advance in science or technology, and the resolution of genuine scientific or technological uncertainty, assessed against what a competent professional in the field could readily work out at the time.

This paper is for CTOs, engineering managers and finance leaders who are asking: ***"If we're using the same AI and low code tools as everyone else, do we still have qualifying R&D?"***

It explains how the existing rules apply in an AI heavy, low code world, where the baseline has moved and evidence expectations have risen.

01 The statutory definition has not changed, but the baseline has

Under the " CIRD81910 - R&D tax relief: conditions to be satisfied: DSIT Guidelines", R&D occurs when a project seeks to resolve scientific or technological uncertainty in order to achieve an advance in overall knowledge or capability in a field, not just for the company.

In software and AI, that means tackling problems where it was not clear, based on publicly available knowledge and the skills of a competent professional, how to make something work in practice.

→ THREE TESTS STILL SIT AT THE HEART OF ANY CLAIM:

ADVANCE IN SCIENCE OR TECHNOLOGY	» an appreciable improvement in underlying capability, performance, reliability or efficiency, beyond routine use of available tools.
SCIENTIFIC OR TECHNOLOGICAL UNCERTAINTY	» it was not readily deducible how to achieve the desired outcome using existing techniques and information.
SYSTEMATIC, INVESTIGATIVE AND EXPERIMENTAL ACTIVITY	» a structured process of analysis, design, prototyping, testing and iteration to overcome those technological uncertainties.

AI does not have its own bespoke regime for R&D tax relief purposes, it is assessed under the same principles as any other area of software or technology. HMRC views it as one of many domains where software R&D can arise, alongside areas like cloud computing, robotics and geoinformatics. Equally, "AI project" is not a magic phrase that makes work qualify, projects still have to be assessed against the same tests as any other software R&D.

The statutory definition has not changed, but the baseline has *(cont)*

How AI and low code have moved the "competent professional" baseline

The DSIT guidelines ask whether a competent professional could readily solve the problem using available knowledge at the time.

The last few years have transformed what "readily available knowledge" looks like:

- Open source models, libraries and frameworks cover many standard AI and data problems.
- Low code and no code platforms ship with extensive blueprints, connectors and best practice patterns.
- Public forums, documentation and now AI copilots can generate working implementations of many routine patterns in minutes.

As a result, some work that felt exploratory in 2018 is now routine implementation.

Problems that previously required significant development time can now often be solved more quickly by a competent professional using today's tools, which means they may no longer meet the technological uncertainty test.

→ The implication is that you must be more precise about where the real technological uncertainty lies, and more disciplined in documenting it.

02 AI projects: what is and is not likely to be R&D?

AI has become a broad label covering everything from simple API integrations to cutting edge model research. For R&D purposes, the distinction between using AI and advancing AI is crucial.

☑ EXAMPLES MORE LIKELY TO QUALIFY (IF DOCUMENTED PROPERLY)

The following types of work will often involve genuine technological uncertainty and a potential advance, provided they go beyond established architectures or publicly available knowledge and are backed by contemporaneous evidence:

☑ DEVELOPING NEW OR SIGNIFICANTLY IMPROVED MODELS OR ALGORITHMS	For example, devising an appreciably improved training approach, architecture or development techniques to achieve materially better accuracy, latency or reliability than current methods for a defined problem.
☑ SOLVING NON-ROUTINE DEPLOYMENT AND SYSTEMS PROBLEMS FOR AI AT SCALE	Tackling instability, model drift, real time latency, throughput, memory constraints, or complex orchestration across distributed systems, where existing architectures or established patterns cannot readily achieve the required performance or reliability.
☑ OVERCOMING SEVERE DATA RELATED CONSTRAINTS	Designing appreciably improved approaches for learning from sparse, noisy, biased or highly regulated datasets, such as privacy preserving training, new methods of synthetic data generation, or domain adaptation techniques where the behaviour of the system could not be predicted in advance using existing methods.
☑ INTEGRATING AI INTO COMPLEX LEGACY OR SAFETY CRITICAL ENVIRONMENTS	Where there is genuine uncertainty around performance, reliability, safety or interoperability, and where resolving that uncertainty requires iterative experimentation rather than straightforward engineering.

In each of these scenarios, the R&D is not “we used AI”, but the specific unresolved technical questions the team had to answer to make a system work, and the experiments they ran to get there.

AI projects: what is and is not likely to be R&D? *(cont)*

☒ EXAMPLES UNLIKELY TO QUALIFY ON THEIR OWN

On the other hand, many common AI related activities look more like routine application of existing knowledge and will not usually meet the statutory tests:

☒ PLUG AND PLAY USE OF EXISTING AI PLATFORMS OR APIS	Calling a standard API, wiring up a basic chatbot using vendor provided templates, or doing ad hoc prompt engineering to improve responses, without materially extending or improving the underlying technology.
☒ PURELY COMMERCIAL OR COSMETIC ADAPTATION OF AN EXISTING MODEL	Re branding, configuring different customer segments, tuning content tone or style, or building new user interfaces, where the underlying model behaviour and system architecture remain unchanged and well understood.
☒ USING AI PURELY AS A PRODUCTIVITY AID	Relying on a coding copilot or text generator to produce boilerplate, tests or documentation, while solving problems using standard patterns that a competent professional could readily apply

AI focused guidance for software developers from government stresses the importance of human oversight, strong testing, and clear accountability for any AI assisted code or content. That guidance dovetails with HMRC's expectation that R&D claims must be grounded in human led technical problem solving and robust evidence, not automated output produced without expert review.

03 Low code and no code: when does work cross into R&D?

Low code and no code platforms blur the line between software engineering and configuration.

They can enable non developers to deliver substantial business value – but that alone does not make activity R&D. **→ A USEFUL WAY TO THINK ABOUT THESE PROJECTS IS IN THREE TIERS.**

TIER 1
Standard configuration (usually not R&D)

This tier covers projects where teams:

- Assemble workflows, CRUD applications, dashboards or portals using out of the box components.
- Follow vendor documentation, templates and community examples to connect common systems (CRM, payment gateways, email, storage, etc.).
- Face issues that can be solved by applying known patterns or asking the vendor/partner support channel.

Here, work tends to involve little or no scientific or technological uncertainty. A competent professional familiar with the platform could readily achieve the same outcome using available knowledge, so the statutory tests are unlikely to be met.

TIER 2
Pushing the platform to its limits (sometimes R&D)

R&D may arise when teams go beyond typical usage patterns and encounter genuine technical roadblocks, for example:

- Performance issues at scale (concurrency, throughput, latency) where the platform's recommended patterns fail under real world loads.
- Security or data segregation constraints (multi tenant architectures, complex permissions models, encryption requirements) that are not addressed in standard documentation.
- Novel integrations with unusual or proprietary systems where there is no existing connector or recipe, and where straightforward approaches do not work.

If resolving these issues requires experimentation, systematic investigation, and possibly extending the platform's capabilities, then the project can contain R&D even though it uses a low code tool. The key is that technological uncertainty lies in whether and how the desired technical behaviour can be achieved on the platform, not in project planning or commercial risk.

TIER 1
Extending or reinventing the platform (often R&D)

At the far end of the spectrum are projects where teams:

- Designing and developing appreciably improved components or execution mechanisms where it was not readily deducible how to achieve the required functionality or performance using existing technologies.
- Develop new runtime architectures, compilers, orchestration layers or scaling mechanisms that change how the platform operates.
- Create new frameworks or SDKs that significantly expand what can be built on top of the low code environment.

These kinds of activities are much more likely to involve genuine technological uncertainty and an advance in the field, particularly when they deliver appreciable improvements in performance, scalability or security over existing approaches. In many respects, they resemble traditional software R&D carried out using a low code environment as a substrate.

04 Project scenarios: how HMRC's tests apply.

The table below illustrates how the same tools can give rise to very different R&D positions depending on how they are used.

SCENARIO (AI / LOW CODE)	LIKELY R&D POSITION	WHY THIS IS / ISN'T R&D
Implementing an AI chatbot using an off the shelf platform and standard flows	☒ **Not usually qualifying**	Activity is mainly configuration and integration using vendor templates and standard APIs; there is little technological uncertainty beyond known behaviour.
Developing a model to handle highly specialised domain language with performance beyond current methods	☑ **Potentially qualifying**	Seeks an advance in capability; team must experiment with architectures, features and training techniques to achieve required accuracy and robustness under constraints.
Building internal apps on a low code platform by following vendor blueprints	☒ **Not usually qualifying**	Work follows established patterns documented by the platform; a competent professional could readily achieve the same result.
Extending low code with custom components to overcome severe performance limits at scale	☑ **Potentially qualifying**	Team must overcome undocumented technical limitations; involves systematic investigation into concurrency, memory usage or architecture beyond the platform's original design.

These examples highlight a theme: the tools do not determine eligibility; the unresolved technical questions do. Two companies can both "do AI chatbots", but only the one tackling uncertain, non routine technical challenges will have qualifying R&D.

Good documentation and human oversight

Two themes emerge from both HMRC guidance for software developers and commentary on AI in R&D claims:

- **Document the uncertainty and the journey, not just the outcome.** Capture, as you go, the specific technical risks you identified, the alternative approaches you tried, the failures you encountered and the benchmarks you used to judge success. This makes it far easier to demonstrate that work met the DSIT criteria if HMRC queries a claim.

- **Keep human experts firmly in the loop.** AI can assist with drafting, coding or summarising, but responsibility for what is submitted to HMRC sits with named individuals. Technical leads should review and approve the way uncertainties are described and ensure that narratives and figures accurately reflect the work done.

Handled well, this environment actually favours companies with genuine software R&D. If you can clearly articulate your technological uncertainties, show how your work goes beyond routine use of AI and low code tools, and evidence a robust process, you are better placed than those relying on generic "AI optimised" claim templates.

05. HMRC scrutiny, AI use and why evidence matters more than ever

HMRC has significantly increased scrutiny of R&D claims in recent years, including those involving software and AI.

There has also been public debate over whether HMRC itself is using AI in its R&D compliance processes, with a recent tribunal ordering greater transparency around any such use. Alongside this, professional commentary has warned that AI generated reports and claim narratives, if not carefully reviewed, can trigger risk flags or undermine credibility.

Risks of over claiming on generic AI and low code work

Over claiming on projects that are essentially routine implementations now carries greater enforcement risk than in the early years of the regime, especially where:

» Claims describe generic AI or low code work without pin pointing specific technological uncertainties.

» Narratives look templated or boilerplate, with vague references to "complex algorithms" or "novel models" unsupported by technical detail.

» Evidence of systematic investigation (design decisions, experiments, failed approaches) is thin or reconstructed after the fact.

As tools make it easier to build complex applications quickly, the temptation is to treat all such work as R&D. HMRC's stance is the opposite: the more tooling and guidance is available, the more you must demonstrate why your problem could not be solved by applying that existing knowledge.

06 What this means for your next claim

For software teams, the practical implications are:

DO NOT WRITE OFF ALL FUTURE CLAIMS JUST BECAUSE YOU NOW USE AI OR LOW CODE – focus instead on where your work still involves non-trivial technological uncertainty.

BUILD R&D THINKING INTO YOUR ARCHITECTURE AND DESIGN PROCESS: when you see a problem that can't be solved by applying known patterns, flag it as potential R&D and track it separately.

STRENGTHEN COLLABORATION BETWEEN ENGINEERING AND FINANCE so that project selection, evidence gathering and claim preparation reflect how work is actually done.

FOR FINANCE AND TAX TEAMS, it may be a good moment to re map your project portfolio, identify where real technological uncertainty remains in an AI rich environment, and adjust your approach to claims and documentation accordingly.

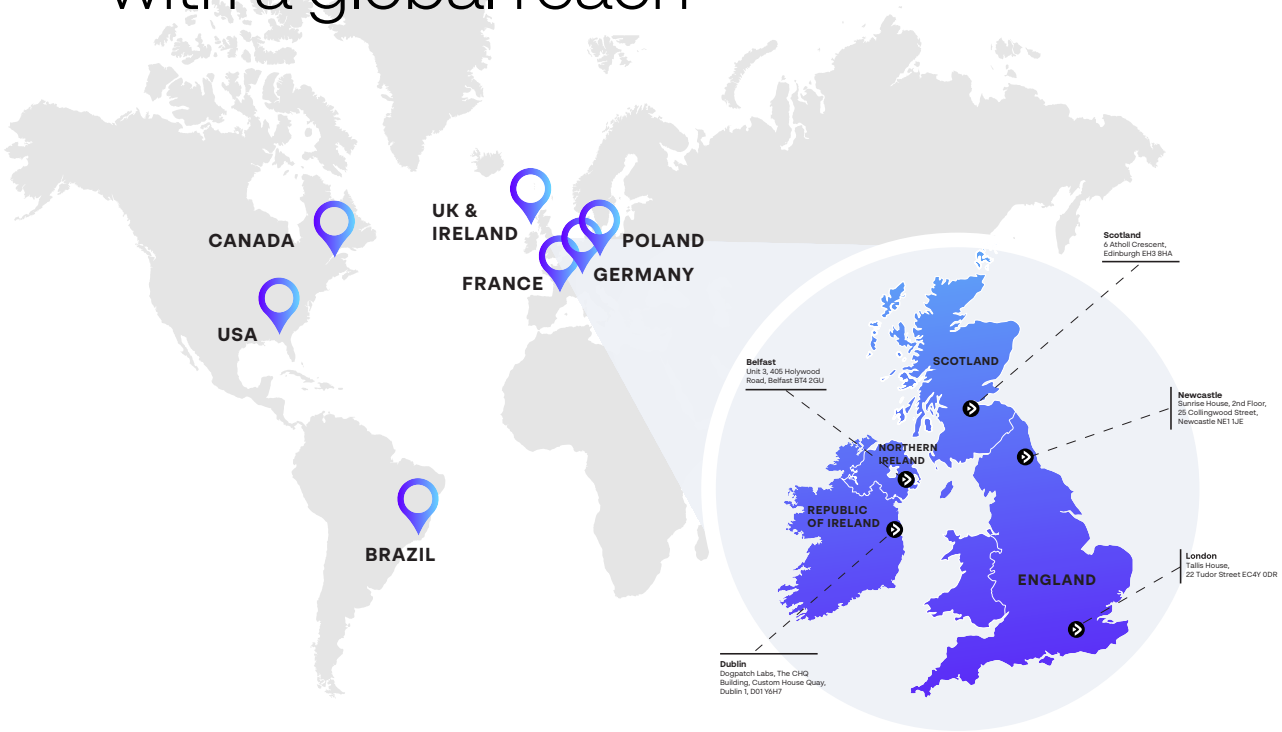
ABGi is a professional services and consulting business, dedicated to helping UK companies accelerate and fund their innovation strategies.

With a 35-year heritage and 520 experts & advisers, we manage £1.7bn of global tax incentives, grants and subsidies annually for a broad range of clients including some of the world's best-known brands.

Our aim is to help clients accelerate and fund their innovations. We operate on the entire value chain of innovation through our multidisciplinary international teams.

It pays to choose a provider with a proven track record – both in the UK and internationally – whose unique combination of highly qualified and experienced technical staff with proven methodologies consistently deliver great results for clients.

Local expertise with a global reach



info@abgi-uk.com
www.abgi-uk.com