

Beta This part of GOV.UK is being rebuilt – [find out what beta means \(/help/beta\)](#)

HMRC internal manual

Corporate Intangibles Research and Development Manual

From: [HM Revenue & Customs](#)
([/government/organisations/hm-revenue-customs](#))

Published 11 March 2016

Updated: 10 September 2025 - [See all updates](#)

[Back to contents](#) > [CIRD80000](#) > [CIRD81000](#)

CIRD81960 - R&D tax relief: conditions to be satisfied: BIS Guidelines (formerly DTI Guidelines) (2004) - application to software

Introduction

The full criteria for R&D to be qualifying for tax purposes are defined by the BEIS (now the Department for Science, Innovation and Technology- DSIT) R&D guidelines. The guidelines which apply for accounting periods beginning on

or before 31 March 2023 can be found at [CIRD81900 \(https://www.gov.uk/hmrc-internal-manuals/corporate-intangibles-research-and-development-manual/cird81900\)](https://www.gov.uk/hmrc-internal-manuals/corporate-intangibles-research-and-development-manual/cird81900) and those for later periods can be found at CIRD81910. These guidelines apply equally to any branch or field of science or technology. Software development is a part of computer science and information technology and falls within these areas described in BEIS guidelines (para 15-18). The references to paragraph numbers throughout this page refer to the paragraphs in the BEIS Guidelines in [CIRD81900 \(https://www.gov.uk/hmrc-internal-manuals/corporate-intangibles-research-and-development-manual/cird81900\)](https://www.gov.uk/hmrc-internal-manuals/corporate-intangibles-research-and-development-manual/cird81900).

[CIRD81300 \(https://www.gov.uk/hmrc-internal-manuals/corporate-intangibles-research-and-development-manual/cird81300\)](https://www.gov.uk/hmrc-internal-manuals/corporate-intangibles-research-and-development-manual/cird81300) states that in some fields of science and technology there are particular difficulties in determining whether a project is R&D for tax purposes. The purpose of this additional Guidance page is to apply the concepts of scientific or technological advances, technological uncertainties and project boundaries to software projects. This page does not change the BEIS Guidelines, but is intended to link scenarios common within the software sector to paragraphs within those guidelines in order to address common questions relating to software projects. There are no differences in the categories of expenditure which qualify for R&D tax credits which are set out in [CIRD81500 \(https://www.gov.uk/hmrc-internal-manuals/corporate-intangibles-research-and-development-manual/cird81500\)](https://www.gov.uk/hmrc-internal-manuals/corporate-intangibles-research-and-development-manual/cird81500).

Software, as with all information technology, is continually changing and developing with a number of well-established branches of software that continue to evolve (e.g. artificial intelligence, cloud or mobile computing), as well as new applications for software being developed (e.g. software robots, augmented reality and internet of things, and others yet to come). This page has also been updated to reflect changes in technology since the last revision.

An advance or appreciable improvement in science or technology means an advance in overall knowledge or capability in a field of science or technology, not a company's own state of knowledge or capability alone (paras 6, 9 and 23). Whether an advance is in overall knowledge or capability rather than a company's own can sometimes be difficult to ascertain in such a fast moving sector. R&D projects which were seeking such advances in previous years and were considered to meet the definition within the BEIS Guidelines may not qualify now because technological capability has increased over time, with solutions to what were previously technological uncertainties becoming available, or readily deducible, as the software industry moves forward to address new ideas.

The words 'research and development' are frequently used in the software sector, and care should be taken not to assume that a commercial project fully aligns with the definition of R&D set out in the BEIS Guidelines. It may do, but in order to be included as a qualifying R&D project for the purposes of the tax reliefs, only those parts of the commercial project that meet the criteria that are described in the BEIS Guidelines at [CIRD81900 \(https://www.gov.uk/hmrc-internal-manuals/corporate-intangibles-research-and-development-manual/cird81900\)](https://www.gov.uk/hmrc-internal-manuals/corporate-intangibles-research-and-development-manual/cird81900) will be qualifying.

Because of the pace of change within the software industry, HMRC caseworkers will be supported by HMRC's own computer specialists in this respect to ensure that knowledge of the industry is kept updated. The computer specialists provide advice based on their knowledge of the software industry. They do not claim to be competent professionals in any company's particular R&D project and whilst they offer advice, the HMRC caseworker will make the decisions in relation to an R&D claim.

Within an enquiry, if there are differences of opinion between competent professionals working in the field (specialist area) of the company's R&D project (para 19) and HMRC case workers or software specialists, the competent professional

may be asked to provide further clarifications with reference to information in public domain that can help better articulate the advance sought and technological uncertainties faced to support the explanation of what prevented these from being readily deduced.

Qualifying R&D software

There are two ways in which expenditure on the creation of software can be R&D within the BEIS guidelines.

1. Paragraph 27(a): activities within an R&D project to create or adapt software, materials or equipment needed to resolve the scientific or technological uncertainty, provided that the software, material or equipment is created or adapted solely for use in R&D

Where software is created or adapted solely as a tool for direct use in a larger R&D project, then development of the software will qualify as part of that R&D project. The software need not in isolation seek a specific advance in computer science or information technology – the R&D project may be seeking an advance in another field of science or technology but if the software directly contributes to that larger R&D project it will qualify as R&D (para 4) and if the software development is one of the R&D projects within a larger commercial project that collectively serve to resolve the scientific or technological uncertainty associated with achieving the advance in the overall project then the software R&D project will qualify (para19). An example might be data handling software needed to record and monitor the results of R&D undertaken by a Life Science company.

2. Development of software as the goal of the R&D project

The BEIS guidelines ([CIRD81900](https://www.gov.uk/hmrc-internal-manuals/cird81900) (<https://www.gov.uk/hmrc-internal-manuals/corporate-intangibles-research-and-development-manual/cird81900>)) apply to a software project in the same way as they apply to any other R&D project.

It does not matter whether the software project is intended to result in a product to be used in-house, licensed or sold.

The definition of R&D for tax purposes is in BEIS Guidelines at [CIRD81900 \(https://www.gov.uk/hmrc-internal-manuals/corporate-intangibles-research-and-development-manual/cird81900\)](https://www.gov.uk/hmrc-internal-manuals/corporate-intangibles-research-and-development-manual/cird81900)

R&D starts when work to resolve the scientific or technological uncertainties start and it ends when these uncertainties are resolved or work to resolve them ceases. (paras 33, 34).

Certain qualifying indirect activities related to the project can also be R&D. Activities other than qualifying indirect activities which do not directly contribute to the resolution of the project's scientific or technological uncertainty are not R&D. (paras 5 and 31)

The three key concepts to consider (the advance, the technological uncertainty and the boundaries of R&D) all need to be considered when identifying any R&D project.

The sections below apply these concepts to software development projects and paragraphs 6 and 13 of the BEIS Guidelines apply throughout this section, in addition to the others noted against specific points.

First: What advance in science or technology is this project seeking to achieve?

An advance in science or technology means an advance in overall knowledge or capability in a field of science or technology, not a company's own state of knowledge or capability alone. The advance may also be an appreciable improvement to an application of software provided these changes or adaptations to the scientific or technological characteristics would be generally acknowledged by a competent professional as a genuine non-trivial improvement (BEIS guidelines paragraphs 9 and 23). There may be more than

one advance and more than one R&D project within a larger commercial project.

A claim which states the advance as being in a certain area or field of software is not specific enough. For example, if the advance sought is in a specific sub-area such as robotic process automation, geoinformatics, augmented reality, artificial intelligence, microservices or computer vision etc., the competent professional for the company's R&D project should be able to explain how the company's R&D project is new in, or is an appreciable improvement to, that field of science or technology relative to that which is available in the public domain (para 21) or was readily deducible. The advances sought in the R&D project(s) could well be separate from the aims of the company's commercial project.

Examples of questions to be considered when considering how to set out the nature of the advance(s) or appreciable enhancements being sought:

a) What is the baseline in technology that any advance sought is being measured against?

It is the underlying technology in which an advance is made rather than the commercial output or outcome. For example, a project involving customisation of existing software which materially affects underlying science or technology can be R&D and so can collaborative work on commercial off the shelf (COTS) products, as long as the technological advance sought seeks to enable the modification of the product substantially beyond existing capabilities, representing the advance or appreciable improvement in science or technology required by the BEIS Guidelines (para 23).

Another example of how an advance or appreciable improvement can be identified is to measure and compare it relative to a comparable family (or category) of software that is viewed as representing the current state of overall capability

in that area (e.g. Proprietary or open source software frameworks)

It is unlikely that customisation such as configuring existing software to a company's own requirements would be qualifying R&D because it is generally already within the capability of the existing software.

b) What was the gap in technological knowledge or capability which necessitated the commencement of the R&D project?

Recognising what technological input is being introduced can often be a key measure of where the technological advance lies, rather than describing what different commercial functionality or output was achieved. Routine adaptation of an existing product or process is not R&D (para 12).

Combining standard technologies, such as integrating platforms, can be R&D if a competent professional in the field can't readily deduce how the separate components should be combined to have the intended function (para 30).

The implementation of a novel algorithm that represents a significant increase in overall capability in the area of software can also be R&D (para 23).

c) What technological changes have been made in seeking a technological advance, or attempted advance?

Making an appreciable improvement to an existing process, material, device, product or service through scientific or technological changes, so it can be used in a new or appreciably improved way will be R&D (paras 9 and 23).

The development of a software product does not represent an advance in science or technology simply because it uses software technology in its creation (para 8). Improvements, optimizations and fine tuning would only be R&D if there has

been a genuine and non-trivial improvement in the underlying technology (para 23).

Second: What are the scientific or technological uncertainties?

What matters is the technological input, rather than the commercial output. To help ascertain the technological uncertainties, consider the following:

a) Identify the challenges accurately: Is the company addressing the feasibility of doing what it needed to do, or, is the issue a matter of practical application, (Either could be R&D, the purpose is to identify where the technological uncertainty lies)?

b) What was it about the technology/computer science that made it uncertain whether software could be made to do what the company wanted it to do? What would be the typical approach that would be applied to resolve each particular technological uncertainty? If something is being claimed as a technological uncertainty, then usual methods (readily deducible) will not work/have worked and an alternative approach is being attempted (para 20, 21, 22).

For example:

System uncertainty arises because of the complexity of the entire system rather than how its individual components behave (para 14). What prevented such routine or established methods to be applied in the specific case?

As System uncertainty results from the complexity of the entire system rather than uncertainty about how the individual components behave, assembling components of a program to an established pattern or using routine solutions for doing so is not R&D (para 29). Simply specifying that there was a large number of components to assemble or integrate is a general statement and does not give enough information. The company should clarify why the relevant routine or

established assembly methods would not resolve the technical uncertainty (para 29).

Further examples of technological uncertainties could be

- Developing new or improved data architectures that cannot be achieved with readily deducible solutions, e.g. pushing beyond the boundaries of existing readily available database engines.
- Extending software frameworks (e.g. software development kits, or software libraries) beyond their original design, where knowledge how to extend these was not available or readily deducible at the time.
- Attempting to partially or fully solve a technological uncertainty that is documented as a known subject of research by computer scientists (e.g. there are relevant and contemporaneous research papers on that specific scientific or technological issue).

Third: Project boundaries (BEIS Guidelines paragraphs 33-35)

Most projects for the development of a commercial product will go further than resolving technological uncertainties and so will not qualify as R&D in their entirety so it is important to identify the projects within the commercial project that do qualify as R&D (para 19)

R&D begins when work to resolve the scientific or technological uncertainty starts, and ends when that uncertainty is resolved or work to resolve it ceases. (Para 33 and 34). The company should identify the boundaries of the R&D project which should encompass all activities that collectively serve to resolve scientific or technological uncertainty associated with achieving the advance (para 19, 36). Software development project(s) may need to be identified as smaller R&D projects within a larger commercial project. Alternatively, it may also mean merging a number of separate workstreams that collectively serve to achieve the scientific or technological advance. The cycle of

activities within a software R&D project can be a useful way of determining the types of activities that qualify.

It is not enough to state that a product is not available to solve the (commercial) problem faced as being the reason why a solution wasn't readily deducible, but this could be the trigger that starts the software development project, or sub projects within a larger project, which in turn may trigger the commencement of qualifying R&D. Breaking the larger problem down and identifying where solutions are known or unknown, can establish where the start of the qualifying R&D lies.

Common features in a software project

Requirement gathering for technological uncertainties

Expenditure on requirement gathering activities where no technological questions are at issue is not R&D and therefore must be excluded (paragraph 33). Technical requirement analysis where specific technological questions are related to resolving scientific or technological uncertainties would qualify for R&D.

In software development this generally means that business requirement gathering typically undertaken by the subject matter experts of the business and business analysts would not qualify for R&D, nor would routine analysis of commercial rather than technological requirements. (paras 33 and 27c).

Planning

Planning activities such as outlining, estimating and scheduling and planning for the R&D elements of the project is R&D, however planning associated with non-R&D elements of the project such as financial, marketing and legal aspects is not R&D (paras 27b, 36 and 37).

Analysis, design and development

The technical analysis, design, development and testing activities that are needed to directly resolve the scientific or technological uncertainties are R&D and may be included, however analysis, design, development and testing that do not contribute to resolving scientific or technical uncertainties are not R&D and need to be excluded (paras 4, 27 c, 41).

Testing

There will inevitably be testing continuously throughout the entire project. It may need to be recurrent or it may resolve issues more quickly, but it is the purpose which needs to be considered. To be included in an R&D project the purpose of the testing work should be to feed back into the development, not to validate that it definitely works properly once the technological uncertainties have been resolved (para26, 29, 34 and example G1).

Non-functional testing

Software testing that happens early on such as unit testing which test smallest units of the software code, or early development integration tests that happen in software development computer environments are likely to feedback to the development required to solve scientific or technological uncertainties and so can be included as R&D.

Other non-functional tests that happen throughout the development such as system integration testing may also be required to solve system uncertainties and could be included as R&D. Similarly performance and security testing may contribute to resolving scientific or technological uncertainty provided these tests are not merely confirmatory (as can be the case with some confirmatory security testing).

However when non-functional tests are performed for confirmatory reasons and do not feedback to the design or development stages of R&D, these should be excluded.

Functional testing

Other types of testing such as functional user acceptance testing which may focus in testing the functionality (rather than underlying issues which are resolved) or focus on the look and feel of the solution which do not contribute to R&D should be excluded (para 42).

However, in some cases user acceptance testing may feed back to development toward the resolution of technological uncertainty, so in such cases these activities may be included.

Deployment

Deployment or release activities that transfer software to production systems generally happen after the uncertainty is resolved and in themselves do not contribute to resolving scientific or technological uncertainties and therefore should be excluded.

Maintenance

Maintenance activities or minor fault fixing where no technological uncertainties arise are not R&D and therefore should be excluded. However, during maintenance new problems may emerge that may require R&D to recommence through the life-cycle stages of the software development (para 35).

Software as capital expenditure

It is possible that a software project, despite being an R&D project, will not qualify for R&D tax relief because the expenditure is capital for tax purposes. This is dealt with at [CIRD81700](#)

<https://www.gov.uk/hmrc-internal-manuals/corporate-intangibles-research-and-development-manual/cird81700>).

← **Previous page**

(/hmrc-internal-manuals/corporate-intangibles-research-and-development-manual/cird81920)

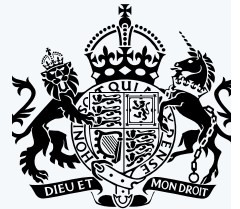
→ **Next page**

(/hmrc-internal-manuals/corporate-intangibles-research-and-development-manual/cird81980)



OGL

All content is available under the Open Government Licence v3.0, except where otherwise stated



© Crown copyright