



Neutral Citation Number: [2023] EWHC 581 (Pat)

Case No: HP-2019-000043

IN THE HIGH COURT OF JUSTICE
BUSINESS AND PROPERTY COURTS
INTELLECTUAL PROPERTY LIST (ChD)

Rolls Building
Fetter Lane
London EC4A 1NL

Date: 22 March 2023

Before:

MR JUSTICE ZACAROLI

Between:

(1) PQ SYSTEMS EUROPE LTD
(2) PRODUCTIVITY-QUALITY SYSTEMS,
INC.

Claimants

- and -

(1) JEFF AUGHTON
(2) FACTORIA LTD

Defendants

Mr Jonathan Hill and Mr Joshua Marshall (instructed by **MFG Solicitors LLP**) for the
Claimants
Mr Jeff Aughton, the **First Defendant**, represented himself and the **Second Defendant**

Hearing dates: 8, 9, 10 and 13 February 2023

JUDGMENT

Mr Justice Zacaroli:

1. Productivity-Quality Systems, Inc (“PQS”), the second claimant, a company incorporated in the United States, is in the business of computer software development to assist with quality assurance in manufacturing. The first claimant, PQ Systems Europe Ltd (“PQE”), an English company, was until it ceased trading in about 2015, in the same business.
2. Until 2014 PQS and PQE were in the common ownership of their founder, Mike Cleary (“Mr Cleary”) and his family. After Mr Cleary died, in 2014, they were under the ownership of Beth Savage (“Ms Savage”). In 2022 PQS was sold, but PQE remains under Ms Savage’s ownership. I will refer to the claimants together as “PQ”.
3. The first defendant, Mr Jeff Aughton (“Mr Aughton”) has been developing software for over 50 years. He was employed by PQE from 1989 as a software developer. On 1 January 1996 he was appointed a director of PQE, but resigned from that position on 18 April 2013. He resigned from PQE on 15 May 2015.
4. The second defendant, Factoria Limited (“Factoria”), is a company incorporated on 19 May 2015 by Mr Aughton and his wife.
5. This judgment follows a trial on issues of liability. At the heart of the case is the question whether software written by Mr Aughton after he left PQE was copied from software, the copyright and confidential information in which is owned by the claimants.

The facts in summary

6. PQ’s principal operations are in the United States. Within PQE, Mr Aughton accepted that he was the “senior partner”. He had a substantial degree of autonomy, but reported to personnel in the US, mainly Mr Steve Daum, currently the Director of Software Engineering at PQS.
7. PQ has developed, marketed and sold over many years statistical process control (“SPC”) software and gauge management software.
8. SPC is an analytical technique for improving industrial processes. Data produced by gauges is monitored, for example by producing charts utilising statistical analysis methods. PQ’s principal SPC product was SQCpack, a full suite of SPC software. This incorporated a chart generating program called CHARTrunner.
9. Gauge management software assists manufacturers monitor gauges and other instruments, enabling them to be recalibrated at intervals. It produces detailed records in the form of grids and specialist charts. Shortly after joining PQE, Mr Aughton began working on PQ’s gauge management software, called GAGEpack.
10. In about 2006 or 2007 PQ decided to re-write SQCpack and CHARTrunner. Mr Aughton was given the task of developing a charting module, written in a

programming language known as VB.NET, called CHARTcore, or sometimes PQChartCore. This was a library of routines to take pre-computed data and information and display it as graphs on a screen or printer. Another of PQ's developers produced an alternative version, written in Windows Presentation Foundation ("WPF"). After Mr Aughton had mostly completed the work on his section, PQ decided to go forward only with the WPF version. Mr Aughton considered this to be a serious mistake and requested, and was granted, permission to leave that project and to continue working on GAGEpack, which he subsequently re-wrote in VB.NET.

11. Mr Aughton also carried on, however, developing an SPC program in VB.NET. The precise timing of its development is unclear. In evidence Mr Aughton said that he spent about 50 working days of his own time on it, over the course of 2007-2008, then set it aside before picking it up again in about 2012 when he shared it with a Mr David Todd, a former employee of PQ who had been fired in 2010. In his written closing submissions he said that he worked on it from 2007 to 2010, later applying a few tweaks when sharing it with Mr Todd. He initially called this "CRJA" ("CR" for Chart Runner and "JA" for Jeff Aughton), but in about 2013 he renamed it "ProSPC".
12. Mr Aughton's conduct in 2012-2013 in sharing ProSPC with Mr Todd led to disciplinary proceedings against him. Mr Aughton disputes the details of what he said during this process, but it is common ground that it resulted in him being given a written warning and being demoted from managing director, although being allowed to stay on at PQE.
13. In May 2015, Mr Aughton resigned from PQ. There does not appear to have been anything particularly acrimonious about his departure, although his relationship with PQ had not been the same after the disciplinary proceedings in 2013. His last day working for PQE was 15 May 2015. He immediately started work on what he contends was an entirely new SPC program for creating charts by tapping into an existing data collection system, written in VB.NET. He initially called this qSPC, but later renamed it "InSPC". It had broadly the same functionality as PQ's CHARTrunner program. He also wrote a further version, called "InSPC+", which provides customers with a self-contained system, maintaining its own database, also written in VB.NET. It had broadly the same functionality as PQ's SQCpack program. He incorporated the company through which he subsequently sought to exploit these products, Factoria, on 29 May 2015. I will refer to these versions of InSPC and InSPC+ as "InSPC v1".
14. By September 2015, Mr Aughton had developed InSPC v1 to the point where he was prepared to demonstrate it to a potential commercial partner, CyberMetrics Corporation ("CyberMetrics"), a US company. Over the course of the next three years, Mr Aughton worked with CyberMetrics to develop the InSPC software.
15. At some point in 2017 or 2018 (the precise date is in dispute), Mr Aughton re-wrote InSPC and InSPC+ in a new language, C#. I will refer to these new versions as "InSPC v2".

16. PQ became aware of InSPC v1 when it was promoted in demo form by CyberMetrics in the summer of 2017. Upon reviewing it they saw not only significant functional similarities with their own products, but also discovered references to Factoria and PQ as author (albeit that these references are not now relied on as part of their claim). On 18 October 2017, PQ's solicitors wrote to Mr Aughton alleging breaches of confidence and infringement of copyright. Shortly afterwards, CyberMetrics removed InSPC v1 from their website.
17. PQS filed proceedings against CyberMetrics in the US on 18 October 2017. Mr Aughton was deposed in those proceedings in October 2021. They were subsequently settled on confidential terms.
18. These proceedings were commenced by a claim form issued on 25 September 2019.
19. The principal claim is that Mr Aughton copied or otherwise made use of software owned by PQ when he wrote InSPC v1 and InSPC v2. PQ contends, in brief, that: the copyright in ProSPC was owned by PQ; Mr Aughton copied from ProSPC when writing InSPC v1; and that he then copied it again (most likely by copying from InSPC v1) when writing InSPC v2, either by use of automatic translation software, or manually, or a combination of the two.
20. Mr Aughton denies this, and contends that: the copyright in ProSPC belonged to him because it was written as a hobby project outside the course of his employment; in any event he did not copy from ProSPC when writing InSPC v1; and he did not copy from InSPC v1 when writing InSPC v2, whether by automatic translation or otherwise.
21. In most cases, questions of copying can be resolved relatively easily by comparing source code (i.e. the code written by the programmer using human-readable language and symbols). In this case, however, that is not possible. There is no source code available for ProSPC or for InSPC v1. Mr Aughton has deleted them. Only the object code (i.e. the code written in computer readable language, which is produced by running source code through a compiler) is available for the two programs.
22. The claimants have therefore sought to de-compile the object code for ProSPC and for InSPC v1, by the use of commercially available de-compiling software. It is common ground that de-compiled code will look very different to the original source code. None of the comments or other extraneous language remains. Moreover, as Mr Aughton demonstrated – albeit by reference to two pieces of code he specifically designed for this purpose – it is possible for two quite different looking pieces of source code, once compiled and then de-compiled, to look exactly the same. I will address the expert evidence in more detail below, but the view of the experts is that when investigating whether copying has occurred the value in comparing decompiled source codes is “limited” (per PQ’s expert) or “marginal at best” (per Mr Aughton’s expert).
23. The position is made worse in relation to InSPC v1 by the fact that the compiled code was also “obfuscated”, and that de-obfuscation does not restore names to their original values, but assigns new names.

24. Although the source code for InSPC v2 remains, since it is written in C# rather than VB.NET it has different syntax, making comparison with a decompiled source code of ProSPC that much more difficult. Mr Aughton's expert's view was that similarities which might appear between lines of C# code in InSPC v2 and the decompiled code of ProSPC provide no more than "supposition" that there may have been copying.
25. Accordingly, while PQ's case that copying has occurred does rest in part on expert comparison of decompiled code, it is based heavily on circumstantial evidence.

The witnesses

26. For the claimants, evidence was given by Ms Elizabeth ("Beth") Savage, the President and owner of PQS and a director of PQE, and by Mr Steve Daum, the Director of Software Engineering at PQS. They each provided witness statements and were cross-examined by Mr Aughton. On the critical questions (i.e. whether Mr Aughton wrote ProSPC in the course of his employment, and whether he copied from ProSPC in writing InSPC v1 and InSPC v2) their evidence is of only peripheral relevance. I address specific points in their evidence where relevant in the course of this judgment. Overall I consider that both of them sought to give honest evidence in order to assist the court.
27. Mr Aughton, who conducted the trial as a litigant in person, provided a witness statement on which he was cross-examined. On a number of key points, for the reasons I set out below when dealing with those key points, I found his evidence to be unreliable.
28. Both parties called evidence from an expert in computing: Dr Nigel Young for the claimants, and Mr David Dufour for the defendants. Both are acknowledged to be experts in the field.
29. Dr Young gave his evidence in a measured and thoughtful manner. He was ready to acknowledge points of weakness in his analysis, and points against PQ's interests. His evidence was balanced and credible.
30. In contrast, I found Mr Dufour's evidence on key points to be unconvincing. His report did not comply with the formal requirements of an expert report, including the requirement to state that he had read, understood and complied with CPR Part 35 and the guidance for experts. As its heading indicates ("Response to the Statement of Case on Copying"), it takes an adversarial approach, rather than identifying points for and against a particular conclusion. I deal in more detail below with the weaknesses in his evidence, when addressing the key issues to which it relates.

The issues

31. The key issues are:

- (1) are the copyright and confidential information in ProSPC owned by PQ?

- (2) was InSPC v1 copied from ProSPC, so as to infringe the copyright in ProSPC or misuse any confidential information in ProSPC? and
- (3) was InSPC v2 copied from ProSPC, either directly or – more likely – indirectly via InSPC v1?
32. The case also raises further, peripheral, issues relating to Mr Aughton’s duties under his employment contract. So far as relevant, I deal with these in the course of considering the key issues.

The law

33. There is no dispute in this case as to the applicable legal principles. Copyright subsists in both source code and object code and such parts of the design or structure of a computer program that are indicative of the creativity and skill of the author. To the extent that the code is dictated by technical function (i.e. where there are essentially no choices to be made by the coder as to how to express the code) it is not protected. Nor is the functionality of a computer program, nor the ideas that lie behind it, protected. See generally for these propositions, Copinger and Skone James on Copyright, 18th ed, at §3-88 to §3-93; and *SAS Institute Inc v World Programming Ltd* [2013] EWCA Civ 1482 at §20-§37.
34. The copying of a computer program, or of a substantial part of it, amounts to infringement (s.16 and s.17 of the Copyright, Designs and Patents Act 1988 (“CDPA”)), as does making an arrangement, altered version or translation of it (s.21(4) of CDPA).
35. Where there is a substantial similarity between the original work and the allegedly infringing work, coupled with proof of the possibility of access, then this raises a prima facie inference of copying, which (as much as a matter of plain rational thought as a proposition of law) it is for the defendant to answer: Copinger (above), at §7-26; *IBCOS Computers Ltd v Barclays Finance Ltd* [1994] FSR 275, p.296-297. As Jacob J there noted,
- “at this stage (namely “was there copying?”) both the important and the unimportant bits of the works being compared count. Indeed it is often identity of trivial matter which traps a copyist. As Hoffmann J observed in *Billhöfer Maschinenfabrik GmbH v Dixon & Co Ltd* [[1990] FSR 105 at 123]:
- “It is the resemblances in *inessentials*, the small, redundant, even mistaken elements of the copyright work which carry the greatest weight. This is because they are least likely to have been the result of independent design.”
36. An example, with a close parallel in this case, was provided by Pumfrey J in *Cantor Fitzgerald International v Tradition (UK) Ltd* [2000] RPC 95, at p.193. This concerned the ordering of a list of variables in a piece of code. Where the order in which they appear is irrelevant, and a matter of arbitrary choice, then

the reappearance of that order in a later code is “certainly a powerful indication” of copying.

37. There is also no dispute in this case that computer code in PQ’s products is protected as confidential information. Mr Aughton’s contract at the time he left PQE defined “Confidential Information” as:

“information (whether or not recorded in documentary form, or stored on any magnetic or optical disk or memory) relating to the business, products, Company software and software development, affairs and finances of the Company for the time being confidential to the Company and together with information relating to the customers, suppliers, agents of the Company and trade secrets including, without limitation, technical data and know-how relating to the business of the Company or any of the business contacts.”

38. By clause 13.2, Mr Aughton was prevented at any time (whether during his employment or afterwards) from using or disclosing to any person any Confidential Information, except in the proper course of his duties or where use or disclosure was authorised by PQ or required by law, where the information is already in the public domain or where it was protected within the meaning of s.43A of the Employment Rights Act 1996. None of these exceptions applies to anything done by Mr Aughton after he left PQ in 2015.

39. The use of source code, even if only as a reminder when writing competing software, would amount to a breach of confidence: *Cantor Fitzgerald* (above) at §87.

40. Given the clear contractual restriction on Mr Aughton’s use of confidential information, it is unnecessary to consider the alternative way PQ put their claim, based on the equitable duty of confidence.

(1) Are the copyright and confidential information in ProSPC owned by PQ?

41. Mr Aughton’s case is that he wrote ProSPC from scratch, as a hobby project in his spare time, without reproducing any source code from any of PQ’s software. He says he wrote it on his personal home computer.

42. PQ disputes this, and contends that ProSPC was written in the course of Mr Aughton’s employment.

43. By s.11(2) of the CDPA:

“Where a literary, dramatic, musical or artistic work, or a film, is made by an employee in the course of his employment, his employer is the first owner of any copyright in the work subject to any agreement to the contrary.”

44. This involves an analysis of two questions, which often merge into one (Copinger, at §5-18):

“(a) whether the work which was done was the kind of work which the employee was engaged to do (i.e. whether it was within the scope of the employment) and, if it was, (b) whether the work was in fact done in the course of that employment at all.”

45. There is no doubt as to the answer to the first of these questions. ProSPC was a piece of software that was precisely of a kind which Mr Aughton was engaged to do. On Mr Aughton’s own admission, ProSPC was his own version of PQ’s CHARTrunner software, but written in VB.NET, which others in PQ had been charged with taking forward in WPF.
46. The second question involves a multifactorial assessment based on all the circumstances of the case: *MEI Fields Designs Ltd v Saffron Cards and Gifts Ltd* [2018] EWHC 132 (IPEC), where David Stone (sitting as an Enterprise Judge) identified (at §42) relevant factors as including the following:
 - “(a) the terms of the contract of employment;
 - (b) where the work was created;
 - (c) whether the work was created during normal office hours;
 - (d) who provided the materials for the work to be created;
 - (e) the level of direction provided to the author;
 - (f) whether the author can refuse to create the work/s; and
 - (g) whether the work is “integral” to the business.”
47. A useful starting point is to consider admissions on two key points made by Mr Aughton in April 2013, the background to which is as follows.
48. In late 2010, PQ discovered that Mr Todd had set up a company planning to market software in competition with PQ, and had stolen confidential information from PQ in order to do so. Following an internal investigation, Mr Todd was fired.
49. On 28 March 2013, PQ received an unsolicited email from Mr Todd, attaching an executable and time-bombed version of ProSPC, saying: “Attached is the product [Mr Aughton] has been writing for the last 18 months. It expires on 31/03/2013 – what is the reason I am telling you this? because he played me – but maybe you already knew he was writing it?”
50. This prompted PQ to conduct an internal investigation. A formal disciplinary hearing took place on 18 April 2013, attended by Ms Savage on behalf of PQ and an independent HR consultant, Andrea Palmer. Ms Palmer took detailed notes at the meeting. A detailed copy of the notes, as revised by Ms Savage, has been disclosed by PQ. Ms Savage’s evidence is that Ms Palmer provided her with a copy of the notes after the meeting, which she (Ms Savage) revised, where her recollections differed from Ms Palmer’s, before providing them to Mr Aughton.
51. The notes purport to record two important admissions by Mr Aughton. The first is that the routines he used in ProSPC were lifted from the CHARTcore routines that he wrote for PQ. The second is that, while he was the sole creator of

ProSPC, he acknowledged that he had written it in the course of his employment and that it belonged to PQ.

52. Mr Aughton contended that the hearing notes are a forgery, being neither contemporaneous nor accurate. He referred to metadata in a pdf version of the notes created six days later, which refers to Ms Savage as its author. This, he says, shows that the reference in the letter before action to the notes being contemporaneous was false, and that Ms Savage was not telling the truth when she said in her witness statement that a copy of the notes was made available to Mr Aughton “at the time of the disciplinary hearing”.
53. The reference to the metadata is an irrelevance. It is the metadata of the pdf document, not the original word version, and both the date of creation and name of author are most likely explained by the fact that it was Ms Savage who first saved it as a pdf. The fact that this was six days after the meeting is consistent with Ms Palmer having made typed notes during the meeting and tidying them up before sending them on to Ms Savage, who made her own revisions before saving it as a pdf which could be sent to Mr Aughton. That does not deprive the notes of the quality of being contemporaneous.
54. There is in evidence a written final written warning letter addressed to Mr Aughton and signed by Ms Savage, which Mr Aughton accepted he received at the time. This letter purported to attach a copy of the notes of the disciplinary hearing taken by Ms Palmer. In my judgment, it is most likely that a copy of the hearing notes was indeed sent to Mr Aughton with this letter. There is no evidence of Mr Aughton having gone back to Ms Savage at the time to complain that the promised hearing notes were missing. This letter corroborates Ms Savage’s evidence that they were made available to Mr Aughton at “the time of” the disciplinary hearing. That clearly did not mean the day of the hearing (given that it was Ms Savage’s evidence that the notes were later revised by her before being sent to Mr Aughton), but was used in the sense of ‘around the time of’ the hearing, as Ms Savage confirmed in her evidence.
55. Mr Aughton also contended that the hearing notes were clearly false and had been altered so as to cheat him out of salary and pension entitlements, whereas he had been assured at the hearing that there would be no change to either of these. He relied on bank statements which he said showed reductions in the amount he received each month for salary, and reductions in the employee contributions to his pension, after the disciplinary hearing. The problem with this contention, however, is that he agrees that he was given, at the hearing, a copy of his revised written contract. That contract stated in clear terms both his salary and pension entitlements, which had not changed.
56. So far as concerns the passages in the hearing notes which purport to record what Mr Aughton said as to the provenance of the source code used in ProSPC, these in fact repeat what he told Mr Cleary in an email on 11 April 2013. In this email he said that ProSPC does what CHARTrunner does, but was never intended to compete with it, “... just stimulate some ideas”, and then:

“When GAGEpack was rewritten for .NET I reused the old PqChartCore routines for it, enhancing them where appropriate.

Further, when I started CRJA/ProSPC I just lifted them again, making the necessary changes. Essentially the bulk of CRJA/ProSPC code and all of GAGEpack’s charting are those PqChartCore routines.”

57. Faced with this email, in cross-examination Mr Aughton accepted that he probably would have said what the hearing notes record him as saying as to the reuse of PqChartCore routines. I find that he did. The hearing notes are more than a summary of points made at the hearing, but record the back and forth of the discussion between Ms Savage and Mr Aughton. They make express reference to what Mr Aughton had said in the email, and record him having reiterated the key point made in it (as to ProSPC and GAGEpack being derived from PQ’s CHARTcore software) on at least three occasions. There would have been no reason – given that the outcome of the hearing was that Mr Aughton was kept on – for either Ms Palmer or Ms Savage to record in the notes anything other than their honest recollection of what was said. Accordingly, I find that the notes are, in this respect at least, an accurate reflection of what Mr Aughton said during the disciplinary hearing.
58. The second admission recorded in the hearing notes is in the following passage:
- “[Mr Aughton] stated that the code that exists in ProSPC is his code. He clarified that it is the Company’s code because he wrote it for PQ systems while employed, but it is code that he wrote, not code written by other PQ developers.”
59. Mr Aughton denies saying this, but I reject his evidence on this point too. This is not the only reference to the point in the notes. Shortly afterwards, Mr Aughton is recorded as saying that he had done something really stupid but that “all the code he wrote was his own”, but when corrected by Ms Savage that it belonged to the company, he said “yes, the Company’s but that he wrote it, and almost all of the code for that was in TFS [PQ’s software repository]...” It is a theme of Mr Aughton’s evidence that he feels a strong sense of ownership of the software that he wrote for PQ. He is scathing about the re-write of SQCPack undertaken in WPF by others, as compared to his own re-write of GAGEpack which he said was extremely successful. I consider that he is inherently likely to have made the point in the disciplinary hearing that the software was his own (in the sense that it was all his own creation), but that he did clarify that this meant he had written it, not that he claimed ownership of it as against PQ.
60. Indeed, given that he had just told PQ that he lifted routines from software belonging to PQ when writing ProSPC, if he had then claimed that he nevertheless owned the intellectual property in ProSPC, it is hardly likely that PQ would have agreed to keep him on as they did.
61. The accuracy of the hearing notes in these key respects is reinforced by the fact that no objection was made by Mr Aughton to them. He maintains that he was not provided with a copy of the notes. As I have already noted, however, I consider it likely that he was sent a copy, attached to the final written warning. More importantly, both of these key admissions were repeated in the final written warning letter, which he accepts he did get.

62. Mr Aughton now says, however, that what he told Mr Cleary, and repeated at the disciplinary hearing, as to having lifted code from PqChartCore when writing ProSPC and GAGEpack, and as to PQ owning the software, was not true. When asked why he would have lied to PQ in his email to Mr Cleary of 11 April 2013 he said:
- “Because I was under threat of losing my job unless I gave him some story and this is the one he really liked. Yes, I have got your code, you know, it is a fair cop, and he was happy with that.”
63. This evidence, given for the first time during cross-examination, makes no sense. This was a very serious moment for him. He faced losing his job, his salary and his pension. If the truth was that he had written ProSPC entirely from scratch, as his own personal hobby project without reference to any of PQ’s source code, then by admitting that he had used PQ’s own confidential source code to create a product which he shared with Mr Todd was more, not less, likely to lead to him losing those things.
64. The suggestion that he was simply acting to placate Mr Cleary also sits uneasily with an episode during this same disciplinary process designed to have the opposite effect. When Mr Cleary attended his home to take delivery of Mr Aughton’s work computer containing the source code for ProSPC, Mr Aughton deliberately gave him a blank computer to drive away with.
65. The truth of the contention that both GAGEpack and ProSPC are derived from the same source code is supported by the fact that there are similarities between (1) the source code of GAGEpack and (2) the decompiled object code of ProSPC which can only be explained by one being copied from the other, or both being copied from the same source. During his cross-examination of Dr Young, Mr Aughton referred to a comparison between two “DotTypeEnums”, one in the GAGEpack source code, and the other in the decompiled ProSPC code. There were significant similarities which Dr Young considered indicated copying. Mr Aughton had previously denied this, but when questioning Dr Young he not only agreed that they were similar, but said that this was because one was indeed copied from the other. He maintained however, that it was because “the code from ProSPC was copied into GAGEpack”.
66. I note that if this is correct, then it reinforces the conclusion that ProSPC was written in the course of his employment, as it shows him using parts of the source code of ProSPC for the purposes of PQ’s business. I do not accept this explanation, however, which is inconsistent with the explanation he gave in his email of 11 April 2013, and in the disciplinary hearing which, for the reasons I have already given, I believe to be true. Mr Aughton suggested in his written closing argument that PQ (and I) had got this the wrong way round – because we assumed that the comparison of the DotTypeEnums code was intended to show that ProSPC was copied from GAGEpack. That, however, is not the point. The point is that the comparison demonstrates that code in both GAGEpack and ProSPC was derived from the same source, and this reinforced the accuracy of what Mr Aughton had told Mr Cleary in April 2013, that they both contained routines lifted from the source code he wrote for PqChartCore.

67. For the above reasons, I conclude that Mr Aughton did use source code from PqChartCore when writing ProSPC. The fact that he did so is in my judgment an important factor pointing towards ProSPC being written in the course of his employment. His admission made in 2013 that he did so in the course of his employment, such that the software was owned by PQ is also an important consideration, although not in itself conclusive (because this is at least in part a legal issue). I go on, therefore, to consider other relevant factors.
68. One factor which points strongly in that direction is the fact that, since writing SPC software was a core part of PQ's business, Mr Aughton's work in writing ProSPC was undoubtedly integral to that business.
69. There are no specific terms of Mr Aughton's contract of employment that assist either way, since until 2013 the contract appears to have been a very basic one. His position as "senior partner" with at least some autonomy over the way he worked and the projects he worked on, however, means that the fact he was not directed to work on ProSPC by anyone else at PQ does not point away from it being done in the course of his employment. While he was under the supervision of PQS personnel in the US, he had a substantial degree of autonomy, as shown by the fact that it was his initiative, for example, to produce a new version of GAGEpack each year, and to write specific interfaces for customers who needed them (despite resistance to that idea from the US).
70. As I have noted, Mr Aughton claims to have worked on ProSPC from home, in his own time (i.e. evenings and weekends). This is of little relevance, however, because between 2007 and 2012 (during at least part of which time he was working on ProSPC) there was no clear delineation between personal and work matters in either respect. For most of that time he worked from home and, while he appears to have been originally contracted (according to brief written terms from 1991) to work from 8:30am to 5:00pm, Monday to Friday, by 2007 (at least) he was accustomed to working in the evenings and at weekends on PQ matters. This was demonstrated by the check-in history on PQ's "TFS" server, which showed Mr Aughton working on PQ matters outside of office hours.
71. If ProSPC was ever to be put to commercial use, there were only two possibilities: either it would be used to compete with PQ or it would be used for the purposes of PQ's business. In his email of 11 April 2013 to Mr Cleary, Mr Aughton said that he had not intended ProSPC to compete with PQ's products, but that he had written it to "stimulate ideas". It is notable that he did not say, either in this email or (according to the hearing notes) during the disciplinary hearing, that he had written ProSPC as a hobby project.
72. In circumstances where he admitted at the time reusing PQ's code when writing ProSPC, I infer that what he meant by writing ProSPC to "stimulate ideas", was to do something that was at least potentially to be used in PQ's business. The fact that he utilised the same code in re-writing GAGEpack (whether that be, as I have concluded, using the same code from PqChartCore twice or, as he suggested at trial, using the code written for ProSPC when re-writing GAGEpack) reinforces the conclusion that the work he was doing was at the time intended to be of at least potential use to PQ.

73. This conclusion is also supported by the fact – as Mr Aughton himself said – that he was strongly opposed to CHARTrunner being re-written in WPF. I accept the submission made by Mr Hill that in these circumstances at least part of Mr Aughton’s motivation was to prove that he was right, and Mr Cleary was wrong, as to the best way to re-write CHARTrunner.
74. Separately, the fact that ProSPC was developed with the use of PQ’s code, coupled with the fact – which he accepts – that it was written using Visual Studio and other resources licensed by PQ, shows that PQ provided at least some of the “materials” that were used in its creation. This is itself a relevant factor in determining whether it was done in the course of his employment.
75. There is a dispute as to whether ProSPC was written by Mr Aughton on a computer he used for personal matters, or on a computer provided by PQ for his use when working at home. The most compelling evidence on this issue is the fact that ProSPC was on the work computer which Mr Aughton (eventually) handed over to Mr Cleary in April 2013. Mr Aughton says that he had copied ProSPC from his personal home computer to his home work computer solely for the purpose of handing the latter over to Mr Cleary. He says he did so to satisfy Mr Cleary’s demand that he hand over his work computer containing ProSPC. That, however, suffers from similar problems as his evidence as to why he initially handed Mr Cleary a blank computer. If, as he now claims, ProSPC was a purely personal hobby, then the fact that he had written it on his personal home computer would be more likely to demonstrate that fact, than if he had written it on his work computer. On balance, therefore, I consider it more likely that it was written on the work computer that he had at home.
76. Taking Mr Aughton’s admission made in 2013 that ProSPC belonged to PQ along with the other points set out above, in my judgment ProSPC was indeed written in the course of Mr Aughton’s employment such that the copyright and confidential information in it belonged to PQ.

(3) Was InSPC v1 copied from ProSPC

77. There is no doubt that Mr Aughton had access to ProSPC when he wrote InSPC v1. He accepts that he retained ProSPC when he left PQ, and that he had access to it throughout the time that he was writing InSPC v1. He maintains, however, that he made no reference to it at all when writing InSPC v1, which he started from scratch.
78. Mr Aughton relies on the fact that the experts agreed that: “there is no evidence of copying of substantial lengths of source code as by copy-and-paste or other methods of exact reproduction”. That does *not* mean, however, that there is evidence that substantial lengths of source code were not copied. This agreed statement is a product of the limitations inherent in comparing de-obfuscated and decompiled code.
79. The experts nevertheless agreed that, within those limitations, there is evidence of similarity of applications, in the capabilities and features of the programs, that the format and naming conventions of certain xml files used in ProSPC and

- in InSPC are highly similar, and that there are similarities in the structures of ProSPC and InSPC v1.
80. Mr Aughton maintains that the similarities are explained by the fact that both programs were written by the same experienced programmer who, over many years, had developed his own idiosyncratic style. Mr Dufour's view is that the similarities could be explained in that way.
 81. One example relied on by Dr Young points compellingly towards copying. PQ, in order to aid comparison, having decompiled the respective object codes for ProSPC and InSPC v1, split the decompiled code into enums (a list of data included in source code, often with corresponding reference numbers assigned to them), methods (defined procedures) and properties (values which can be obtained, or set).
 82. One enum, labelled "ChartTypeEnum" contained a list of chart types, each with an assigned reference number. It was common ground that, save for one glitch, the de-obfuscation and decompiling processes would not have changed the names or ordering of chart types or the assigned reference numbers. The one glitch is that some of the decimal numbers were changed to hexadecimals. That was accepted to be caused by decompiling. In fact it arose only when one particular decompiler was used (I explain below the use of two different compilers). Since the changes were the same in the decompiled code for both ProSPC and InSPC v1 it is not material.
 83. The list of chart types in the ProSPC ChartTypeEnum is almost precisely replicated in the InSPC v1 ChartTypeEnum, albeit the latter contains a number of additional chart types. The chart names (save in one respect) and ordering are identical. The assigned reference numbers are also (save again in one respect) identical, notwithstanding that the numbering is not consecutive (for example in each case "Histogram" is assigned the number 1 and the next listed item, "Pareto" is assigned the number 5).
 84. Mr Aughton accepted that the ordering and numbering of the charts is arbitrary; there is no structural or functional need for them to be in any particular order or to be assigned particular numbers.
 85. He nevertheless maintained his denial that he had copied or otherwise made any use of ProSPC when writing InSPC v1. He pointed, in addition to the fact that there are more chart types in InSPC v1 than in ProSPC, to two differences which he said indicated the latter had not been copied from the former. First, in ProSPC the chart type with assigned number 14 is the wrongly spelt "invididuals", whereas the equivalent in InSPC v1 is the correctly spelt "individuals". Second, in ProSPC "ParetoCatOnly" and "ParetoWithCount" are assigned consecutive numbers 33 and 34, whereas in InSPC v1 they are assigned, respectively, numbers 33 and 35, with an additional chart type, "ParetoCatCols" being assigned number 34.
 86. Neither of these differences begins to explain the coincidence of names and numbers for the charts that *are* included in both ProSPC and InSPC v1, and there are obvious explanations for both differences that are consistent with the

latter having been copied from the former. The spelling correction appearing in the later program is consistent with Mr Aughton having copied and pasted from ProSPC, reviewed what was there, spotted the error and corrected it. The numbering change is explained by the fact that an additional pareto chart type was added in InSPC v1 after ParetoCatOnly, which necessitated the renumbering of the next chart type. The fact that there are more chart types in InSPC v1 is similarly explained by the fact that Mr Aughton started with the enum from ProSPC, and added to it as he developed the program for commercial use.

87. When asked what possible explanation there could be for the similarities in the two enums, Dr Young said that he knew of no technical reason why they should be the same.
88. Mr Aughton was driven, during cross-examination, to suggest that he might have copied both of them, independently, from some third source, such as a book or the internet, although he could not remember doing so, or that he might have retained the original list on a scrap of paper. I do not accept that these are realistic possibilities. While a list of chart types might be found in a book or on the internet, it is not realistically possible that such a list would have had the same numbering system (including the same gaps in sequential numbering of charts) as appears in the enums. Further, Mr Aughton made much of the fact that he does not make preparatory notes of any kind when writing code, so the idea that he made a detailed list of chart types and associated numbers when writing ProSPC – let alone that he retained it for many years – is fanciful.
89. Mr Dufour, in his expert report, suggested that the fact that the two enums did not precisely match positively undermined the assertion of copying. For the reasons I have explained above, I do not accept that. Nowhere in his report did Mr Dufour acknowledge the point that the extent of the similarities between two lists of charts, which could not be explained by any technical reason, is any pointer at all towards copying. In the witness box, Mr Dufour maintained his position, citing his experience of being surprised, having managed teams of engineers, “at the uncommon creativity of engineers to repeat things”. This evidence was unconvincing. When pressed on it, he accepted that he had not seen someone repeating code with such precision as is found in the two enums.
90. Mr Dufour’s failure to acknowledge the obvious force of the similarities between the two pieces of code, noting only points which he contended (wrongly, as I have found) pointed against copying, suggested a lack of independence of approach which undermines his conclusions more generally. In fact, it appeared from his later answers that his approach had been to deny the possibility of copying if two pieces of code were not exactly the same. That would limit the concept of copying to exact replication, which sets the bar unreasonably high and is not the relevant test.
91. In my judgment, the explanation for the coincidences in the ordering and numbering of the chart types, as between the two programs, is that the ChartTypeEnum in InSPC v1 was indeed copied from ProSPC.

92. That, in itself, does not prove that Mr Aughton copied the whole or a substantial part of ProSPC when writing InSPC v1. In circumstances where Mr Aughton's evidence is that he made no reference at all to ProSPC, but wrote InSPC v1 from scratch, my conclusion that he is not telling the truth in relation to this ChartTypeEnum nevertheless undermines the credibility of his denial of copying from ProSPC more generally. As Jacob J pointed out in *IBCOS* (above), it is often identity of trivial matter that traps a copyist. Having regard to the totality of the evidence I am satisfied that he did copy at least a substantial part of ProSPC when writing InSPC v1, for the reasons set out in the following paragraphs.
93. As for the experts, I have already noted the weaknesses in Mr Dufour's evidence, and I accept Dr Young's evidence that – within the limitations due to the lack of source code for either ProSPC or InSPC v1 – there are other indications within the decompiled code consistent with copying. In some instances – as Mr Aughton correctly pointed out – the pieces of code are too short to be of any significance. In other cases, however, Dr Young concluded that, even taking account of the limitations imposed by decompiling and de-obfuscation, there were similarities that were indicative of copying. One example he gave was of a code to save a scatter chart: having enumerated the similarities between them, he concluded that “this means that, not only is the code identified in [the statement of case on copying] similar, it is part of a design and structure which is carried over from ProSPC to InSPC v1”. In fact, these were statements with which Mr Dufour, in the joint statement, agreed.
94. Aside from this, there are a number of matters which cumulatively provide support for the conclusion that Mr Aughton did copy from ProSPC in writing InSPC v1. These are: the circumstances in which Mr Aughton retained ProSPC after leaving PQ; the circumstances in which he deleted it; the circumstances in which he deleted InSPC v1; and his attempts in evidence to downplay the utility of ProSPC.
95. As to the first point, on his departure from PQ, Mr Aughton's contract required him to delete any PQ data from any equipment remaining in his possession. He was specifically reminded of this obligation in a letter dated 20 May 2015. Notwithstanding that I accept that Mr Aughton felt a very strong sense of ownership of any software that he wrote, including that which on any view he wrote for PQ, he knew – having confirmed this to PQ at the time of his disciplinary hearing just two years earlier – that he had agreed with PQ that although he alone had written ProSPC it was owned by PQ. Having been disciplined for what appeared to PQ to be an attempt to offer ProSPC to a competitor, and narrowly survived losing his job in the process, I do not believe that he would have forgotten this at the point he left PQ, particularly when he left PQ to start work on developing a program with (at the very least) the same functionality as ProSPC.
96. Mr Aughton's account of when he deleted ProSPC and, more importantly, why, are inconsistent. His first witness statement, dated 25 June 2020, was provided specifically in response to an order requiring him to explain what had happened to software he once had, but no longer retained. In it, he said that he deleted the source code for ProSPC in the summer of 2015, because “essentially qSPC

[which was what he later called InSPC v1] superseded ProSPC and therefore there was no need for me to retain the source code for ProSPC”. In his second witness statement dated 7 December 2022, he said that, rather than deleting the source code, it was simply overwritten sometime in 2016 when he reformatted the computer on which it sat.

97. This inconsistency cannot be explained simply as a mistake over the dates. His second account involves no positive decision to *delete* the source code. Moreover, Mr Aughton accepted that it was his practice to keep a back-up of software he wrote at home on another computer, having transferred a copy of the software each week via a USB stick. That meant that he retained at any one time three copies of the source code. Reformatting the computer on which one version sat would not have caused it to be deleted from the other computer or the USB stick. Mr Aughton did not provide any explanation for the fact that neither of these other versions remains. Accordingly, I reject this explanation for the deletion of the ProSPC source code.
98. The reason he gave in his first witness statement is in my judgment more plausible. It is a reason, however, which *supports* the conclusion of copying. If it was when InSPC v1 superseded ProSPC that the “need” to retain ProSPC ceased, then this suggests that there was a need to retain it until that point in time. The obvious inference, particularly where there is compelling evidence that at least part of the source code for ProSPC was replicated in InSPC v1, is that he needed the source code for ProSPC until he had made use of it in writing and developing InSPC v1.
99. The circumstances in which Mr Aughton deleted InSPC v1 are also troubling. In his first witness statement, he said that he deleted the source code for InSPC v1 in about August 2018 because it was no longer required, having been superseded by the source code for InSPC v2. On 18 October 2017, Mr Aughton had been sent a letter before action, following PQ having discovered that CyberMetrics had made a demo version of InSPC v1 available.
100. Undertakings were sought from Mr Aughton, including that he would not delete any document, including electronic document, derived from any confidential information of PQ. Although Mr Aughton refused to give such undertakings, he was no doubt aware of the need to preserve relevant materials.
101. He was then specifically asked to make available to PQ the source code for InSPC v1. In a letter dated 19 December 2017, PQ’s solicitors requested disclosure of “the source code for the entire INSPC programme, as built 14 July 2017 and published/offered for sale shortly thereafter.” Mr Aughton’s solicitor’s response, on 18 January 2018, was to say that “the July 2017 version of InSPC is no longer in our client’s possession or control. It has been significantly re-written, not least for the commercial and pragmatic reasons as stated in our previous correspondence.”
102. This was a reference back to their letter of 7 November 2017, which stated that Mr Aughton had taken the commercial and pragmatic decision that the version of InSPC referenced in PQ’s solicitors’ correspondence would not be made available to would be purchasers. The clear implication was that the version

from July 2017 no longer existed because it had been re-written following the letter before action. When Mr Aughton was deposed in connection with the US proceedings against CyberMetrics, on 6 October 2021, he said that the meaning of his solicitor's letter of 18 January 2018 was that he had by then commenced writing InSPC v2.

103. In his second witness statement, however, Mr Aughton claimed that the source code for InSPC v1 was deleted around 17 July 2017, according to his standard "grandfather/father/son" practice, under which any given version of the code is retained only for three days. In his written closing submissions he developed this point, contending that PQ's claim has always been specifically against the 14 July 2017 version of InSPC, and that this no longer exists because it was overwritten by his grandfather/father/son approach.
104. While it would be technically correct to say that on this basis the precise version which existed on 14 July 2017 had ceased to exist within three days, it must have been apparent to Mr Aughton and his solicitors that PQ were not interested solely in the precise version that existed on any particular date within his grandfather/father/son process of developing the software. It is a re-writing of history to suggest that this was what was meant by Mr Aughton's solicitors' letter of 18 January 2018, or the explanation in his deposition.
105. Importantly, the solicitors' correspondence from January 2018 shows that Mr Aughton was aware of the continuing importance to PQ of the source code for InSPC v1.
106. It was his evidence, however, that by August 2018 he believed it was now safe to delete InSPC v1, because he had not heard anything from PQ's solicitors in relation to the threatened claim against him since his own solicitors' letter of 18 January 2018. I do not accept this explanation.
107. Although Mr Aughton was not named as a party to the US proceedings, he was closely connected with them. He had indemnified CyberMetrics against loss resulting from use of his software. If InSPC was created independently, then the source code for InSPC v1 would have been important evidence in support of CyberMetrics' defence. Not surprisingly, therefore, Mr Aughton remained interested in the progress of the US proceedings. On 8 March 2018 he emailed CyberMetrics to ask if they had made progress with the strike out of the proceedings in the US. They replied that they were still working through the process with their lawyers, but would let him know the outcome when they had a ruling. He followed up with a further email on 23 April 2018 asking about progress on the strike out application and saying: "if there's any information you need that would help please let me know." He chased again on 8 June 2018 (noting that he had not heard from PQ that year and did not expect to do so again), and CyberMetrics replied the same day saying: "We are still working on this once we get a resolution we will let you know."
108. In August 2018, therefore, Mr Aughton knew that the US proceedings had not been resolved, and must have appreciated the importance of the original source code for InSPC v1 to CyberMetrics' defence in those proceedings.

Accordingly, I do not believe that he thought matters had concluded such that it was safe for him to delete the source code for InSPC v1.

109. In my judgment, the circumstances are such that I can and do draw the adverse inference that Mr Aughton has not retained the source code for InSPC v1 because it would have been unhelpful to him and CyberMetrics in defending PQ's actual or threatened claims. I return below to the other explanation given by Mr Aughton for deleting InSPC v1 – namely it was superseded by InSPC v2 – but for now note that in circumstances where litigation had been threatened in this jurisdiction, and was ongoing in the US, that is not a sufficient reason to mitigate the adverse inference to be drawn from his failure to retain InSPC v1.
110. Mr Aughton sought to downplay the importance of ProSPC, for example by referring to it on occasions as a piece of “junk”. That assertion is inconsistent with other passages in his evidence to the effect that he worked hard at ProSPC for (at least) more than a year until a point that it was working satisfactorily, and also inconsistent with his behaviour in 2013 in relation to Mr Todd. He accepted that he shared ProSPC with Mr Todd, so that it could be looked at by an SPC consultant in connection with Mr Todd's work. I find it unlikely that Mr Aughton, who set himself high standards for his work, would have been prepared to share a piece of junk with an outside consultant. Moreover, the fact that he knew this consultant was involved with Mr Todd in a work capacity, and the fact that in the course of his exchanges with Mr Todd, he agreed to change the name to ProSPC, makes it unlikely that he was doing no more than sharing a hobby project with a consultant for his own personal interest. If the software was never intended to be seen by others, then its name was irrelevant. I infer that his attempt to downplay the usefulness of ProSPC in this way was intended to make it look less likely that he would have copied from it when writing InSPC v1.
111. For all of the above reasons, I conclude that, in writing InSPC v1 and seeking to make it available on commercial terms to CyberMetrics, Mr Aughton did copy and make use of at least a substantial part of ProSPC in writing InSPC v1.

(3) Was InSPC v2 copied from ProSPC, whether directly or indirectly?

112. While there is only Mr Aughton's word for the fact that ProSPC was deleted, I accept that it was, and that it happened in the circumstances I have set out above (i.e. when it was no longer needed as it had been superseded by InSPC v1). PQ's case under this head therefore rests on indirect copying: did Mr Aughton copy from the source code of InSPC v1 in writing InSPC v2?
113. As I have already noted, the fact that InSPC v2 is written in C# means that it is not possible to carry out a meaningful comparison with the decompiled object code of InSPC v1 and/or of ProSPC, so as to determine whether there has been copying. It is common ground that, given the different syntaxes, there is no direct matching between lines of code.
114. The experts are agreed that there are similarities in lines of code relating to the forms definitions, as between ProSPC, InSPC v1 and InSPC v2, and that this indicates a common functionality between them. Given the lack of ability to

compare source codes, the existence of these similarities in functionality is consistent both with InSPC v2 being written by reference to a running version of InSPC v1 (which would not constitute an infringement) or by reference to its source code.

115. Dr Young was asked to consider whether there was evidence that InSPC v2 had been created to any extent by a process of automatic translation from another computer language. He accepted that there were no specific indications, such as tags, markers or idiosyncrasies left in the InSPC v2 code to suggest that that it had been mechanically translated from v1.
116. In cross examination, Dr Young was shown certain extracts from the source code of Mr Aughton's language editor program written in VB.NET, and the same passages from the C# version of the language editor. He agreed that the differences were such that these parts of the C# version had *not* been copied by an automatic process from the VB.NET version.
117. He was also shown a further extract of code from InSPC v2, containing a Lambda expression, something that is common in C#, whereas the equivalent VB.NET code did not contain such an expression. Dr Young accepted that this was a further passage that could not have been automatically translated from InSPC v1.
118. Dr Young said that automatic translators are available for blocks of source code or entire projects, and that it would therefore have been *possible* to have translated VB.NET code to C# for further modification, or reference. He agreed, however, that it was not possible to prove, from the comparisons of InSPC v1 and InSPC v2 he had seen, that the latter had been automatically translated from the former.
119. In cross-examination he added that anyone seeking to develop software for commercial purposes would, if they used an automatic translator, have to do a very substantial amount of work on the translated code to make it work. For that reason he did not consider that it was a sensible approach. He said:

“If you translate the entire program, what you would get in my opinion ... is effectively a first draft in the new language. It might be helpful but you would still need, I think, to spend time correcting and improving it and you might or might not want to create new forms to link that to. You might want to do this as an exercise simply to see what you got; or to look and see if you are not familiar with the new language what an automatic program would give you.”
120. He also said that translation of forms in this way would be very difficult, and that he did not think that the forms in this case had been automatically translated.
121. His opinion that InSPC v2 *may* have been automatically translated from InSPC v1 was based principally on the fact that InSPC v2 is written in a way which does not make use of features available within C#, but continued to employ methods used within VB.NET. In particular, when writing in VB.NET it is

necessary to use separate data storage languages for different forms of data. C# makes use of Language-Integrated Query (“LINQ”) to overcome that problem. He noted that InSPC v2 had not made use of this, and in comparing certain strings from each program, he found VB.NET implementations for Access and SQL data Server data access which were similar in InSPC v1 and InSPC v2.

122. I accept that automatic translation provides a possible explanation for the use of VB.NET implementations in InSPC v2. It is equally consistent, however, with Mr Aughton, as someone who was far more familiar with VB.NET than C#, continuing to use methods that he had spent years using when writing in VB.NET. That was Mr Aughton’s case. He said: “I have absolutely no interest in LINQ. I do not want to use it and I am not obliged to use it. When I write software, I reserve the right to write it how I like. I choose not to use LINQ. I hate it. I have no interest in it.”
123. Accordingly, the expert evidence in this case does little more than show that a possible explanation for the similarities in functionality of the two programs is that some parts of InSPC v2 were created by a process of automatic translation from InSPC v1. PQ’s claim that Mr Aughton automatically translated the source code of InSPC v1 (or parts of it) when writing InSPC v2 is therefore based primarily on circumstantial evidence. It is not just automatic translation that would amount to infringement, however. Translating ‘by hand’ into C# by reference to the source code for InSPC v1 would also constitute infringement. The claim, in that respect, also rests primarily on circumstantial evidence.
124. In approaching this question, I start from the fact that I have rejected Mr Aughton’s evidence on the key prior questions: that he had written ProSPC purely as a hobby project without making use of any source code derived from PQ’s products; and that he had not copied or otherwise made any use of ProSPC in developing InSPC v1. It does not necessarily follow from this that Mr Aughton is not telling the truth in relation to how he wrote InSPC v2, but it provides a reason to be sceptical, at least, as to his claim that InSPC v2 was created independently.
125. I also have regard to the inherent likelihood that Mr Aughton would have made significant reference to the source code for InSPC v1 when compiling InSPC v2, notwithstanding that it was written in a different language. Having spent a very considerable amount of time writing ProSPC and InSPC v1, so that InSPC v1 was ready to be demonstrated to potential purchasers, it is surprising to say the least that he would have discarded the whole of his work product and started again from scratch, seeking to write a program with essentially the same functionality by reference only to the running version of InSPC v1.
126. The very fact that (as I have found) he copied from ProSPC in developing InSPC v1 supports that conclusion. As does the fact that he retained InSPC v1 while writing InSPC v2, only discarding it on his case when he no longer needed it because it was superseded by InSPC v2. In the same way that the fact that his deletion of the source code for ProSPC occurred only when it was superseded by InSPC v1, and so no longer needed, suggests that his continued need for ProSPC while he wrote InSPC v1 was because he was developing InSPC v1 from it, the fact that he deleted the source code for InSPC v1 only when he

considered it was superseded by InSPC v2 similarly supports the inference that he needed it because he was writing InSPC v2 from it.

127. That inference is also supported by: the fact that he deleted InSPC v1 at all; the timing of its deletion (i.e. after proceedings had been threatened against him, and while proceedings against CyberMetrics were ongoing in the US); and my rejection of his evidence that he deleted it because he felt it was “safe” to do so. As I have already noted, if Mr Aughton had not copied from InSPC v1 when writing InSPC v2, then the source code for InSPC v1 would be valuable evidence to defend the claims threatened in the UK and pending in the US. The fact that he did delete it does not establish copying, but it is a pointer towards it, to be assessed with the other factors I set out in this part of the judgment.
128. Mr Aughton placed emphasis on the fact that, as he said, his practice when writing code was always to start with a blank slate, rather than using any earlier versions of code. That is, however, undermined by his admission in April 2013 (which I have found to be correct) that he re-used routines from PqChartCore both in re-writing GAGEpack and writing ProSPC. Mr Daum confirmed, from his own knowledge acquired when supervising Mr Aughton’s work, that Mr Aughton would indeed re-use code when it was a close match to what he was working on, and that he had done so when re-writing GAGEpack (using code from PqChartCore). I accept that evidence, particularly as it is consistent with Mr Aughton’s own admission made in April 2013.
129. Finally, I consider that the inference is supported by the timing of his re-write of the program in C#, which I find, for the following reasons, was after he was put on notice of PQ’s claim of copying. An innocent explanation for re-writing in C# would have been that he sought to start again in a new language precisely to avoid the claims of copying which had by now been made. If that were so, however, he would have been bound to keep the source code of InSPC v1 precisely to demonstrate he had not copied from it.
130. InSPC v1 was removed from CyberMetrics’ website shortly after the letter before action in October 2017. The first reference in any of the documents available at trial to InSPC v2 is in an email from Mr Aughton’s solicitors to CyberMetrics, dated 29 August 2018, following a telephone call in which various questions had been raised. The email contains Mr Aughton’s responses, which referred to uploads he had recently made to CyberMetrics’ website, commenting: “The demo versions of InSPC and InSPC+ were written in VB.NET – that code no longer exists. The release versions have been rewritten in Visual C# for technical reasons.” The clear implication is that the first time InSPC v2 had been uploaded to CyberMetrics’ website was in or around August 2018.
131. Contemporaneous emails show that Mr Aughton was working on updates to InSPC v1 during the first half of 2017. On 15 May 2017, for example, he emailed CyberMetrics with “the next updates”. He confirmed in cross-examination that this continuing work on InSPC v1 was written in VB.NET. In June 2017, the version of InSPC that was uploaded to CyberMetrics’ website was written in VB.NET.

132. In his written evidence, Mr Aughton did not specify when he started re-writing InSPC in C#. In his first witness statement, he referred (as I have noted above) to the deletion of InSPC v1, which he said occurred in August 2018, once it was superseded by InSPC v2, but he said nothing about when he started writing in C#. He did not deal with this point in his second witness statement.
133. In cross-examination, he said that he started re-writing InSPC in C# in around February 2017. He said that he wrote separate and self-contained portions of the code in C# in order to fix particular bugs. If so, it is surprising that he continued to release updates to InSPC v1 in VB.NET up to October 2017, at which point the letter before action caused him to ask CyberMetrics to take down InSPC v1 from their website. In response to a question pointing out that he would have been doubling his workload by continuing to produce updates in VB.NET at the same time as re-writing the code in C#, he said:
- “We were having these kinds of things where I am changing the language database, which is a 10-second change, and the development of the VB.NET version just slowed to zero. We were also being litigated so there did not seem to be any point. I was losing enthusiasm for everything by now, but I realised that the best thing to do was to ... carry on with the C# rewrite. At least it was something to do.”
134. This evidence, contrary to his claim that he began rewriting the program in C# in February 2017, is consistent with him having done so only after the letter before action, because it was only then that there was any actual or threatened litigation.
135. In his defence, he provided a further explanation, namely that he had completely re-written InSPC in C# as a result of discovering, in late 2016 or early 2017, that there were differences between the obfuscated software that he had uploaded to CyberMetrics and the de-obfuscated version on his own computer.
136. In my judgment, it is most likely that Mr Aughton commenced re-writing InSPC in C# after the letter before action because: (1) it is inherently less likely that he would have continued to develop InSPC v1 for uploading onto CyberMetrics’ website if he had determined to re-write it in C#; (2) this is consistent with InSPC v1 being taken down from CyberMetrics’ website shortly after the letter before action and there being no reference in any document to the new C# version until the end of August 2018; and (3) it is consistent with his answer in cross-examination that he re-wrote the program in C# in the context of there being litigation and losing enthusiasm for everything.
137. A significant part of Mr Aughton’s defence is that PQ cannot establish their case on copying because they have not produced any source code of their software to compare with InSPC v2. He claims that PQ have deliberately avoided such a comparison because they know it would be damaging to their case “...so they have persisted with the ‘decompiled’ approach which, as well as being highly prejudicial, has allowed them to generate their own version of what they claim is valid ‘decompiled’ code without the inconvenience of expert oversight.”

138. It is true that the decompilation process was undertaken by PQ and not overseen by the experts. I accept, however, the evidence of Mr Daum, that the process undertaken was as described in the Statement of Case on Copying, which was, as confirmed by Dr Young, a recognised standard procedure. Importantly, it was common ground in relation to the most critical evidence (the comparison of the decompiled and de-obfuscated ChartTypeEnums) that the features relied on would not have been altered by either process.
139. Mr Aughton's suspicions were understandably aroused by PQ's failure to provide proper disclosure of all decompiled source code until very shortly before trial. In the decompiling process PQ had initially used a program called JustDecompile, but had switched to a program called .NET Reflector. In preparing Annex 1 to the statement of case on copying (which described the decompiling process), no reference was made to JustDecompile because it was wrongly believed by PQ's legal team that the use of .NET Reflector had entirely replaced the use of JustDecompile. In fact, only a subset of the comparisons relied on (and referred to in Annexes 3 and 5 of the statement of case on copying) had used the .NET Reflector tool. In a further error, while PQ's servers have two different versions of file shares containing decompiled code and comparisons – one containing both versions of decompiled code and the other containing only the JustDecompile version – only the second file share was uploaded to the data room for disclosure purposes.
140. The experts had noted some discrepancies when seeking to match the comparisons in Annexes 3 and 5 of the statement of case on copying with the decompiled code in the data room. These were traced by PQ (and Dr Young) back to the fact that the wrong version of the decompiled code had been uploaded to the data room.
141. Mr Aughton made it clear at the outset of the trial that he did not wish the trial to be adjourned to give him the opportunity to review the newly disclosed decompiled code, but he was given the opportunity to request the recall of Mr Daum (who had overseen the decompiling process) once he had had such an opportunity, so that he could put to him any points arising out of his review of the code. He declined the opportunity to take a day out (in addition to the weekend) between the end of the evidence and closing argument so that he could review the code. He did seek to recall Mr Daum, at the start of closing submissions on the final day of the trial, but given that he had not reviewed the newly disclosed decompiled code I did not allow that, since the possibility of recalling Mr Daum was only for the purpose of putting to him any points arising out of Mr Aughton's review of the new material. Mr Aughton complained that the email with a link to the newly disclosed decompiled code had been sent to his work email address, which he would have had to travel home to access. I do not accept that Mr Aughton was unable to review the material. Even if, which I find difficult to believe, he has no access to his work email while travelling, he could easily have requested it to be forwarded to his personal email address in time for him to review it. In my judgment he made the deliberate choice not to review the new material.
142. While, in light of the above errors on PQ's part, it is understandable that Mr Aughton's suspicions were aroused, I am satisfied that the errors were just that,

and were not the result of any deliberate decision to withhold relevant material on the part of PQ. Given that by reference to the correct version of the decompiled code certain discrepancies between the compared versions (which pointed against, not towards, copying) disappeared, the failure to disclose the correct version was against PQ's interests.

143. More broadly, Mr Aughton's attack on PQ's case based on their failure to make a comparison between source codes ignores the obvious fact that PQ have had to resort to the 'decompiled' approach because he himself had deleted the original source code for both ProSPC and InSPC v1, the latter in circumstances that invite inferences against him.
144. Mr Aughton submitted that PQ's evidence – in particular that of Ms Savage – was itself inconsistent as to when they had retained, or deleted, ProSPC from their systems. He relied on passages in the particulars of claim which, having defined GAGEpack and ProSPC as "the Works", a number of other software programs as "the Further Material", and the source code for all of these as the "Source Code" then stated: "the Claimants have taken steps to keep the said Source Code secure, secret and confidential". It is true that on a literal reading of the relevant paragraphs in the pleading, this implies that PQ has retained and kept secret and secure the source code for ProSPC. That is clearly not, however, PQ's case, and this is an instance in my judgment of a poorly drafted pleading. In cross-examination, Ms Savage said that she had not understood the pleading to say that PQ had kept and safeguarded a copy of the source code for ProSPC.
145. It is true that PQ did *see* the source code for ProSPC in April 2013. That was when Mr Aughton (having initially handed over a blank computer) supplied his work computer on which was loaded ProSPC. Ms Savage's evidence is that this was reviewed by PQ's technical personnel in the US via a remote connection to that computer but that there had been no need to take or keep a copy of it, and that having made enquiries within PQ, they do not now have a copy of it. This is, in my view, inherently likely to have occurred. Once PQ had established the similarities between ProSPC and their own software, and had concluded disciplinary proceedings against Mr Aughton, deciding to keep him on as an employee under renewed and more stringent contractual terms, they had no need for the source code.
146. Mr Aughton relies on that very fact to contend that these proceedings are nothing more than a vendetta against him, pursued in order to protect rights that PQ claim to have in software which they regarded as worthless. That, however, is not the point. Having reached the conclusion in 2013 that ProSPC had been written with the benefit of PQ's software (as Mr Aughton then admitted) and other resources, the fact that they had no use for ProSPC (because their own products had been developed within a different environment) does not mean that it ceased to be of value to a competitor, and thus something which they continued to have an interest in protecting.
147. Mr Aughton also relied on a passage in Ms Savage's witness statement in which she referred to having asked for all references to ProSPC to be removed from the company network. He interpreted this as saying that she had asked for the source code itself to be deleted. That is not, however, what it says. The context

for her request was that neither the disciplinary proceedings involving Mr Aughton nor the reason for it (his sharing of ProSPC with Mr Todd) had been publicised within the company. Accordingly, what she wanted to be deleted (for Mr Aughton's benefit as much as that of PQ) was material, such as email exchanges, which could reveal either of those things. This passage in her evidence was immediately followed by her statement that she did not recall Mr Aughton ever sending the source code to PQ, but that she recalls it being viewed remotely on his work computer.

148. Taking into account each of the matters I have referred to above, I conclude that it is more likely than not that Mr Aughton did copy from at least a substantial part of the source code for InSPC v1 when writing InSPC v2.
149. Given the deletion of source code for InSPC v1 and the lack of any record of how InSPC v2 was created save for the program itself, I am unable to conclude whether this was done with the use of an automatic translator. In view of my conclusions as to the timing of the re-write of InSPC into C#, there was sufficient time either to conduct the re-write manually, making reference to the source code of InSPC v1, or by a process of automatic translation followed by months of work needed to correct the raw initial product.
150. It is sufficient, however, to establish infringement of copyright and breach of contractual duties of confidence, that he wrote InSPC v2 by copying from the source code of InSPC v1 whether or not he used an automatic translator. It is therefore unnecessary for me to decide precisely how he did this.

Legal conclusions

151. In light of the above factual conclusions, it follows that Mr Aughton infringed PQ's copyright and breached his contractual duties of confidentiality in writing both InSPC v1 and InSPC v2.
152. PQ also advanced a case based on Mr Aughton's breach of duty as a director. As I have noted, Mr Aughton was a director of PQE between 1996 and April 2013. He contends that he was only a director in name, because PQE had to have someone in England appointed as a director. Once appointed in law as a director, however, I am satisfied that Mr Aughton assumed the fiduciary and other duties in law of a director.
153. It is unnecessary to consider, however, the consequences of that conclusion. The fact that he owed such duties would not have prevented him writing software in his own time as a purely hobby project. If ProSPC was not a hobby project, then the copyright and confidence in it belonged to PQ irrespective of whether he owed fiduciary duties. His fiduciary duties would have been centrally relevant to any claims that PQE might have wished to bring against Mr Aughton arising out of his conduct in 2012-2013, but that is water under the bridge. Accordingly, the fact that he owed fiduciary duties does not add materially to the analysis in relation to any of the three issues I have addressed above.