

E.I.P.R. 2019, 41(5), 271-274

European Intellectual Property Review

2019

Reverse engineering of software: a safe harbour in Europe but not
safe in the US: SAS Institute Incorporated v World Programming Ltd

Vincent Capati ¹

© 2019 Sweet & Maxwell and its Contributors

Subject: Intellectual property

Other Related Subject: Contracts. European Union. Information technology.

Keywords: Breach of contract; Click-wrap agreements; Copyright; EU law; Licensing agreements; Reverse engineering; Safe harbour; Software; United States;

Legislation:

Directive 2009/24

Cases:

[SAS Institute Inc v World Programming Ltd \[2013\] EWCA Civ 1482; \[2015\] E.C.D.R. 17; \[2013\] 11 WLUK 538 \(CA \(Civ Div\)\)](#)

SAS Institute Inc v World Programming Ltd 874 F.3d 370 (4th Cir (US))

**271 The [EU's Council Directives 2001/29 and 2009/24](#) (EU Software Directive) created a sharp split between American and British jurisprudence when interpreting the scope of software licences. The [EU Software Directive](#) entitles a software user "without the authorization of the [software developer], to observe, study or test the functioning of the program in order to determine the ideas and principles which underlie any element of the program" and that "[a]ny contractual provisions contrary ... shall be null and void".¹ The SAS Systems dispute tested the teeth of the [EU Software Directive](#). This Opinion reviews recent inconsistent judgments between the British and American courts and sets forth business and contractual considerations for software developers operating in those jurisdictions.*

Opinion

A recent cross-Atlantic dispute between two software developers laid bare some startling realities for both American and European software companies: EU and UK laws protect competitors who reverse-engineer software in violation of US copyright laws and contractual restrictions—and go so far as to void contract terms to the contrary. Meanwhile, the US holds firm to the principle that a contract is a contract, even if those terms are void elsewhere. This paradox recently created the stark outcome where the same accused software product, the same alleged acts of infringement, and the same alleged breach of contract resulted in a verdict in favour of a defendant in the UK but an almost \$80 million damages award in the US.

In [SAS v WPL](#), the UK and US courts independently concluded that World Programming Ltd (WPL) breached its licence agreement with SAS Institute Inc (SAS) by reverse-engineering the SAS Systems software. But when SAS sought to enforce the

contract, it *lost* in the UK and *won* in the US, even though the UK courts recognised that US law governs. How did this happen? This Opinion will focus on the divergent rulings effected by EU law and the US courts' refusal to recognise the UK courts' earlier judgment, and then offer practical tips on navigating this situation until it can be remedied by courts and law-makers.

A binding clickwrap agreement and a breach

SAS, the world's largest privately held software company, is an American company based in North Carolina.², ³ It sells the Base SAS, SAS/ACCESS, SAS/Graph and SAS/STAT software packages (collectively, "SAS Systems") that enable users to process and analyse data.⁴ Users can manipulate data by writing scripts in SAS Language, and SAS Systems execute those user-made scripts to generate reports of the inputted data.⁵ SAS also offers the SAS Learning Edition, a limited, lower-cost version of SAS Systems meant to teach users SAS Language.⁶ While the Learning Edition provides the same functionality as SAS Systems, it processes only a limited amount of data. *272 ⁷

SAS Systems and the Learning Edition are available to both individuals and corporations.⁸ But to install the Learning Edition on a machine, a user must click "Yes" to a clickwrap agreement before installation proceeds.⁹ This clickwrap agreement prohibits licensees from reverse-engineering the Learning Edition and limits use of the software to "non-production purposes".¹⁰ That is, licensees cannot use the program to produce (or build) a similar, competing program.¹¹ The clickwrap agreement further requires that North Carolina state and US federal laws govern the licence.¹²

WPL is a privately held British company based in Hampshire.¹³ It sells World Programming System (WPS), an analytical software capable of executing user-made SAS scripts.¹⁴ Before WPL launched WPS, only SAS Systems could read scripts written in SAS Language.¹⁵ Although other analytical software was available, switching away from SAS Systems meant that users would have to rewrite their own SAS scripts in a different language readable by the new platform.¹⁶ So, to avoid rewriting scripts already in SAS Language, SAS customers had no alternative but to continue licensing SAS Systems.¹⁷ To service those customers seeking a new software platform but opposed to rewriting new scripts, WPL sought to develop WPS as an alternative to SAS Systems.¹⁸

In 2003, WPL acquired its first copies of SAS's Learning Edition (and it acquired a few more in 2005, 2007, and 2009) and the corresponding user manuals, the SAS Manuals.¹⁹ Using the SAS Manual as a model, WPL programmers wrote code to emulate features of the SAS Systems in WPS.²⁰ And despite the clickwrap agreement prohibiting use for "non-production purposes", WPL breached the contract by using the Learning Edition to build WPS.²¹ WPL executed identical scripts on both the Learning Edition and WPS to identify any differences in outputs and to gain insight to the Learning Edition's functionality.²² If the WPS output was different, WPL programmers revised the WPS code to better emulate SAS Systems.²³ And because WPL never had access to either the SAS Systems source code or object code, it did not, in fact, copy any source code or structural design of the source code.²⁴

Since WPL launched WPS, some SAS customers have declined to renew their SAS Systems license in favour of WPS.²⁵ In September 2009, SAS filed suit in the UK and later *lost*.²⁶ In January 2010, SAS filed the same breach of contract complaint in the US and later *won*.²⁷

Why did the UK's High Court not hold WPL in breach?

WPL breached its contract by reverse-engineering SAS Systems, so SAS sought to enforce its contract, but the High Court ruled against SAS.²⁸ The Court of Appeal affirmed, and the Supreme Court declined review.²⁹

The High Court and the Court of Appeal both recognised that the laws of North Carolina and the US govern the licence, and the High Court proceeded to trial with the understanding that there was no difference between US and English laws.³⁰ Not so, apparently. At trial, SAS proved that WPL breached when it ran scripts through the Learning Edition to learn more about the Learning Edition's functionality.³¹ This use was for "non-production purposes" and went beyond the scope of the licence.³² After trial, the High Court concluded that it needed the Court of Justice of the European Union (CJEU) to interpret [Council Directives 2001/29](#) and [2009/24](#) (the EU Software Directive) to determine whether the [EU Software Directive](#) voided the "non-production purposes" term that WPL violated.³³

Under the [EU Software Directive](#), a licensee is entitled to observe, study, and test a program's functionality to determine underlying ideas and principles, *regardless* of the owner's authorisation.³⁴ If a licence agreement says otherwise, those terms are void, and a licensee is therefore not in breach.³⁵ Here, the High Court found that the *273 "non-production purposes" term was void.³⁶ Even though WPL is a licensee for SAS Systems, SAS cannot contract around the [EU Software Directive](#) and restrict what are WPL's rights by law. As a licensee, WPL was entitled to observe, study and test the functioning of the Learning Edition.³⁷ Because the "non-production purposes" term contradicted the [EU Software Directive](#), the term was void, and WPL did not breach.³⁸

Did the UK litigation preclude the same US litigation?

SAS's breach of contract claim in the US required ruling on the same contract that the UK courts considered, but the North Carolina District Court gave no preclusive effect to the High Court's final judgment and held WPL in breach of its contract with SAS.³⁹ WPL breached the term prohibiting reverse-engineering when it analysed how the Learning Edition works to recreate its functionality in WPS.⁴⁰ WPL also breached the term limiting use to "non-production purposes" when it used the Learning Edition to build a competing product.⁴¹ After trial, SAS recovered \$79,129,905 in damages.⁴² WPL appealed to the US Court of Appeals for the Fourth Circuit (Fourth Circuit), which affirmed the District Court's judgment, and the Supreme Court declined review.⁴³

On appeal, the Fourth Circuit ruled that the doctrine of claim preclusion did not bar the District Court from hearing SAS's breach of contract claim even though the UK had already made a final judgment on the same contract between the same parties.⁴⁴ Under US law, claim preclusion bars a later proceeding only if (1) there is a final judgment on the merits; (2) the cause of action is the same; and (3) the parties are the same.⁴⁵ Despite the UK courts issuing an earlier, final, judgment on the same contract to the same parties to satisfy the first and third prongs, the Fourth Circuit held that the cause of action (i.e. the breach of contract claim) in the US was not the same as that in the UK for two reasons.⁴⁶

First, causes of action are rooted in law, and the relevant US and UK/EU laws are different.⁴⁷ The US does not have a similar [EU Software Directive](#) that voids contractual terms. And because the laws are different, mechanically applying the doctrine of claim preclusion, even though the same parties are disputing the same allegations would undermine US and North Carolina public policies in favour of applying EU and UK laws.⁴⁸ On balance, the Fourth Circuit concluded that the US is more protective of intellectual property, and North Carolina is more protective of the sanctity of a contract, than UK and EU laws.⁴⁹ And when a foreign judgment conflicts with the public policy of the state of North Carolina or the US writ large, then North Carolina courts may and should refuse to recognise that foreign judgment.⁵⁰

Secondly, the Fourth Circuit reasoned that the UK was not an adequate forum to hear SAS's claims.⁵¹ Even though SAS and WPL agreed to a contractual relationship governed by North Carolina law, and even though the UK courts recognised that North Carolina and US federal law govern the licence, the UK courts nevertheless invalidated portions of the contract according to EU and UK laws.⁵² Applying claim preclusion in this instance would bar a North Carolina company from asserting its contractual rights, as governed by North Carolina law, in a North Carolina court.⁵³ Because the cause of action prong was not satisfied, claim preclusion did not bar the District Court from hearing the case even after resolution in the UK.⁵⁴

The sequitur? A second jurisdiction, a second bite of the apple

This case serves as a harsh reminder, once again, that rights and obligations are shaped by where business entities engage in commerce. It was easy for SAS and WPL to enter into a contract when WPL needed only to click "Yes". While this clickwrap agreement established a binding contract that defined rights and obligations as governed by North Carolina laws, the UK court was still bound to UK laws, and voided terms that were otherwise enforceable in the US. This case is thus another reminder that laws are territorial. And while WPL prevailed in the UK, SAS had a second opportunity for relief in the US.

For software developers, this case also highlights obligations (or lack thereof) for computer programmers. Overarching EU laws appear to favour competition over contracts, and the [EU Software Directive](#) not only endorses the testing of software, but it also strikes *274 contractual provisions restricting the same behaviour. This creates a safe harbour in Europe that does not exist across the pond. The effect of this safe harbour is vague—would it encourage competition, or would it increase the cost of doing business? Would it encourage software development in the UK, but maintain the status quo in the US? Perhaps US innovators might double down and include specific liquidation damages to contracts governed by US law and enforced by US courts for reverse-engineering in the EU. If this dispute suggests a roadmap to relief, do innovators have any other avenue for relief beyond filing a lawsuit in the US? And as the UK prepares to separate from the EU, it may be worth reconsidering whether the EU's safe harbour is in line with the UK's national policy goals.

Vincent Capati

Wiley Rein LLP, Washington DC

Footnotes

- ¹ Vincent Capati is an Associate in the Intellectual Property and Litigation groups at Wiley Rein LLP, an international law firm based in Washington, DC. Dr Capati's practice focuses on patent litigation, and patent and trade mark prosecution. He earned his Juris Doctor (JD) degree from the University of New Hampshire School of Law and holds Doctor of Pharmacy (PharmD) and Master of Science (MS) degrees from the University of Southern California. Dr Capati is a member of the Bar of the District of Columbia and is a registered pharmacist with the State of New Hampshire. Any opinions expressed in this commentary strictly reflect Dr Capati's personal views based on his current understanding of the relevant laws, and do not necessarily reflect the views of the law firm or its clients.
- ¹ [SAS Inst Inc v World Programming Ltd \[2013\] EWCA Civ 1482; \[2014\] R.P.C. 8](#) at [95]–[96].
- ² SAS Inst. Inc v World Programming Ltd 64 F. Supp. 3d 755, 759 (E.D.N.C. 2014).
- ³ All background facts and factual assertions in this article are taken directly from judicial opinions and thus assumed to be true and accurate. The author takes no independent position on these facts or contrary arguments.
- ⁴ [SAS Inst. Inc v World Programming Ltd \[2013\] EWHC 69 \(Ch\); \[2013\] R.P.C 17](#) at [1].
- ⁵ [SAS Inst. v World Programming \[2013\] EWHC 69 \(Ch\); \[2013\] R.P.C 17](#) at [1]–[2].
- ⁶ SAS Inst. Inc v World Programming 874 F. 3d 370, 375 (4th Cir. 2017); [SAS Inst. v World Programming \[2013\] EWCA Civ 1482; \[2014\] R.P.C. 8](#) at [2].
- ⁷ SAS Inst. v World Programming 874 F. 3d 370, 376 (4th Cir. 2017); [\[2013\] EWCA Civ 1482, \[2014\] R.P.C. 8](#) at [2].
- ⁸ SAS Inst. v World Programming 874 F. 3d 370, 375 (4th Cir. 2017).

- 9 SAS Inst. v World Programming 874 F. 3d 370, 376 (4th Cir. 2017).
- 10 SAS Inst. v World Programming 874 F. 3d 370, 376 (4th Cir. 2017).
- 11 SAS Inst. v World Programming 874 F. 3d 370, 383 (4th Cir. 2017).
- 12 [SAS Inst. v World Programming \[2013\] EWCA Civ 1482; \[2014\] R.P.C. 8](#) at [92].
- 13 SAS Inst. v World Programming 64 F. Supp. 3d 755, 759–780 (E.D.N.C. 2014).
- 14 SAS Inst. v World Programming 874 F.3d 370, 376 (4th Cir. 2017); [SAS Inst. v World Programming \[2013\] EWCA Civ 1482; \[2014\] R.P.C. 8](#) at [3].
- 15 [SAS Inst. v World Programming \[2013\] EWHC 69 \(Ch\); \[2013\] R.P.C 17](#) at [2].
- 16 [SAS Inst. v World Programming \[2013\] EWHC 69 \(Ch\); \[2013\] R.P.C 17](#) at [3].
- 17 [SAS Inst. v World Programming \[2013\] EWHC 69 \(Ch\); \[2013\] R.P.C 17](#) at [2].
- 18 [SAS Inst. v World Programming \[2013\] EWHC 69 \(Ch\); \[2013\] R.P.C 17](#) at [3].
- 19 SAS Inst. v World Programming 874 F. 3d 370, 376 (4th Cir. 2017); [SAS Inst. v World Programming \[2013\] EWCA Civ 1482; \[2014\] R.P.C. 8](#) at [17].
- 20 [SAS Inst. v World Programming \[2013\] EWCA Civ 1482; \[2014\] R.P.C. 8](#) at [17].
- 21 SAS Inst. v World Programming 874 F. 3d 370, 382–383 (4th Cir. 2017).
- 22 [SAS Inst. v World Programming \[2013\] EWCA Civ 1482; \[2014\] R.P.C. 8](#) at [18].
- 23 [SAS Inst. v World Programming \[2013\] EWCA Civ 1482; \[2014\] R.P.C. 8](#) at [18].
- 24 [SAS Inst. v World Programming \[2013\] EWCA Civ 1482; \[2014\] R.P.C. 8](#) at [13]; [SAS Inst. v World Programming \[2013\] EWHC 69 \(Ch\); \[2013\] R.P.C 17](#) at [3].
- 25 SAS Inst. v World Programming 874 F. 3d 370, 376 (4th Cir. 2017).
- 26 SAS Inst. v World Programming 874 F. 3d 370, 376 (4th Cir. 2017).
- 27 SAS Inst. v World Programming 874 F. 3d 370, 376 (4th Cir. 2017).
- 28 [SAS Inst. v World Programming \[2013\] EWCA Civ 1482; \[2014\] R.P.C. 8](#) at [4], [19], [81], [90], [102], [110].
- 29 [SAS Inst. v World Programming \[2013\] EWCA Civ 1482; \[2014\] R.P.C. 8](#) at [8]; SAS Inst. v World Programming 874 F. 3d 370, 377 (4th Cir. 2017).
- 30 [SAS Inst. v World Programming \[2013\] EWCA Civ 1482; \[2014\] R.P.C. 8](#) at [92].
- 31 [SAS Inst. v World Programming \[2013\] EWCA Civ 1482; \[2014\] R.P.C. 8](#) at [93].
- 32 [SAS Inst. v World Programming \[2013\] EWCA Civ 1482; \[2014\] R.P.C. 8](#) at [93].
- 33 [SAS Inst. v World Programming \[2013\] EWHC 69 \(Ch\); \[2013\] R.P.C 17](#) at [55], [62]–[63], [72]–[73], [79].
- 34 EU Software Directive, para.12.62.
- 35 [SAS Inst. v World Programming \[2013\] EWHC 69 \(Ch\); \[2013\] R.P.C 17](#) at [79].
- 36 [SAS Inst. v World Programming \[2013\] EWHC 69 \(Ch\); \[2013\] R.P.C 17](#) at [79].
- 37 [SAS Inst. v World Programming \[2013\] EWCA Civ 1482; \[2014\] R.P.C. 8](#) at [97]–[98], [101]–[102].
- 38 [SAS Inst. v World Programming \[2013\] EWCA Civ 1482; \[2014\] R.P.C. 8](#) at [102]; [SAS Inst. v World Programming \[2013\] EWHC 69 \(Ch\); \[2013\] R.P.C 17](#) at [79].
- 39 SAS Inst. v World Programming 874 F. 3d 370, 377 (4th Cir. 2017).
- 40 SAS Inst. v World Programming 874 F. 3d 370, 382 (4th Cir. 2017).
- 41 SAS Inst. v World Programming 874 F. 3d 370, 383 (4th Cir. 2017).
- 42 SAS Inst. v World Programming 874 F. 3d 370, 377 (4th Cir. 2017).
- 43 SAS Inst. v World Programming 874 F. 3d 370, 378 (4th Cir. 2017).
- 44 SAS Inst. v World Programming 874 F. 3d 370, 378 (4th Cir. 2017).
- 45 SAS Inst. v World Programming 874 F. 3d 370, 378 (4th Cir. 2017).
- 46 SAS Inst. v World Programming 874 F. 3d 370, 378 (4th Cir. 2017).
- 47 SAS Inst. v World Programming 874 F. 3d 370, 379 (4th Cir. 2017).
- 48 SAS Inst. v World Programming 874 F. 3d 370, 379 (4th Cir. 2017).
- 49 SAS Inst. v World Programming 874 F. 3d 370, 379 (4th Cir. 2017).
- 50 SAS Inst. v World Programming 874 F. 3d 370, 379 (4th Cir. 2017).
- 51 SAS Inst. v World Programming 874 F. 3d 370, 379 (4th Cir. 2017).
- 52 SAS Inst. v World Programming 874 F. 3d 370, 379 (4th Cir. 2017); [SAS Inst. v World Programming \[2013\] EWCA Civ 1482; \[2014\] R.P.C. 8](#) at [92].
- 53 SAS Inst. v World Programming 874 F. 3d 370, 379–380 (4th Cir. 2017).
- 54 SAS Inst. v World Programming 874 F. 3d 370, 380 (4th Cir. 2017).

© 2019 Sweet & Maxwell and its Contributors

E.I.P.R. 2019, 41(5), 271–274