

Recent software case developments

EUCJ

CJEU, 22 Dec. 2010, case C-393/09, Bezpečnostní softwarová asociace (BSA)

CJEU, 2 May 2012, case C-406/10, SAS Institute Inc. v. World Programming Ltd

CASRIP, Strasbourg, March 23d 2015



Franck Macrez



Introduction

Two judgements from the ECJ concerning the interpretation of the “Computer Programs” Directive of 14 May 1991 (for the first time) and the “Information Society” Directive of 22 May 2001

→ questions raised cover all the different aspects of a computer program

BSA 2010:

1.- Inclusion of the graphic user interface (GUI) in the category of computer programs within the meaning of Directive 91/250 ?

2.- Could television broadcasting of such a graphic user interface be considered to constitute communication to the public within the meaning of Directive 2001/29?

→ ECJ's response : GUI = graphic work and not a program (“traditional” rules of copyright rather than the specific system for software)



SAS 2012 (Grand chamber):

Other aspects of computer program :

- source code
- object (compiled / binary) code
- functionality (itself)
- format data files (→ logical interface)
- programming languages
- program's user manual



- Programming languages : WPL's program allowed users to keep their "scripts" (i.e. small personal programs) written in the language initially created by SAS
 - Formats of data files (→ Interoperability): WPL offered software capable of reading, understanding and interpreting files in the format created by SAS so that customers could easily migrate from one system to the other while retaining their personal files
- ECJ's response : functionality of a program, programming languages and formats of data files are not protected as such



A question of method...

I.- THE CLASSIFICATION METHOD

A – The Distributive Method Established by the Court

B – Difficult Distributive Method

II – THE RESULTS OF THE CLASSIFICATION

A – Non-Copyrightable Forms

B – Copyrightable Forms



I.- The classification method

Distributive application of the legal regimes:

Directive 91/250 constitutes a *specific law* for computer programs which should be coordinated with the “ordinary law” discovered at the time by the Court of Justice...



Franck Macrez



I.- The classification method

A – The Distributive Method Established by the Court

1 – The Foundations: Special Law and “Ordinary Law”

2 – Distributive Classification

B – Difficult Distributive Method

1 – The Porosity between the “Ordinary Law” and the Special Law

2 – Confusion between the Object and the Criterion



I.- A – The Distributive Method Established by the Court

I.- A.- 1 – The Foundations: Special Law and “Ordinary Law”

I.- A.- 2 – Distributive Classification



I.- A.- 1 – The Foundations: Special Law and “Ordinary Law”

BSA, §44 : Directive 2001/29 = the “ordinary law of copyright”
Confirmed by *SAS*

→ Europe has a (new?) *general law of copyright* with Directive 2001/29

// UsedSoft decision, C-128/11 (2012) : “Directive 2009/24 [the codified version of Directive 91/250], which concerns specifically the legal protection of computer programs, constitutes a *lex specialis* in relation to Directive 2001/29”



- *BSA* : while the graphic interface is not viewed as a “form of expression of a computer program” based on Directive 91/250, the courts must ascertain whether the interface meets the criteria for the default copyright protection (Dir. 2001/29)
- *SAS* : identical about languages. The programming language’s exclusion from the category of “forms of expression” of a program under Article 1(2) of Directive 91/250 “cannot affect the possibility that the SAS language [...] might be protected, as works, by copyright under Directive 2001/29 if they are their author’s own intellectual creation” (§45 ; comp. AG’s Opinion, §75-76 : no protection at all)



I.- A.- 2 – Distributive Classification

→ to split the program up by analysing its various aspects separately : fragmented analysis

SAS (§42) : a computer program’s functionality, programming language and format of data files do not “constitute a form of expression of *that* program and, *as such*,” are not protected by the specific copyright in programs

→ No *accessorium sequitur principale* method : the important thing is to conduct a **differentiated analysis**



B – Difficult Distributive Method

1 – Foundations : The Porosity between the “Ordinary Law” and the Special Law

2 – Implementation : Confusion between the Object and the Criterion



Franck Macrez



I.- B.- 1 – Foundations : The Porosity between the “Ordinary Law” and the Special Law

- Seeds of the (confusing) reasoning : *Infopaq* decision (C-5/08, 2009) → the phrase used in three specific European directives concerning computer programs (1991), databases (1996) and photographs (2006) was extended to the whole of copyright, namely a work is original if it constitutes “*its author’s own intellectual creation*”

→ generalises to all works a criterion introduced for very specific creations // the sole protection criterion is the result of *induction* from the special to the general = porosity



- **Problem** : At the same time, there is a displayed autonomy (distributive application of the *acquis*) and an undeniable porosity

→ the special law acts as a conceptual ferment for the law of general application even though it is supposed to derogate from it (« *Specialia generalibus derogant* »)



- **Consequence** :

- *BSA* : GUI = work under the “ordinary law”.

- **But question 2** : Is broadcasting of that “work” constitutes communication to the public (Dir. 2001/29) ? In principle, YES (*BSA* §55), but in that case NO : “*television viewers receive a communication of that graphic user interface solely in a passive manner, without the possibility of intervening*” (§57)

- ➔ incomprehensible, « error of judgement » (Derclaye 2011)



I.- B.- 2 – Implementation : Confusion between the Object and the Criterion

The legal reasoning process should determine whether

- there is **a form** eligible for protection and
- then analyse it to look for its originality which is the only protection **critierion**, whatever the definition adopted.

But *Infopaq* seems to completely evade the concept of work.



- *BSA / GUI* : the “criterion cannot be met by components of the graphic user interface which are differentiated only by their technical function” (originality) (§48) ...

... because “the idea and the expression become indissociable” (subject-matter) (§49, AG Opinion §77)

→ the copyrightable form and originality are confused



→ Does the ECJ adopt the “merger doctrine” known in American copyright law ?

Yes, but did it « ineptly » (Derclaye 2011 ; Gervais and Derclaye 2012) :

the merger is effected between the subject matter and the criterion, whereas the principle of the merger doctrine is to deny protection when the expression and the idea are inextricable



- Software Dir. : clear distinction between the object (subject matter) and its character in differentiated textual provisions :
 - “the expression in any form of a computer program” (Article 1(2))
 - originality, defined as its author’s “own intellectual creation” (Article 1(3)).
- “the expression in any form of a computer program” = ?
 - source code and object code : YES without any doubt (expressly, see : Article 10(1) of the TRIPs Agreement)



→ As to the other forms that may orbit around a program, the key is to view the program as a ***functional form*** : “any form of expression of a computer program must be protected from the moment when its reproduction would engender the reproduction of the computer program itself, thus enabling the computer to perform its task” (*BSA* §38 ; *SAS*, implicitly, §38)



→ Is **functionality** a « form of expression » of a program ? (1st question in *SAS*)

→ AG Opinion : NO, because no originality (§55)

→ the true reason should be that the functionality as such is not sufficiently formalised to enjoy protection. Functionality is « assimilable to an idea » (ECJ)...

Finally... The detours taken in the field of originality perhaps make it a European-style merger doctrine (?). However, the end result seems the same (*Altai* 1992, etc.)



II – THE RESULTS OF THE CLASSIFICATION

A – Non-Copyrightable Forms

1 – Affirmation of the Principle: The Exclusion of Ideas

2 – Application of the Principle: the Exclusion of Functionalities

B – Copyrightable Forms

1 – The Relative Scope of the Exclusion of Ideas

2 – Existing Protection Possibilities: “Ordinary Law” and Special Law



II.- A – Non-Copyrightable Forms

The principle of the exclusion of ideas in the *SAS* judgment is a major assertion but its application seems intended only to exclude functionalities

1 – Affirmation of the Principle: The Exclusion of Ideas

2 – Application of the Principle: the Exclusion of Functionalities



II.- A.- 1 – Affirmation of the Principle: The Exclusion of Ideas

Article 1(2) of Directive 91/250 provides:

“Ideas and principles which underlie any element of a computer program, including those which underlie its interfaces, are not protected by copyright under this Directive.”

(see also : recitals 13-15 ; WIPO Copyright Treaty of 20 Dec. 1996, Article 2)



→ *“to accept that the functionality of a computer program can be protected by copyright would amount to making it possible to monopolise ideas, to the detriment of technological progress and industrial development” (SAS, §40)*



Franck Macrez



Reason :

The legislator's objective was to limit protection to the program's expression alone in order to leave "*other authors the desired latitude to create similar or even identical programs provided that they refrain from copying*" (§41)

Two remarks :

- “copyright” here is the (industrial) *lex specialis*
- Very broad assertion: maybe the Court runs over into the field of patents where industrial progress is the traditional goal ?



II.- A.- 2 – Application of the Principle: the Exclusion of Functionalities

- Functionality not protected, with no referral to the “ordinary law” of copyright (unlike about programming languages and formats of data files)
 - + contractual provisions must respect the principle of the non-appropriation of ideas (7th and 8th preliminary questions)
 - Consistent with US merger doctrine (Altai, Lotus / Borland 1995, etc.)



- **But** : What is an “idea” ? a “functionality” ? How to identify the boundary between idea and expression?
 - AG’s Opinion : functionality = the “service” (the result) that the user expects from the computer program (§52)
 - the Court’s decision does not present any method to follow in order to distinguish between a protectable form and a non-protectable function/idea



→ In the US it is sometimes criticized : Laddie, Prescott and Vitoria, *The Modern Law of Copyright and Designs* (3rd ed., 2000), nos. 3.74 to 3.80, deploring “the ‘idea-expression fallacy’”: “the maxim is obscure or, in its broadest sense, suspect”



Franck Macrez



- “Functionality”...
 - 1. Expected result : excluded from protection (comp. with the concept of “technical function” in the field of patent law)
 - 2. Process followed to obtain it: form describing the process to the machine: structure, arrangement, combination...
 - A form is present (i.e. the structure of the writing) : should be excluded only because not original



II.- B – Copyrightable Forms

1 – The Relative Scope of the Exclusion of Ideas

2 – Existing Protection Possibilities: “Ordinary Law” and Special Law



Franck Macrez



II.- B.- 1 – The Relative Scope of the Exclusion of Ideas

- Programming languages excluded as an idea ?
 - AG : Yes, programming language is “the means which permits expression to be given, not the expression itself” (§71)
 - ECJ : very elliptical...



- Formats of data files : same reasoning from the Court
 - ➔ ideas that are not sufficiently formalised to qualify for protection: they are a method of organising information... But...



- ... But : express referral to the “ordinary law”...
 - non-protection of ideas cannot be the justification for their exclusion from protection under the special law
 - Maybe the exclusion is based on the fact that (as in the *BSA* judgment concerning graphic user interfaces), the analysed elements are not programs because they do not directly enable the computer to perform its function



II.- B.- 2 – Existing Protection Possibilities: “Ordinary Law” and Special Law

- Programming languages and formats of data files =
Fallback protection under the “ordinary law” of copyright
(*SAS*)
- GUI : Not a (form of expression of a) program (*BSA*)

But...



But : an interface is in fact a program as such, being made up of source code and object code

➔ The *graphical expression* of the GUI is not a program : confusion between both in AG and ECJ's minds ?

And : Programming languages may include reproduction of software libraries, compilers or interpreters... i.e. : programs (in itself ? not an accessory to the program)



CONCLUSION

- The Court of Justice's main contribution concerns the method adopted by it. Its distributive classification highlights the complex character of software creations



Franck Macrez



- Yet the method's implementation proves to be tricky :
 - complexity of the subject-matter

 - + legislative gap
- the “ordinary law” that the Court of Justice endeavours to construct has no textual basis)



- The statement of principle concerning the **non-appropriation of ideas** in the *SAS* decision testifies to the Court's intention to show a true vision.

→ the policy stated by it is destined to extend beyond the strict framework of copyright: the non-appropriation of ideas should be seen as a *general principle of law* (Macrez 2009, 2012)



→ **And:** *generalia specialibus non derogant*: the principle should apply to other branches of law:

- theory of parasitic conduct (“free riding”) / unfair competition

But, for instance in France : C. cassation, 13 déc. 2005, n°03-21.154, Microsoft et Softimage :

“functionality of a software (...) is not protected, as such, by copyright as it represent only an idea”.
But...

« faulty parasitic behavior » due to a « misuse of know how »



- patent law...

But the EPO may not agree... Is it consistent with its “computer-implemented inventions” doctrine ? :-/

→ BTW, same problem about GUIs : EPO, , T-928/03, 3.5.1, Konami (Technical consideration: the interactions between the player and the computer are more efficient)

→ What interaction between the ECJ and the future UPC ?...



Thank you

franck.macrez@ceipi.edu
[@FranckMacrez](#)



Franck Macrez



Further readings (in English only, far from exhaustive)
CJUE, C-393/09, 22 déc. 2010, Bezpečnostní
softwarová asociace:

- *Computer and Telecommunications Law Review* 2011, 17(3), p.70, L. J. Smith
- *RIDA* 2011, no. 227, chron. P. Sirinelli

CJEU, 2 May 2012, case C-406/10, SAS Institute
Inc. v. World Programming Ltd, RLDI

- D. Gervais and E. Derclaye : *EIPR* 2012, 34(8), p. 565
- Nickless, David: *Journal of Intellectual Property Law and Practice* 2012 p.709-711 (EN)
- Onslow, Robert ; Jamal, Isabel: *EIPR* 2013 p.352-356 (EN)



On both decisions (personal advertising 😊)

Macrez, Franck : « Copyright, computer programs and the Court of Justice / El derecho de autor, el programa de ordenador y el Tribunal de Justicia / Le droit d'auteur, le programme d'ordinateur et la Cour de Justice », *Revue Internationale du droit d'auteur*, oct. 2012, p. 189-287

Full paper at : <http://franck.macrez.net/?p=264>



Franck Macrez

