

令和元年11月13日

関係者各位

株式会社カンバス

不正競争防止法違反に基づく差止請求等訴訟に関するお知らせ

株式会社カンバス（東京都千代田区神田神保町1-50。以下「弊社」といいます。）は、標記の件につきまして、謹んでご連絡申し上げます。

弊社は、株式会社フェイスらを被告とし、弊社の重要な営業秘密である字幕制作ソフトのソースコードを不正に取得、使用して同社が販売する「Babel」（以下「Babel」といいます。）が製作されていることを理由として、不正競争防止法違反に基づき、前記「Babel」の製造・使用・販売等の禁止、損害賠償を求めて訴訟を提起し、（東京地裁平成27年（ワ）第16423号。）この訴訟に勝訴致しました。しかるに、控訴審（知財高裁平成30年（ネ）第10092号。）は、令和元年8月21日、当該地裁判決を破棄し、弊社の主張を斥けたため、弊社は上告の受理を申し立て、令和元年10月23日、上告受理申立理由書を最高裁判所に提出致しました（上告受理申立理由書の全文を添付致します）。ここに上告受理申立理由書の理由全文を公開致します。

控訴審判決の不当性については、ソフトウェア開発企業やエンジニア、字幕制作会社の皆様におかれて添付理由全文をご覧頂ければ容易にご理解頂けるものと考えております。例えば、SSTG1もBabelも「ITALIC_SYNC」と「ITALIC_ASYNC」という普通ではありえない文字列をルビにもった字幕データをインポートすると、それらルビが消去されるという共通の事象が発生します。このような機能は字幕制作ソフトには不要であり、なぜ裁判所がこの事象を決定的な証拠としないのか甚だ疑問です。

映像コンテンツという知的財産に関わる字幕業界の方々に、その知識と経験並びに良心に基づき、本件についての公平なご判断頂戴致したく、本書及び上告受理申立理由書を公開させて頂くことに致しました。

末筆ながら、今後ともSSTG1シリーズをご愛顧の程、よろしくお願い致します。

<この件に関するお問い合わせ先>

株式会社カンバス 担当：執行役員 榎谷（ますたに）

E-mail : sales@canvass.co.jp

TEL:03-6811-0806（平日 9:30~18:30）

上告受理申立事件番号・令和元年（ネ受）第10020号

不正競争行為差止等請求上告受理申立て事件

上告受理申立人 株式会社カンバス

相手方 株式会社フェイス、渡辺進一

上告受理申立理由書

令和元年10月24日

最高裁判所 御中

上告受理申立人

訴訟代理人弁護士

同



第1 本事案の争点及び上告受理申立理由の概要

1 第一審は、3年以上もの長期にわたる慎重な審理を経て判決に至ったものであるが、ソースコードの対比という鑑定が実施され、その鑑定結果に基づき、原告ソフトウェアと被告ソフトウェアのそれぞれのソースコードの類似箇所の営業秘密該当性及び不正競争該当性を認定し、一審被告フェイス及び一審被告渡辺に対し、被告ソフトウェアの生産、使用、譲渡、貸し渡し又はその譲渡若しくは貸渡しの申出等を禁止し、被告ソフトウェアのプログラムを収納したフロッピーディスク、CD-ROM、ハードディスク等の記憶媒体の廃棄を命じ、本件ソースコードの使用を禁止し、本件ソースコードを記録したフロッピーディスク、CD-ROM、ハードディスク等の記憶媒体の廃棄を命じ、さらに、損害賠償を命じた。

これに対し、控訴審は、口頭弁論を1回開いたのみで、第一審の認定判断及び鑑定結果を無視して独自の認定を行い、一審被告らが本件ソースコードを使用したとは評価できないとして、第一審判決を取り消し、上告受理申立人（一審原告）の請求を棄却した。

2 しかし、控訴審判決は、一審被告らが本件ソースコードを使用して被告ソフトウェアを開発したかどうかという争点に対する唯一の直接証拠である鑑定結果について、自らが事実認定した鑑定書の記載と矛盾した判断を下すなど（後記第3）、鑑定結果の認定判断を誤り、実質的には前記争点に対する唯一の直接証拠を排除している。

鑑定結果の詳細は、第一審及び控訴審においても詳述したとおりであるが、前記争点に対する鑑定人の意見が端的に示されている箇所として、以下の点を列挙することができる。

「原告ソフトウェアと被告ソフトウェアの開発者が同一人物であったとしても、これほど共通点の多いソースコードになることは不自然であり、原告ソフトウェアを参照したと考えるのが自然である。該当する箇所で、SSTG1とBabelのソースコードに共通して見られる特徴を説明する。」（鑑定書24頁）

「…仮に、原告ソフトウェアと被告ソフトウェアの開発者が同一人物で、その人

物の記憶を手がかりとしても、原告ソフトウェアのソースコードを参照せずに、これほど細かい特徴を一致させるのは難しいのではないかと考える。…

表1.3で示した共通点に加え、ここで示した(3)–(7)の共通点から、原告ソフトウェアと被告ソフトウェアの開発者は同一人物であると考えられる。さらに、表1.3で示した共通点が多いこと、(6),(7)の共通点を鑑みると、被告ソフトウェアが原告ソフトウェアを参照して開発されたと考えるのが自然である。」(同25頁)

「この仮定を置くと、原告ソフトウェアの開発者と被告ソフトウェアの開発者は別人ではなく必ず同一人物である。被告ソフトウェアを開発する際に原告ソフトウェアを参照したと考えるか、それ以外の特別な事情がない限り、この類似箇所[類似箇所4]が生み出されることはない。…それぞれ1から5単語の英単語の連結で命名された変数が、大文字・小文字や単語の並び順の相違なく、完全に一致するというのは、同一人物の記憶を頼りにしたとしても不自然である。」(同26頁)

にもかかわらず、控訴審判決は、鑑定結果を独自に解釈し、鑑定結果の証明力を減殺する証拠が何ら存在しないにもかかわらず、前記引用にかかる鑑定結果を全て排除し、「一審被告らが本件ソースコードを使用したとは評価できない」との結論を導いている。

このように、控訴審判決は、唯一の直接証拠である鑑定結果を実質的に採用せず、鑑定結果とは真つ向反する独自の認定判断を行っているが、かかる認定判断は経験則に違背したものであり、かつ、採証法則にも違背している。これらの違背は、結論に直接かつ重大な影響を及ぼすことが明らかな法令解釈の誤りとして、法令の解釈に関する重要な事項を含んでいる(民事訴訟法318条1項)。

- 3 補足すれば、第一審が同種事案としては異例ながらも鑑定に踏み切ったのは、長期にわたる慎重な審理の結果、本件訴訟に至った背景事情、間接事実等を総合勘案したことによるものであり、鑑定人もまた、かかる事情を十分に理解して鑑定を行っている。被告ソフトウェアは平成25年2月1日より販売開始されたが、一審被告らは当

業者において当然に履歴管理されている開発当時のソースコードを保持していないなどと主張したため、鑑定の対象としては、せめて、開発開始後約2年が経過した前訴（別件の著作権侵害訴訟）の審理期間中に提示された被告ソフトウェアの改訂版とせざるを得なかった。これは、一審被告らが自認するとおり、バージョンアップが重ねられた後のもの（「Babel」バージョン2.0.0.39）であり、言うまでもなく、原告ソフトウェアとの類似箇所は初版より大幅に削減されている。

この点、控訴審判決の当該認定判断は、控訴審判決書末尾40頁に示されるように、第一審判決の認定が過大だという認識が背景にあることが端々に窺われるが、これは単なる感覚的なものであり、何ら不正競争行為の認定の障害となるものではない。控訴審判決は、鑑定の対象となった被告ソフトウェアのソースコードが、開発開始後約2年が経過し、改変が重ねられているという重大な事実を看過しているものと推認される。被告ソフトウェアの開発が本件ソースコードの使用によって成り立っている以上、本件ソースコードという営業秘密の不正な取得がなければ、そもそも被告ソフトウェアは製品として存在しないのであり、経時的にバージョンアップ等されることによってソースコードの類似箇所が漸減していったとしても、不正競争防止行為が治癒されるものではない。第一審は、この点も熟慮し、鑑定事項を慎重に検討し、また、鑑定人においても、これらの背景事情を十分に理解し、前記のとおり専門家の経験則に基づいて鑑定結果を述べているにもかかわらず、控訴審判決は、何ら合理的な理由も無く、唯一の直接証拠である鑑定結果を排斥しているのである。

このほか、控訴審判決の認定判断には、結論に直接かつ重大な影響を及ぼすことが明らかな鑑定結果の誤認及び理由の齟齬があり、これもまた、結論に直接かつ重大な影響を及ぼすことが明らかな経験則違背であるという法令解釈の誤りがある。この点については、後記第3において詳述する。

- 4 そして、控訴審判決は、訴訟に至る経緯等の重大な事情を全く理解せず、第一審において鑑定が採用された経緯や3年にも及ぶ慎重な審理経過を全て無視していることを指摘しなければならない。

控訴審判決は、僅か1回の口頭弁論をもって第一審判決を取り消したが、その拙速な審理経過からも窺われるように、鑑定結果を実質的に排除しながら第一審判決を全部取り消しており、余りに不意打ちである。このように、控訴審判決は、自らの誤解について当事者にその是正の機会を与えることなく、第一審判決を取り消し、一審原告に著しく不利益な判決を下したものであり、手続の適正という手続法上の基本原則に違背した法令違反がある（民事訴訟法318条1項）。

第2 唯一の直接証拠である鑑定結果の排除

- 1 前記のとおり、控訴審判決は、唯一の直接証拠である鑑定結果とは真っ向反する独自の認定判断を行っているが、かかる認定判断は経験則に違背したものであり、かつ、唯一の直接証拠を何ら正当な理由も無く実質的に排除するものであって、唯一の証拠法則の排除を違法とする最高裁判決（最高裁判所第一小法廷昭和53年3月23日判決・最高裁判所裁判集民事123号283頁）の趣旨にも違背する。これらの違背は、結論に直接かつ重大な影響を及ぼすことが明らかな法令違反であり、控訴審判決は破棄を免れない。

他方、第一審判決は、不正競争行為の認定判断について、鑑定結果に基づき、原告ソフトウェアと被告ソフトウェアのそれぞれのソースコードの類似点を認定した上で、鑑定意見同様、当該類似点が一審被告らの不正競争行為によって作出されたものであることを認定している。しかし、控訴審判決は、鑑定結果を独自に解釈し、前記引用にかかる鑑定結果を全て排除し、「一審被告らが本件ソースコードを使用したとは評価できない」との結論を導いているのである。

- 2 控訴審判決は、鑑定意見の記載事項について、控訴審判決書19頁～26頁において縷々述べているが、前記にも引用したように、「原告ソフトウェアと被告ソフトウェアの開発者が同一人物であったとしても、これほど共通点の多いソースコードになることは不自然であり、原告ソフトウェアを参照したと考えるのが自然である。」（鑑定書24頁）、「仮に、原告ソフトウェアと被告ソフトウェアの開発者が同一人物で、

その人物の記憶を手がかりとしても、原告ソフトウェアのソースコードを参照せずに、これほど細かい特徴を一致させるのは難しいのではないかと考える。…被告ソフトウェアが原告ソフトウェアを参照して開発されたと考えるのが自然である。」（同25頁）、「被告ソフトウェアを開発する際に原告ソフトウェアを参照したと考えるか、それ以外の特別な事情がない限り、この類似箇所が生み出されることはない。…それぞれ1から5単語の英単語の連結で命名された変数が、大文字・小文字や単語の並び順の相違なく、完全に一致するというのは、同一人物の記憶を頼りにしたとしても不自然である。」（同26頁）という、各類似箇所に関する専門家の意見については、その一切を不正競争行為の成否の判断において無視している。換言すれば、一審被告らの不正競争行為の成否を裏付ける、唯一の直接証拠である鑑定意見を排斥するのであれば、その正当な理由、特段の事情を裁判所は示さなければならないが、その正当な理由は、控訴審判決書には一切見当たらない。

なお、鑑定人が認定した各類似箇所について、控訴審判決による事実誤認が経験則に違背することについては、後記第3において別途詳述する。

3 控訴理由書においても補充で主張立証したとおり、鑑定において認定された類似箇所は、その鑑定の手法から来る制約から限定的なものである。鑑定書は、原告ソフトウェアと被告ソフトウェアのそれぞれのソースコードにおける類似箇所や共通性を全て網羅的に指摘できたわけではなく、鑑定人が実施した鑑定手法によって発見された部分のみを指摘したものであることを十分に理解しなければならない。それでもなお、鑑定人は、前記引用を始め、鑑定書の随所において、類似箇所の認定及び本件ソースコードの使用について専門家として認定した事実及び意見を述べているのである。

第一審は、3年にもわたり、背景事情、間接事実その他一審被告らの訴訟態度等弁論の全趣旨を勘案し、鑑定書の認定内容を網羅的に吟味し、不正競争行為の認定を行っているが、かかる専門家の認定判断を、控訴審判決が何ら正当な理由も無く、鑑定意見の主要部分ではない一部を抽出して異なる結論を導くことは、経験則に違背する事実誤認のみならず、手続違背の法令違反があると言わざるを得ない。

あくまでも、鑑定書は、その鑑定の手法による限りにおいて、類似箇所1ないし5を指摘するものであり、甲第143号証で別の専門家（鑑定人が鑑定において参照した文献の作成者）も指摘しているとおおり、類似箇所1ないし5以外の部分について、「被告らによる使用等はなかった」と認定判断することはできない。実際、一審原告が証拠（甲第109号証の1、2及び甲第110号証の1～4）を掲げつつ、第一審における一審原告の第7準備書面において主張した事実（原告ソフトウェア及び被告ソフトウェアのいずれも、「ITALIC_SYNC」と「ITALIC_ASYNC」という文字列をルビにもった字幕データをインポートすると、それらルビが消去されるという、字幕制作ソフトでは全く必要がなく、かつ、ユーザーも認識し得ない共通プログラムを具備すること）が鑑定では指摘されていないのである。

また、鑑定書は、SSTDB.cppファイルとMdb.cppファイルが類似・共通すると指摘しており（類似箇所4）、この両ファイルは、表1.11のSW基準では、全ファイル中類似度が1位であり、STTDB.cppファイルの行数は優に3000行を超えるものであるから、類似箇所が限定的かつ少量であるかのように誤解してはならない。加えて、控訴審判決は、後記第3記載のとおり、認定した事実と矛盾する類似箇所4の解釈を失っている。一審原告が主張立証してきたように、一審被告らが本件ソースコードに基づいて被告ソフトウェア全体を開発したことは明らかであり、前記甲第143号証6頁4項以下に記載された前記専門家の意見もそれを裏付けるものである。

- 4 以上のとおり、控訴審判決は、一審被告らの不正競争行為の成否を裏付ける唯一の直接証拠である鑑定結果について、何ら正当な理由又は特段の事情を認定することもないまま、かつ、当事者に何ら反駁等の機会を与えることもないまま、鑑定結果に真向反する独自の認定判断を行っている。かかる認定判断は、専門家による鑑定結果に対して裁判所が異なる結論を導いているものであり、直接かつ重大な影響を及ぼすことが明らかな経験則違背であり、採証法則に関する手続違背という法令違反がある。

なお、被告ソフトウェアのソースコードには、原告ソフトウェアにかかる本件ソースコード上のバグまでもが存在するという重大な間接事実がある。これは、経験則上、

本件ソースコードをコピーして被告ソフトウェアが製作されたことを強く推認するものであるが、控訴審判決は、判決の結論に相容れない重要な事実認定を回避している。

第3 経験則に違背する事実誤認

1 はじめに

前記第2記載のとおり、控訴審判決は、一審被告らの不正競争行為の成否を裏付ける唯一の直接証拠である鑑定結果について、何ら正当な理由又は特段の事情を認定することもないまま、かつ、当事者に何ら反駁等の機会を与えることもないまま、鑑定結果に真っ向反する独自の認定判断を行っているが、かかる控訴審判決の認定判断は、専門家による鑑定結果に対して裁判所が異なる結論を導くものであり、明白な経験則違背がある。

以下、一審被告らが本件ソースコードを使用して被告ソフトウェアを開発したかどうかという争点に直接関連する、鑑定人が認定した各類似箇所について、控訴審判決による事実誤認が経験則に違背することを明らかにする。

2 類似箇所1について

まず、類似箇所1について、鑑定結果によると、「Babel のソースコード SourceDefault.h で宣言されている変数30個のうち、20個の宣言が型、コメント、インデントも含めて SSTG1 のソースコード GlobalSettings.h のものと完全に一致している。残りの10個のうち、5個では少なくとも変数の名前が SSTG1 のものと一致している。」（鑑定書4頁）、「Babel のソースコード SourceDefault.h の `_FontName` 変数の宣言は、SSTG1 のソースコード GlobalSettings.h のものと、空白部分も含めて完全に一致していることを表している。詳しくは1.3節で説明するが、同一人物がコーディングを行ったとしても、これだけの一致が起こるのは不自然である。」（同5頁）とされ、一審被告らが本件ソースコードを使用したことが認定されている。これに対し、控訴審判決は、「本件ソースコードの変数定義部分を参照した可能性が否定できないとしても」としながら、「そのことをもって、一審被告らが本件ソ

ソースコードを使用したとは評価できない」などと、一見して矛盾した認定判断を行っている（控訴審判決書30頁16行～18行）。

しかし、「ソースコード」は、不正にアクセスしなければ内容を解読することのできない高度な営業秘密であり、一審被告らがこれを「参照した」ことは、本件ソースコードに不正に取得し、使用したことに他ならない。このように、控訴審判決は、「ソースコード」そのものが営業秘密であることを看過し、その結果、前記のような経験則に違背した重大な事実誤認を犯しているものであり、これは明らかに経験則の適用を誤った法令違反に該当する。なお、控訴審判決は、類似箇所1について、「類似箇所1に係る本件ソースコードの情報の内容（変数定義）自体は、少なくとも有用性又は非公知性を欠き、営業秘密とはいえない」（控訴審判決書30頁8行～9行）と認定判断しているが、これもまた、鑑定意見を無視した経験則に違背するものであり、この点についての詳細は、第一審における一審原告の第7準備書面16頁～19頁に詳述したとおりである。

3 類似箇所2及び3について

次に、類似箇所2について、鑑定結果によると、「Babel のソースコード SourceDefault.cpp が実装する CsourceDefault クラスの代入演算子のオーバーロードと、SSTG1 のソースコード GlobalSettings.cpp の CGlobalSettings クラスの代入演算子のオーバーロードの実装に共通点が見られる。…この表 [表 1.5] で示したとおり、演算子のオーバーロードで参照される30個の変数のうち、21個の変数では変数名と注釈の一致が見られた。同様に、GlobalSettings.cpp の側から SourceDefault.cpp で実装されている代入演算子のオーバーロードを分析した結果を表 1.6 にまとめた。詳しくは 1.3 節で説明するが、同一人物がコーディングを行ったとしても、これだけの一致が起こるのは不自然である。」（鑑定書6頁）とされ、一審被告らが本件ソースコードを使用したことが認定されている。また、類似箇所3について鑑定結果によると、「Babel のソースコード SourceDefault.cpp が実装する CsourceDefault クラスの比較演算子のオーバーロードと、SSTG1 のソースコード

GlobalSettings.cpp の CGlobalSettings クラスの比較演算子のオーバーロードの実装に共通点が見られる。…この表 [表 1.7] で示したとおり、演算子のオーバーロードで比較される 29 個の変数のうち、20 個の変数では変数名と注釈の一致が見られることが分かった。同様に、GlobalSettings.cpp の側から SourceDefault.cpp で実装されている比較演算子のオーバーロードを分析した結果を表 1.8 にまとめた。詳しくは 1.3 節で説明するが、同一人物がコーディングを行ったとしても、これだけの一致が起こるのは不自然である。」(同 7 頁～8 頁) とされ、一審被告らが本件ソースコードを使用したことが認定されている。

これに対し、控訴審判決は、「類似箇所 2, 3 は、類似箇所 1 に基づいて発生したものと解される」(控訴審判決書 30 頁 23 行～24 行) とした上で、「類似箇所 2, 3 が、類似箇所 1 とは別個に生じた類似箇所であることを認めるに足りる証拠はなく、類似箇所 2, 3 によって、一審被告らが本件ソースコードを使用したことを推認することはできない」(同 31 頁 19 行～21 行) と認定判断している。しかし、かかる認定判断は、前記類似箇所 1 に基づく誤謬を前提として、控訴審判決自身の論理に基づいて導かれたものであるから、類似箇所 1 の認定判断において経験則違背の違法がある以上、類似箇所 2, 3 の前記認定判断も論理必然的に経験則の適用を誤った法令違反に該当する。なお、類似箇所 2 及び 3 が営業秘密に該当することについては、第一審における一審原告の第 7 準備書面 19 頁～21 頁に詳述したとおりである。

4 類似箇所 4 について

- (1) 類似箇所 4 について、鑑定結果によると、表の 1. 2 (鑑定書 4 頁) で SSTDB.cpp と Mdb.cpp のファイル同士の類似であることが示された上で、「Babel のデータベースで用いられている 52 件のフィールドの名前は、すべて SSTG1 のデータベースで用いられているものと同じである。」(鑑定書 9 頁)、「SSTG1 は 63 件のフィールド名を、Babel は 52 件のフィールド名を用いているが、前者が後者を完全に包含する関係にある。詳しくは 1.3 節で説明するが、同一人物がコーディングを行ったとしても、これだけの一致が起こるのは偶然ではなく、何らかの要因 (例えば Babel と

SSTG1 で互換性を持たせたかった) があると考えられる。」(同10頁)とされ、さらに、「2.4.3 項で説明するように mdb ファイルのレコード名(テーブル名)、フィールド名を調べることは比較的容易である。ただし、各フィールドがどのようなセマンティクスを持つのか正確に把握することは、容易なことではない。例えば、iGlobOrientation フィールドが格納している整数値のセマンティクス[当該整数値をソースコード上で、字幕データの一部に分析・解釈するアルゴリズム。この場合、当該整数値がどの字幕の位置を表すかを解釈する。]はかなり複雑である。また、被告ソフトウェアが読み書きするフィールドは、原告ソフトウェアのものと比較すると少ない。なぜ、原告が少ないフィールドだけで互換性を持たせることができると考えたのか、その理由も不明である。」(同27頁)とされ、一審被告らが本件ソースコードを使用したことが認定されている。なお、類似箇所4が営業秘密に該当することについては、第一審における一審原告の第7準備書面21頁～30頁に詳述したとおりである。

前記鑑定結果が示す「類似箇所4」とされる SSTDB.cpp は、Template.mdb に字幕データをセマンティクスに従って記述、保存する機能、Template.mdb 形式の字幕ファイル(.mdb)からセマンティクスに従って字幕データを解析する機能、及び、Template.mdb 形式の字幕データ(.mdb)をセマンティクスに従って原告ソフトウェアのデータメモリ領域に展開する機能を有するものであり、鑑定書は、「各フィールドがどのようなセマンティクスを持つのか正確に把握することは、容易なことではない。例えば、iGlobOrientation フィールドが格納している整数値のセマンティクスはかなり複雑である。」(鑑定書27頁)と述べているのである。にもかかわらず、控訴審判決は、iGlobOrientation フィールドよりも更に複雑な strFormat フィールドについての一審原告の主張を無視し(一審原告の前掲第7準備書面21頁の「4. 類似箇所4の SSTDB.cpp の営業秘密性」及び控訴理由書7頁参照)、いとも容易く、一部のフィールドのみ都合良く例示的に引用し、「かかるセマンティクスは、本件ソースコードを使用しなくても把握可能であるものと認められる」(控訴審判決書34頁～35頁)

と判示するものであるが、如何なる経験則をもって、前記鑑定書の見解を裁判所が否定する論拠は無い。

- (2) そもそも、鑑定人が、鑑定書26頁～27頁「類似箇所4」において述べていることは、原告ソフトウェアと被告ソフトウェアがmdbファイルで互換性が取れるのは、Template.mdbの各フィールドの名称という形式的な一致ではなく、各フィールドの複雑なセマンティクスが一致しているからである、ということである。これは、一審原告の主張、つまり、Template.mdbは本件ソースコード上の定義を基準として動作し、本件ソースコードはTemplate.mdbの情報の選択と体系的な構成に依拠することでしかmdbをエクスポートすることもインポートすることもできず、一審被告らがTemplate.mdbを複製し、利用し、頒布しているということは、本件ソースコードに記載された定義を参照しつつ、当該データが何を示すか、当該データが何を意味するかという情報(鑑定書が述べるセマンティクス)を使用したことと同義である。仮に、当該類似箇所4が鑑定書「表1.9」(上記レコード名、テーブル名、フィールド名のみ)だけの類似であるとするならば、鑑定人はこのような「セマンティクス」の議論を持ち出すまでも無く、控訴審判決同様、単に「類似箇所4は本件ソースコードを使用しなくても把握可能である」と述べることは明白であろうが、鑑定人は「mdbファイルはMicrosoftAccessで開くことができるため、原告ソフトウェアが生成するように、mdbファイルのレコード名(テーブル名)、フィールド名[表1.9の部分]を調べることは比較的容易である」(鑑定書27頁)としつつ、「セマンティクス」の概念を持ち出し、前記のとおり「ただし、各フィールドがどのようなセマンティクスを持つのか正確に把握することは、容易なことではない」(同頁)と述べているのである。加えて、鑑定人は「被告ソフトウェアを開発する際に原告ソフトウェアを参照したと考えるか、それ以外の特別な事情がない限り、この類似箇所が生み出されることはない」(鑑定書26頁18行。控訴審判決書25頁においても認定されている。)とも鑑定している。

つまり、鑑定人は、「表1.9」(上記レコード名、テーブル名、フィールド名の

み。控訴審判決が類似箇所4として認定した部分。)は、mdb ファイルを MicrosoftAccess で開けば調べられるので、その表記自体の類似はソースコードを参照せずとも実現可能であろうが、各フィールドがどのようなセマンティクスを持つのかを正確に把握するのは容易ではないところ、なぜか一審被告らは当該セマンティクスを正確に把握しており、これは、原告ソースコードファイル SSTDB.cpp (当該セマンティクスの記述がある)と被告ソースコードファイル Mdb.cpp の類似を意味していると指摘しているのである(鑑定書4頁の表1. 2の記載どおり)。

この点、控訴審判決は、「一審原告は、セマンティクスとは、解析アルゴリズムであると主張する。しかし Template.mdb は mdb 形式のファイルを作成するためのひな型であり、プログラムではないから、そのセマンティクスに解析アルゴリズムが含まれるとは解されず、一審原告の主張は採用できない。」(控訴審判決書32頁25行～33頁2行)と判示しながら、その直後の34頁2行目以降において、Template.mdb の strFormat フィールドの解析アルゴリズムを「セマンティクス」として認定した上で「セマンティクスを把握することができる」と判示し、セマンティクス自体の解釈、認定において相容れない判断を示し、理由の齟齬を来している。このことから、控訴審判決は、あれほどまでに第一審から議論されてきた、鑑定書の示す「セマンティクス」を全く理解しないまま、鑑定意見を排斥していることが容易に窺われるのである。

- (3) 控訴審判決書(24頁6～8行目)が鑑定書の記載内容として認定した、前記引用にかかる鑑定書26頁18行の「被告ソフトウェアを開発する際に原告ソフトウェアを参照したと考えるか、それ以外の特別な事情がない限り、この類似箇所[類似箇所4]が生み出されることはない」という記述は、フィールド名だけ(表1. 9=第一審判決書別紙d)の類似であれば、鑑定人も「調べることは比較的容易」と述べている以上、フィールド名の類似(表1. 9=第一審判決書別紙d)ではなく、ソースコードの観点から見たファイルの類似であることは自明である。この点、鑑定書表1. 2にもファイルの類似であると記載されている。

他方、控訴審判決は、「フィールド名を調べることは比較的容易」（控訴審判決書 25 頁 12 行）という点、及び、前記「被告ソフトウェアを開発する際に原告ソフトウェアを参照したと考えるか、それ以外の特別な事情がない限り、この類似箇所〔類似箇所 4〕が生み出されることはない」という点を、鑑定書記載の事実として認定しておきながら、この 2 つの認定事実と矛盾して「類似箇所 4 は、原判決別紙 d のとおりである」と判断し、単に「フィールド名」だけの一致、換言すれば、鑑定人でなくても一般通常人でも Microsoft Access で開けば調べられる点を捉え、鑑定結果を全て排斥し、類似箇所 4 は「本件ソースコードを使用しなくても把握可能である」などと極めて稚拙な認定を行っているのである。このように、自らが認定した事実と矛盾する類似箇所 4 についての控訴審判決の認定判断には、理由の齟齬があり、かつ、極めて基本的かつ重大な点において経験則に違背する重大な誤謬がある。

このように、控訴審判決は、唯一の直接証拠である鑑定結果の、しかも、最も基本的かつ重要な記載部分を全く理解せず、類似箇所 4 が一審被告らによる本件ソースコードの使用を裏付けるものであるという鑑定結果を、何ら合理的かつ具体的な理由も示さずに採用しなかったのである。そのため、控訴審判決は、控訴審判決書 35 頁 18 行「本鑑定において、鑑定人は、原告ソフトウェアの SSTDB.cpp と被告ソフトウェアの Mdb.cpp との目視確認を行った上で、類似箇所は 52 個のフィールド名にあると鑑定した」、同頁 25 行「被告ソフトウェアにおいて、本件ソースコードを参照して原告ソフトウェアの解析アルゴリズムを把握し、同じ処理を行っていることを認めるに足りる証拠はない」などという、自身の事実認定にも矛盾した事実誤認を犯しているものである。

- (4) 以上のとおり、控訴審判決には、経験則に違背した明らかな法令解釈があり、法令の解釈に関する重要な事項を含むものと認められる事件であることから、民事訴訟法 318 条 1 項に規定する上告受理申立理由に該当する。

第 4 結論

以上のとおり、控訴審判決には、明らかな経験則違背及び手続適正の法令違反並びに判例違反があり、法令の解釈に関する重要な事項を含むものと認められる事件であることから、民事訴訟法318条1項に規定する上告受理申立理由に該当し、直ちに破棄されるべきである。

以上