

No. 138(2014/8)

ORACLE AMERICA, INC v. GOOGLE INC
米連邦控訴審裁判所 (CAFC) 2014 年 5 月 9 日判決¹
～アプリケーションプログラミングインターフェースの著作物性が肯定された事例～

弁護士 石新智規

◆ 目次

- 1 はじめに ... 1
- 2 事件の経緯 ... 2
- 3 論点 ... 4
- 4 CAFC の判断 ... 4
 - (1) Java API の著作物性について ... 4
 - ① Declaring Code に対するあてはめ
 - ② API パッケージの SSO について
 - (2) フェアユースについて ... 9
 - ① コンピュータの互換性を達成する必要性と著作物性の関係性
 - ② Google の Java API の複製はフェアユースか
- 5 コンピュータプログラムの保護範囲を巡る裁判例の動向 ... 10
 - (1) Franklin 事件から Paperback 事件まで ... 10
 - (2) Altai 事件から Lotus 事件まで ... 11
- 6 本件の評価 ... 13
 - (1) Baker 判決と著作権法 102 条(b)の解釈 ... 13
 - (2) コンピュータプログラムの法的保護の在り方 ... 15

1 はじめに

近時、米国やヨーロッパにおいて著作権リフォームが注目されている。インターネット時代の本格化した2000年以降に普及したデジタル技術と著作権の対立の止揚が求められている時代が到来していると言えそうである。

¹ Oracle Am., Inc. v. Google Inc., 750 F.3d 1339

歴史的にみて、技術と著作権の対立は、著作権という法概念（又は権利）の誕生以来の宿命であり、米国では、ビデオカセットテープへのテレビ番組の複製がフェアユースかが争われた 1984 年のソニー判決以降、数多くの著作権裁判においてテクノロジーと著作権の調和が論点とされてきた。

特に、ソニー判決後の 1980 年代後半から 90 年代は、Whelan 判決（1986）²がプログラムの文字的(literal)要素であるコードだけではなく、非文字的 (nonliteral) な要素であるコンピュータプログラムの構造 (Structure)、シーケンス (Sequence) 及び組織 (Organization)（以下、「SSO」と略す。）にも広範に著作権の保護を及ぼしたことを契機に、コンピュータプログラム著作権の保護範囲を巡る議論が盛んに行われた時期であった。

SSO に対しても著作権保護を及ぼす Whelan 判決はその後厳しい批判に晒され、その批判を受け入れる形で、Altai 判決（1992）は、Whelan 判決とは異なる、非文字的要素の保護範囲を限定的に解釈する基準を採用し、その基準は多くの支持を得た。³

さらに、Accolade 判決（1992）⁴は、Sega 製のゲームコンソール (Game Console) 上で動作するゲームカートリッジ (Game Cartridge) を製作する過程で、コンソール・カートリッジ間のインターフェースの機能を理解するために、Sega のプログラムをリバースエンジニアリングすることをフェアユースとして許容した。こうした流れの中で、ソフトウェアの著作権による保護は相当程度厳格に解釈されるようになり、SSO に法的保護を求める者は特許権による保護を求めるべきとの思潮が一般的なものとなっていた。

ところが、本判決は、すでに定着したと考えられていた裁判例の動向を踏まえ API（アプリケーションインターフェース、以下「API」という。）の SSO の著作物性を否定した原判決⁵の判断⁶を覆したため、驚きをもって受け止められている。⁷

2 事件の経緯

2005 年、Google は、携帯電話用のプラットフォーム構築を計画し、そのプラットフォーム構築のためにオープンソースの Java のライセンスを著作権者である Sun Microsystem と協議した。しかし、ライセンスの妥結には至らず⁸、Google は独自のプラットフォーム構築

² 【日本・アメリカ】コンピュータ・著作権法（デニス＝カージャラ・楢山敬士共著、日本評論社（1989）、「カージャラ＝楢山」）168 頁以下

³ Dennis S. Karjala, A Coherent Theory For The Copyright Protection of Computer Software and Recent Judicial Interpretations, 66 U. Chi. L. Rev.53, 83 は、Altai 判決「に続くほとんど全ての裁判所がその（Altai 判決の）理由づけと分析手法を受け入れ、従おうとしている」と評価する。

⁴ *Sega Enters. Ltd. v. Accolade, Inc.*, 977 F.2d 1510, 1525 (9th Cir. 1992)

⁵ Appeals from the United States District Court for the Northern District of California in No. 10-CV-3561, Judge William H. Alsup.

Oracle Am., Inc. v. Google Inc., 872 F. Supp. 2d 974, 2012 U.S. Dist. LEXIS 75896 (N.D. Cal., 2012)

⁶ 著作物性に関する原決定（CAFC が Copyrightability Decision と呼ぶ決定）は、Whelan 判決から Accolade 判決にいたる過去の裁判例を第 9 巡回区の裁判例に限らず広く検討し、結論を導いている。

⁷ Altai @ 21: Software Copyrights Revisited, <http://www.law.berkeley.edu/15775.htm> 本判決前から、果たして、CAFC（連邦巡回区控訴裁判所）が回帰的にプログラムの SSO を保護するのかどうか注目が集まっていた。

⁸ 交渉がまとまらなかったのは、Google が Android プラットフォームに Java virtual machine やその他の Java プログラムと互換性を持たせることを拒絶したためである。Sun の “write once, run anywhere” のポリシーが Google の姿勢と合致しなかったようである。

に乗り出した。Google は最終的に Java 言語を利用することに決め、168 の API を開発したが、そのうち 37 は Java API パッケージを実装 (implementation) したものであった。

なお、API とは、一般的なコンピュータの機能を実行させるプログラムを事前に書いておき、それをいくつかまとめてパッケージ化したものである。API を利用することによりプログラマーは、一定の機能を実現するために最初からプログラムを構築する必要がなく、一定の機能を実現するために事前に用意されたプログラムを自らのプログラムに流用することができる。いわばプログラミングのショートカット (近道) をプログラマーに提供するものである。

Google は 2007 年、Android プラットフォームをリリースし⁹、翌年、Android ベースの携帯電話が発売され、300 万台以上の Android 携帯電話が販売された。

2010 年、Sun を買収した Oracle は、Google に対して、カリフォルニア連邦地方裁判所に著作権侵害に基づく訴訟を提起した (当初は特許権侵害も主張していたが¹⁰、その主張は陪審によって否定され、その点について Oracle は不服申立てしていない)。

Google は上記 37 の Java API パッケージを実装する際に、37 のパッケージ、その下に連なる 600 を超える Class、さらにその下の 6000 を超える methods の組織化された体系を全て複製したが (7000 の declaring code の複製を含む)、実装プログラムについては、① rangeCheck 機能と②8 つの逆コンパイルされたセキュリティファイル¹¹を除き、Google が自ら書いていることについて当事者間に争いはない。

2012 年 4 月 16 日に審理が開始され、Java API の著作物性、Google のフェアユースその他の抗弁に関して 24 人の証人尋問が行われた。2012 年 5 月 7 日、陪審は、Google による 37 の Java API と rangeCheck の複製を認め、8 つのセキュリティファイルについては非侵害としたが¹²、フェアユースの抗弁については意見が分かれた。¹³

その後、2012 年 5 月 31 日、カリフォルニア連邦地裁は、Java API のパッケージの SSO には著作物性が認められないと判断し、6 月 20 日、①rangeCheck 機能と②8 つの逆コンパイルされたセキュリティファイルの侵害の点を除き、Google の主張 (著作権侵害不成立) を認める最終的な判断をした。その判断を不服として Oracle と Google 双方が控訴したのが本件である。¹⁴

なお、前述のとおり提訴はカリフォルニアであるが、特許権侵害も主張されているので、その不服申し立てについて CAFC が管轄を持つが、その準拠すべき判例法は第 9 巡回区控訴審裁判所の判例法である¹⁵。¹⁶

⁹ Google は Java 言語を用いたものの、最終的に Android プラットフォームは、Java プラットフォームとは互換性を有さず、一方のプラットフォームで動作するソフトウェアは他方では動作しない。

¹⁰ そのため、著作物性の判断に対する上訴の管轄が、カリフォルニアを管轄する第 9 巡回区控訴審裁判所ではなく、CAFC となった。

¹¹ Google はこの部分の複製は認めたくなくて、de minimis の抗弁を主張した。

¹² この非侵害の認定について連邦地裁は Oracle の異議を認めた。

¹³ 裁判所は陪審員に対して、API の著作物性を前提とした検討をするように指示 (instruction) を与え、著作物性については別途裁判所が検討を進めた。

¹⁴ Google の控訴は、rangeCheck とセキュリティファイルの侵害を認めた点についてであるが、本件の中心争点ではないので、この点は省略する。

¹⁵ この時点で、第 9 巡回区の判例法といえる *Sega Enterprises Ltd. v. Accolade, Inc.*, 977 F2d. 1510(9th Cir.1992) の先例としての位置づけ (CAFC はこの判決をフェアユース判例であると位置づけ、著作物性が唯一の争

3 論点

コンピュータプログラムの著作権法上の保護が及ぶ範囲はどこまでか（102条(a)(b)の解釈）。保護される部分か否かの基準はどのように判断するのが論点である。

米国著作権法上、コンピュータプログラムは保護を受ける。同法は、107条で「コンピュータプログラム」とは「ある一定の結果を生じさせるためにコンピュータにおいて直接的又は間接的に用いられる言明又は指示の組合せ」（A “computer program” is a set of statements or instructions to be used directly or indirectly in a computer in order to bring about a certain result）であると定義しているため、言語の著作物（literary work）として保護される。¹⁷¹⁸

しかし、コンピュータプログラムのどこまでが保護されるのか条文上明確ではない。

37のAPIを実装する際にGoogleが複製した、Java APIパッケージのパッケージ、class及びmethodsの名称とそれが組織化された体系は、冒頭で言及したプログラムのSSO（Structure、SequenceとOrganization）であるところ、Oracleの主張は、GoogleがOracleの許諾なく37のJava APIのdeclaring codeとSSOを無断で複製したことが著作権侵害であるというものであった。

連邦地裁は、Java APIのSSOに創作性・独創性があることを認めながらも、コマンド構造・システム・動作方法（事前に付与された一定の機能を実行するコマンドの階層構造）、すなわちJava APIのSSOは、著作権法102条(b)を理由に著作権による保護を受けないと判断した。

4 CAFCの判断

(1) Java APIの著作物性について

CAFCは、連邦地裁の判断を覆し、Java APIのSSOの著作物性を肯定し、Googleのフェアユース抗弁についてさらに審理をするように地裁に差し戻した。

CAFCは、コンピュータプログラムの著作権の保護範囲について詳細に議論を展開しつつ概要、以下のような点から、Java APIの著作物性（declaring codeというliteral[文字的]な部分だけではなく、SSOというnonliteral[非文字的]な部分についても）を認めるOracleの主張を支持した。

点である本件において重要視していないように思われる。)も重要な意味を持つ。SSOの著作物性を否定するためにGoogleの援用する他の裁判所の判断に対するCAFCの評価（特にLotus判決の軽視）に影響があった。なお、CAFCは、Lotus判決を引用した先例を有する。

¹⁶ 先例として挙げられているのは、CAFCが第9巡回区の判例法の解釈を行った *Atari Games Corp. v. Nintendo of Am., Inc.*, 975 F.2d 832, 838 (Fed. Cir. 1992)である。手続法に明文でその旨を定める規定はなく、判例法としてそのように解釈されている。判決は、当事者も異議を述べていないことについても言及している。

¹⁷ Article 102(a), “Copyright protection subsists, in accordance with this title, in original works of authorship fixed in any tangible medium of expression, now known or later developed, from which they can be perceived, reproduced, or otherwise communicated, either directly or with the aid of a machine or device. Works of authorship include the following categories:

(1) literary works;...” 102条は著作物の列挙規定である。

¹⁸ See *Atari Games Corp. v. Nintendo of Am., Inc.*, 975 F.2d 832, 838 (Fed. Cir. 1992) (“As literary works, copyright protection extends to computer programs.”).

・ *Baker v. Selden*¹⁹判決（以下「**Baker 判決**」）という。）と米国著作権法 102 条(a)(b)項の意義について

著作物性の判断において CAFC は、まず、102 条(a)において最小限の創作性が必要とされるとともに、同 102 条(b)において、著作権法の保護はアイデアではなく、表現にだけ及ぶという原則を確認する。特に、102 条(b)の規定をアイデア／表現二分論を明文化したものの（逆にいえば、それ以上の意味を持たない確認規定である）と理解する。

“In no case does copyright protection for an original work of authorship extend to any idea, procedure, process, system, method of operation, concept, principle, or discovery, regardless of the form in which it is described, explained, illustrated, or embodied in such work.”²⁰

（著作物に対する著作権による保護は、それが表現、説明、描写又は具現化される形態に関係なく、いかなるアイデア、手続、プロセス、システム、動作方法、概念、原則又は発見に対しても及ばない。）

そして、この 102 条(b)は **Baker 判決**によって確立された原則を法文化したものであることを指摘し、**Baker 判決**が、①著作権保護が表現にのみ及び、アイデア、システム、プロセスには及ばないこと、②機能と不可分なコンピュータプログラムの要素は著作権による保護を受けないという原則の法源となっているという。

Google は、問題となっている Java API に 102 条(a)で要求される創作性があることを認める一方、102 条(b)によって、それが機能的なものである場合には著作権による保護を認めないのが **Baker 判決**であり、102 条 (b) であるという立場を採ったが、CAFC は、Google の主張を排斥し、次のように述べた。

Section 102(b) "in no way enlarges or contracts the scope of copyright protection" and that its "purpose is to restate . . . that the basic dichotomy between expression and idea remains unchanged." *Feist*, 499 U.S. at 356 (quoting H.R. Rep. No. 1476, 94th Cong., 2d Sess. 54, reprinted in 1976 U.S.C.C.A.N. 5659, 5670). "*Section 102(b)* does not extinguish the protection accorded a particular expression of an idea merely because that expression is embodied in a method of operation." *Mitel, Inc. v. Iqtel, Inc.*, 124 F.3d 1366, 1372 (10th Cir. 1997).

102 条(b)は、「著作権保護の範囲を拡大ないし制約するものではなく」、そしてその「目的は、アイデアと表現の基本的な区分は変わらないままであることを再度述べること」（引用省略）である。102 条(b)は、表現が動作方法（a method of operation）に具現化されているというだけの理由で、あるアイデアの特定の表現に付与される著作権保護を消滅させるものではない（引用省略）。

他方で、アイデアと表現とを区別する作業が容易ではなく、裁判所の適用する基準が統一していないことを指摘、**Whelan 判決**（3d Cir.1986、「ある作品の目的又は機能に必要とはされないものは全て表現である」）と **Lotus 判決**（9th Cir. 1992、「動作方法（method of

¹⁹ *Baker v. Selden*, 101 U.S. 99, 101(1879) 会計士の Charles Selden は、会計帳簿の独特のシステムの使い方を説明する簡単文章と 20 頁のフォームから成る書籍を発行したが、その死後、その酷似するシステムを発表した Baker に対し、遺族が、書籍及び書籍で示されているシステムの著作権に基づき著作権侵害訴訟を提起したものである。本件については、カージャラ＝楢山 231 頁以下も参照。Baker 判決は、102 条(b)の意義を理解するうえで極めて重要なものである。

²⁰ See, *Golan v. Holder*, 132 S. Ct. 873, 890, 181 L. Ed. 2d 835 (2012) ("The idea/expression dichotomy is codified at 17 U.S.C. § 102(b).").

operation) は、それによって、ユーザーが何かを動作させる手段であり、その動作を実行するために用いられる言語は、著作権による保護を受けない表現である」)を対照的な(両極端の)裁判例として挙げつつ、第9巡回区裁判所の採る判例法は、そのいずれでもなく、**Altai 判決の Abstraction-Filtration-Comparison** テスト(以下「AFC テスト」という。)²¹を採用したという。AFC テストは次の3段階からなるテストである。

- (i) 抽象化ステップ：侵害されたプログラムを構成要素に細分化する。
- (ii) ろ過ステップ：全ての保護されない部分(アイデアだけでなく、あるアイデアと必然的に同一のものとなる表現も)²²²³を除外する。
- (iii) 比較ステップ：ろ過後に残った創作的表現と侵害プログラムと主張されているプログラムの比較を行う。

このAFC テストは、ある機能を実行する対象が動作方法(method of operation)であることをもって著作物性を否定するというLotus 判決の概念を拒絶する(つまり、「動作方法」と分類される場合でも、それにもかかわらず著作物性が肯定される場合もあるという考え方を採る)一方で、同テストは、いったんあるアイデアを分離できれば、その他を全て表現と扱うWhelan 判決の仮定も拒絶するものだとCAFCは論じる。

そのうえでCAFCは、連邦地裁がAFC テストに言及しながら、AFC テストを適用しようとせず、102条(b)に基づき、形態を問わず、機能性のゆえにJava APIが著作権による保護を受けないと判断したことを批判し、AFC テストを適用した。

① Declaring Code に対するあてはめ (A) Merger 理論

前述のとおり、Google による Declaring Code の複製及び同 Code の創作性の点については争いがなかったものの、地裁は、102条(b)(同条に具現化されている Merger【アイデアと表現が混同(一体化)するため、著作権による保護を否定する】理論と短いフレーズ法理)に基づき、著作物性を否定した。すなわち、地裁は、Java ルールに従う以上、プログラマーは同じ機能を指定する方法を示すために、同一の declaration または方法の header lines を利用しなければならないとし、Java パッケージのそれぞれに対する declaring code を書く方法は一つしかない、よって、Merger 理論により、著作権侵害は成立しないと判断した。

²¹ *Sega Enters. Ltd. v. Accolade, Inc.*, 977 F.2d 1510, 1525 (9th Cir. 1992) ("In our view, in light of the essentially utilitarian nature of computer programs, the Second Circuit's approach is an appropriate one." [コンピュータプログラムの本質的に実用的な性質に照らし、第2巡回区控訴審裁判所のアプローチは適切なものであると考える])。詳細については、SLN38号(1992.8.21)参照。

²² より具体的には、原決定20-21頁、Altai, at 709-710, (1)特定のプログラムが動作することが意図されているコンピュータの機械的な仕様、(2)他のプログラムとの互換性の必要性、(3)コンピュータ製造業者の標準設計、(4)業界の需要、(5)コンピュータ産業において広く受け入れられているプログラミングの実務といった要素が挙げられている。

²³ Altai 判決のフィルタリングテストは、Melville Nimmer 教授の息子である David Nimmer の提唱する基準に由来する。See, 3 Nimmer on Copyright §13.03[F][3], at 13-65, David Nimmer et al., A Structured Approach to Analyzing the Substantial Similarity of Computer Software in Copyright Infringement Cases, 20 ARIZ. ST. L.J. 625 (1988).

しかし、CAFC は、アイデアと表現の Merger を否定し、著作物性を肯定した。²⁴

- (a) 証拠によれば、Google が複製した 7000 ラインの選択と配列には無制限の選択があり得た。例えば、“java.lang.Math.max”という code の名称は、“Math.maximum”、“Arith.larger”などでもよく、そもそも予定されたものなどなく、そこに選択の余地があった。
- (b) 地裁は、Google が複製する時点で他の選択をする余地があったか否かを検討しているが、著作物性の問題は、複製時ではなく、API パッケージの創作時の選択可能性を検討しなければならない。この点、前述のとおり、Sun（当時）は、様々な呼称が可能な中で、“java.lang.Math.max”という名称で呼ぶことを選択した。

(B) 短いフレーズ (Short Phrase) 法理

地裁は、Android プラットフォームにおける名称の選択可能性を認めるものの、名前であるとか短いフレーズには著作権の保護は及ばないと判断した (Short Phrase 法理)。

CAFC は、問題となるのはフレーズの長さではなく、当該フレーズが創作的なものであるか否かであるかであるとして、前述のとおり、declaring code の記述や API パッケージには創作性があるとした。また、地裁が個々の line に細分化して短いフレーズとして著作物性を検討していること自体も適切ではなく、Oracle の主張は、細分化された個々の短いフレーズそれ自体ではなく、(いわば line の集合体としての) 編集著作物としての全体の保護であると指摘した。

(C) Scenes a Faire 法理

Google は、Scenes a Faire 法理 (あるテーマや状況について、ありふれた、標準的な表現であるという場合に著作物性を否定する法理) に基づき、declaring code の著作物性が否定される旨も主張していた。地裁では、ありふれた表現であることの立証がされていないと判断されていたが、CAFC は、さらに、第 9 巡回区控訴審裁判所の判例法においては、Scenes a Faire 法理は著作物性の判断要素ではなく、抗弁であると位置づけているので、仮にそのような法理を認めるとしても、API の著作物性を否定する理由にはならないとした。

また、Merger 理論の適用時と同様、この法理が問題となるのは Java API の複製時ではなく、創作時に創作者が置かれた状況が問題とされるべきであり、創作時に Sun のプログラマーによる Class、Method 及び Code の選択を指図する外部的要因の有無を問うべきだが、そのような事情を Google が立証していないと結論づけた。

② API パッケージの SSO について

(A) 連邦地裁の 102(b)の解釈—Lotus 判決への依拠²⁵

連邦地裁は、Java API パッケージの SSO が創作的かつ独自性のあるものであると判断したが、しかし、それが米国著作権法 102 条(b)の「システム又は動作方法であり...それゆえ著作権保護を受けられない」と判断した。その結論に到達するうえで、連邦地裁は、

²⁴ 但し、CAFC は、2つのパッケージについて Merger 理論の適用の可能性を示唆したが、Google がその点を主張していないと指摘した。

²⁵ Pamela Samuelson, “Legally Speaking, Are APIs Copyrightable?” (Vol.25, No.11, Communication of the ACM, 25), available at http://www.aipia.org/committees/committee_pages/Copyright-Law/Committee%20Documents/Oracle%20v.%20Google%20-%20AIPLA%20Primer%20on%20APIs/Samuelson%20Article.pdf

Lotus 判決²⁶（第 1 巡回区控訴審裁判所）に大きく依拠した。すなわち、同判決は、コマンドのヒエラルキー（階層）は、102 条(b)の「動作方法」（method of operation）であるため、著作権による保護を受けないと判断し、さらに、特定の言葉がある動作に不可欠なものである場合には、その言葉が「動作方法」の一部になるため、それ自体が著作権の保護の対象となり得ないとも判示した。

(B) Oracle の主張

Oracle は、本件と Lotus 判決(の前提事実)との違いを指摘して地裁の判断を批判した。

(a) Lotus 判決で被告 Borland は、Google とは異なり Code を複製していないので、事実関係がそもそも異なる。

(b) Lotus 判決は、問題となったコマンド（copy、print など）は創作性がないとしたが、本件では、declaring code、API パッケージの SSO に創作性があることに争いが無い。

(c) Lotus 判決では、問題となったコマンドが動作に不可欠のものであると判断されたのに対して、本件では、3 パッケージを除き、Java 言語でプログラムを書くために、Java API パッケージを複製する必要がなかった。

(C) CAFC の判断：Oracle の主張を認める。

(a) 「method of operation」に該当したら著作権保護を否定する Lotus 判決は、第 9 巡回区控訴審裁判所の判例法ではなく、また、アイデアそれ自体ではなく、アイデアを表現するものであればコンピュータプログラムの SSO に著作物性を認める第 9 巡回区の判例法にも抵触する。²⁷

(b) Lotus 判決のような解釈は、「プロセスや方法の表現」²⁸を著作権が保護することを認め、アイデアと表現とを区分する Altai 判決が採用した AFC テストを是認する第 9 巡回区の考え方に合致しない。

(c) そして、Lotus 判決の解釈と議会の意思との不整合を指摘する。

“If we were to accept the district court's suggestion that a computer program is uncopyrightable simply because it "carr[ies] out pre-assigned functions," no computer program is protectable. That result contradicts Congress's express intent to provide copyright protection to computer programs, ..., despite their utilitarian or functional purpose.”

（コンピュータプログラムは「事前に与えられた機能を実行」というだけの理由で著作物性がないという地裁の示唆を我々が万が一受け入れたら、コンピュータプログラムは法的に保護されなくなる。その結果は、その実用的・機能的な目的にも関わらずコンピュータプログラムに著作権の保護を与えた議会の明白な意思と矛盾するものである。）

(d) CAFC は、以上のような理由に基づき、著作者が根底にあるアイデアを表現する複数の方法を持つ限り、機能的な創作物に対しても著作権を付与するものであると解釈し、Google が依拠する Lotus 判決が解釈したように、102 条(b)は機能的なコンピュータプログラムの要素に対する著作権の保護を（カテゴリー的に）否定するものではなく、伝統的なアイデア／表現の二分論を定めたに過ぎないものとした。

²⁶ *Lotus Development Corp. v. Borland International, Inc.*, 49 F.3d 807 (1st Cir. 1995), *aff'd without opinion by equally divided court*, 516 U.S. 233, 116 S. Ct. 804, 133 L. Ed. 2d 610 (1996) 上告受理申立てが受理されたが、裁判官の意見が等しく分かれたため、第 1 巡回区控訴審裁判所の判断が是認された。

²⁷ *Johnson Controls, Inc. v. Phoenix Control Sys., Inc.*, 886 F.2d 1173, 1175-76 (9th Cir. 1989).

²⁸ *Altai*, 975 F.2d. at 839

(e) Java API の SSO により実現する機能を表現する方法は複数ある以上、SSO が機能を提供するというだけでは、102 条(b)によって著作権保護を否定されないとした。

(2) フェアユースについて

① コンピュータの互換性を達成する必要性と著作物性の関係性

連邦地裁は、Java API の SSO を「method of operation」と特徴付ける際に「コマンド構造の複製は互換性の達成のために必要である」と説明した。すなわち、pre-Android Java code を Android 上で実行するためには、Google は同一名称・同一分類・同一の機能仕様を有するコマンドシステムである Java package, class, method を提供する必要があったと判断し、このことが Java API の著作物を否定する結論に結びついた。

その結論を導くに際し、連邦地裁は第 9 巡回区裁判所の互換性に関連する裁判例である *Accolade* 事件と *Connectix* 判決に主に依拠した。²⁹

この点、CAFC は、上記 2 事件の判決は、被告が著作物の機能的な側面を理解するために中間的な複製物を作ったうえで最終的に新たな著作物を作っている点で、Google が Java API をリバースエンジニアリングして、著作物性のない機能的な部分へアクセスしたのではなく、declaring code と 37 の Java API パッケージの SSO を全て複製した本件は、事案を異にするものであるとした。

そして、両事件はあくまでフェアユースの成否が問題になったものであり、API の著作物性を問題とする本件とは異なることを強調し、Google が示唆した（連邦地裁は採用しているようにも思われる）著作物性に対する「互換性の抗弁」を正面から否定した。³⁰

CAFC は、互換性の必要に関する Google の利益は³¹、開発者がすでに有するベースを利用するために Java を梃子として開発プロセスを加速化することにあるとし、そのような利益は、著作物性ではなく、フェアユースの分析の中で検討されるべき問題であるとした。

② Google の Java API の複製はフェアユースか

この点について CAFC はフェアユース法理の一般論について詳述するものの、具体的な事実関係が十分に明らかになっていないとして、API の著作物性を認めただうえで、Google による複製行為がフェアユースに当たるか否かの審理に差し戻した。³²

²⁹ In reaching this conclusion, the court relied primarily on two Ninth Circuit decisions[この結論に達するため、裁判所は主に二つの第 9 巡回区裁判所の判断に依拠した。]: *Sega Enterprises v. Accolade, Inc.*, 977 F.2d 1510 (9th Cir. 1992), and *Sony Computer Entertainment, Inc. v. Connectix, Corp.*, 203 F.3d 596 (9th Cir. 2000).

³⁰ Google は、主張書面において、102 条(b)に基づき、互換性に関連する要素は著作権の対象とならないという前提を認めるものとして *Accolade* 事件と *Connectix* 事件を引用している。

³¹ 互換性の必要性が AFC テストの仕様を画するような外部要因として機能するのかを検討することがあることは認めるものの、CAFC は、そのような検討は「創作者」の互換性実現の必要性であって、本件のように「複製者」（後行者）の必要性ではないと指摘する。しかし、互換性を求めるのは通常後行者であり、そのような必要性の有無がフィルタリングの際の判断対象になるのであるから、CAFC の解釈は説得的とは思われない。

³² その他、rangeCheck file と逆コンパイルされたセキュリティファイルの著作権侵害を認めた連邦地裁の判断に対し Google が de minimis 法理に基づく異議申立てを行ったが、Google の主張が退けられている。

5 コンピュータプログラムの保護範囲を巡る裁判例の動向

1980年代から90年代にかけて米国においてコンピュータプログラムの著作権保護に関する議論を牽引したのは、Pamela Samuelson 教授と Dennis Karjala 教授である。³³ 本判決が驚きをもって受け止められた理由を理解するためには、これまでの裁判所の動向を大雑把にも押さえておく必要がある。³⁴

ここでは、主に Samuelson 教授の整理に従って、本判決でも言及されているコンピュータプログラムを巡る裁判例の動向を確認しておきたい。³⁵

(1) Franklin 事件から Paperback 事件まで

最初のソフトウェア著作権をめぐる事案は、Apple がそのオペレーティングシステムを複製した Franklin を訴えた事件 (3d Cir. 1983) である。本件で Franklin は、Apple のオペレーティングシステムプログラムは、Baker v. Selden 判決と 102 条(b)により著作権保護の対象から外されると反論した。

第3巡回区控訴審裁判所は、102 条(b)をアイデア／表現の二分論を規定したものととして、そして Baker 判決は Merger (アイデアを表現する方法が一つ又は非常に限られた数しかない場合には、その表現に著作物性を認めない) 理論を確立した裁判例として扱った。同裁判所は、その前提から、Apple のオペレーティングシステムとして同じ機能を実行するために複数のプログラムを書ける限り (その結果 Merger (混同) は否定される)、³⁶ プログラムはアイデアの「表現」であり、それゆえに著作物性が肯定されると判断し、Franklin の著作権侵害を認定した。

1986 年の Whelan 判決では、デンタルラボの SSO の著作物性を広く認め、異なるプログラミング言語・アルゴリズムを利用して SSO だけを複製した場合にも著作権侵害を肯定した。その際、Franklin 判決と同様、102 条(b)は、アイデア／表現の二分論を再確認するにすぎず、Baker 判決は Merger に関する裁判例であるとの立場をとった。

102 条(b)を制限的に解釈し、コンピュータプログラムの機能的側面を保護する (プログラムの複雑さとは関係がなく、一つのアイデアに基づきプログラムの全体を保護してしまう) ため、Whelan 判決に対する批判は強かった。³⁷

主な争点ではないので、本稿では省略する。

³³ Google の準備書面 (Brief) にも、両教授の論文が引用されている。

³⁴ 前掲注 6 に指摘したとおり、原決定はコンピュータプログラムの著作権保護の歴史を丁寧に遡って検証している。Samuelson, *supra* note 25 は、この点を高く評価する。

³⁵ Pamela Samuelson, Why Copyright Law Excludes Systems and Processes from the Scope of Its Protection, 85 *Tex. L. Rev.* 1921 (2007), at 1921

³⁶ カージャラ＝椋山 105 頁以下及びその原文にあたる、Dennis S. Karjala, Copyright, Computer Software, and The New Protectionism, 28 *Jurimetrics J.* 33 (1987) を参照。

機能的著作物に対する侵害判断について、「伝統的な著作権が技術的な『表現』について著作権侵害を認定するにはほとんど文字通りの類似性 (筆者注、デットコピーに近い類似性という意味であろう。) を要求しているにもかかわらず、小説や戯曲に適用されるべき表現についての広範な考え方を技術に対して誤って適用している。」とする。

³⁷ カージャラ＝椋山 118 頁以下を参照。

その批判の要は、①102 条(b)の下地となった Baker 判決の理解が間違っていることと、②コンピュータプログラムという機能的著作物と他の文芸的著作物との差異を全く無視していることにある。「アイデアを表現から区別するための基準を決定するにあたり、(筆者注、Whelan 判決の) 裁判所は、Baker v. Selden 判決を読み違え、同判決が、目的を達成するためのさまざまな方法があることから、それぞれの方法が

しかし、Whelan 判決に対する強い批判にも関わらず、Paperback 事件³⁸において、裁判所は Whelan 判決の結論に従った。

被告の Paperback は、Lotus1-2-3 のメニューコマンドシステムの構造を複製したとして著作権侵害に問われた。Paperback は、一定の場合 SSO は著作権法の保護対象となることを認めたが、メニューコマンドは「system」（システム）であり、102 条(b)によって著作権法による保護の対象から外れる旨を主張した。

裁判所は、Whelan 判決と同様、102 条(b)と Baker 判決は、抽象的アイデアと保護される表現とを区別する規定であるとし、他の手段の存在を SSO が著作権法によって保護されるものかどうかを判断するための鍵となる要素であると判断した。すなわち、「(Lotus) の一つのメニュー構造の特定の表現は、電子的なスプレッドシートに不可欠なものでも、スプレッドシートに関する一つのメニュー構造の少し抽象的なアイデアと混同 (merge) するものでもない。」から、そのようなアイデアは、「文字通り無制限ではないにせよ、数多くの方法で表現することができる」と指摘し、Lotus のメニューコマンドシステムの SSO は著作権法上保護されると判断した。

(2) Altai 事件から Lotus 事件まで

この流れを大きく変えたのが Altai 判決である。

Altai は、ZEKE というスケジュールソフトを様々なオペレーティングシステムにおいて動作するように修正を施すにあたり、Computer Associates International Inc. (以下「CA」という。) の元従業員を雇用したところ、同従業員は、CA が販売していたスケジューラーの互換性を達成する ADAPTER というモジュールのソースコードの 30%を無断で複製し利用していたため (ソフト名は OSCAR)、CA が Altai に対して著作権侵害訴訟を提起したものである。

Altai は、ソースコード (文字的部分) の複製については著作権侵害を認めてコードを書き直し、新しいバージョンのスケジュールソフトを制作・販売した。しかし CA は、この新しいバージョンのソフトも実質的に CA のスケジュールソフトのパラメータリスト (モジュール間でやりとりされる情報の形態) と Macros (命令) を実質的に複製するものであり、著作権侵害であると主張した。パラメータリスト及び Macros はモジュールの構造を決定づけるもので SSO の一つであり、Whelan 判決と同様、SSO の著作物性が問われた。

Altai 判決は、プログラムの非文字的な (nonliteral) 要素が著作権法上保護されることを認めながら、Whelan 判決が「ある実用的な著作物の目的又は機能はその作品のアイデアであり、その目的又は機能にとって必要ではない部分はそのアイデアの表現の部分というこ

保護された表現であるということを暗示しているとの前提を指示するものとして理解している。Baker 判決の裁判所は、帳簿を付ける他の方法があるかないかについては注意を払わず、たんにその表に著作権を認めることが原告が開発した特別のシステムを独占することになるかどうかについて注意を払った。さらに重要なことは、Whelan 事件の裁判所は、(略) 製作物の質的に重要な部分を利用することは仮にその部分が量的に僅少であってもその著作権を侵害する、と判示した文芸作品に関するケースを参照した。このアプローチは、技術的製作物とその他の製作物との重要な違いを無視している。革新的な技術開発は、それが極めて重要であったとしても、著作権によっては保護されていない。」(カージャラ=相山 120 頁、下線は筆者)

³⁸ Lotus Development Corp. v. Paperback Software International, 740 F.Supp.37 (D. Mass.1990).

とになろう。望ましい目的を達成する手段が様々ある場合、そこから選ばれた一つの特定の手段は、その目的に必然的なものではない。よって、それは表現であり、アイデアではない」とする点を誤りであるという。すなわち、各モジュールが一つのプログラムであり、「アイデア」を持っているので、一つの全体的な目的とそのプログラムのアイデアを等置する Whelan 判決の一般的公式は、説明としても不十分なものであると指摘する。

そして Altai 判決は、前出の AFC テスト³⁹に従って著作権侵害の有無を判断すべきだとし、特に、AFC のフィルタリングテストにおいて、①効率性によって決定づけられるプログラム設計の要素、②プログラムがともに機能するハードウェア・ソフトウェアといった外部要因によって制約を受けるプログラム設計、③プログラムのパブリックドメインの要素（一般化したプログラミング技術など）を除くこととする点が、Whelan の一つのアイデアを抽出したほか全てを「表現」とする基準とは全く異なる。

Altai 判決は、AFC テストのフィルタリングの結果を比較し、SSO であるパラメータリスト及び Macros について、パブリックドメイン又はプログラムの機能上の制約（それが作動する IBM オペレーティングシステムのインターフェースによる制約）に基づく類似点であるとして、侵害を認めなかった地裁の判断を支持した。

その後、コンピュータプログラムの著作権に基づくユーザーインターフェースの保護の可否が争われた Lotus 判決（1995）は、Altai 判決の AFC テストを利用せず、102 条(b)を根拠に端的に著作権侵害を否定した。

被告の Borland は、原告 Lotus の Lotus1-2-3 のメニューコマンド構造とは異なるスプレッドシート（ユーザーインターフェース）を自ら開発していたが、Lotus1-2-3 の既存利用者の移行を誘引すべく、Lotus1-2-3 で構築された Macro（命令）をそのまま Borland のスプレッドシートで利用できるようにするエミュレーション機能を付加する目的で Lotus1-2-3 のメニューコマンド構造を複製したため、Lotus が著作権侵害訴訟を提起したものである。

連邦地裁は、コマンド名称を変えることで、非常に多くのメニュー、サブメニューについて非常に多くの Lotus とは異なる選択肢があること、さらにコマンドの配列や階層化にも同様に選択肢があることから、Lotus のコマンド構造に著作物性を認めた。

これに対して第 1 巡回区控訴審裁判所は、102 条(b)及び Baker 判決に依拠し、Lotus 1-2-3 のコマンド構造は「method of operation」（動作方法）であるとして著作物性を否定し、地裁の判断を覆した。

第 1 巡回区控訴審裁判所の判断（Lotus 判決）は、102 条(b)の明文上著作権の保護対象から外す「method of operation」であるとして、Altai 判決のような AFC テストを経由せずに著作権法による保護を否定した点に特徴があり、Altai 判決のような複雑な評価を経ることなく⁴⁰、端的に、コンピュータプログラムの著作権保護の範囲を絞るものといえよう。⁴¹⁴²

³⁹ AFC テストは、文芸の著作物で確立した侵害判断の基準(抽象化テスト)をコンピュータプログラム事件に応用したものである。Karjala, *supra* note 3, at 79, *Nichols v. Universal Pictures Co.*, 45 F 2d. 119(2d Cir. 1930) (L. Hand, J.), cert. denied, 282 U.S. 902(1931) 参照。Altai 判決は、Whelan 判決との決定的な違いは、抽象化テストが、著作物が無数のアイデアと表現が混合されて構成されている可能性を暗黙の前提としている点であるという。また、Altai 判決は、コンピュータプログラムという、その機能的性質ゆえに特別な著作物の特質を Whelan 判決よりも意識しており、「コンピュータプログラムの本質的に実用的な性質が表現からアイデアを分離する作業を更に複雑なものとする」(Altai, 982 F 2d. at 704) と述べている。

⁴⁰⁴⁰ Pamela Samuelson, A Fresh Look At Tests for Nonliteral Copyright Infringement, 107 Nw. U. L. Rev. 1821,

6 本件の評価

(1) Baker 判決と著作権法 102 条(b)の解釈

5 で検討したとおり、コンピュータプログラムの著作権法の下での保護範囲について Whelan 判決と Lotus 判決を左右の両極としてその中間点に Altai 判決が位置付けることが可能であり、本判決も同様の理解を示している。⁴³

本判決は、連邦地裁が Lotus 判決に依拠した点を批判した Oracle の主張を是認した。

主な批判は、①Lotus 判決（第 1 巡回区控訴審裁判所）は本件事件が準拠すべき第 9 巡回区控訴審裁判所の判例法ではないこと、②Lotus 判決の 102 条(b)の解釈によれば、同条(a)でコンピュータプログラムの創作性・独創性を認めながら、コンピュータプログラムが（性質上当然に）「システム」であり、「method of operation（動作方法）」であることから、同条(b)によってその保護が否定されてしまうが、かかる解釈は、コンピュータプログラムを著作権により保護することを認めた議会の意思に反する点にある。

コンピュータプログラムの保護範囲を巡る裁判例の動向は前述のように 80 年代から 90 年代にかけて大きく揺れた。その分岐点は、コンピュータプログラムという特殊な著作物を著作権法によって保護することの意味と 102 条(b)を定めた議会の意思をどのように解釈するか、という点にある。

CAFC は、Lotus 判決ではなく Altai 判決を基礎に、102 条は「システム」「動作方法」をカテゴリカルに著作物から除外したのではなく、Baker 判決で確立した Merger 理論を定めたものとした。⁴⁴ その上で、Java API の SSO により実現する機能を表現する方法は

1839, 1840 (2013) は、これまでの裁判例上の nonliteral な著作権侵害の判断基準を考察する中で、Altai 判決を評価しつつ、102 条(b)が明確に「手続、プロセス、システム、操作方法」を著作権の保護対象から除外しているにもかかわらず AFC テストがそれを求めなかった点を指摘し、102 条(b)の除外を AFC テストのろ過プロセスに位置づけるべきであったが、そうしなかったと述べる。

⁴¹ Lotus はこれを不服として上告受理申立てを行った。その申立理由の一つは、102 条(b)の著作権からの除外対象とされる「システム」や「method of operation(動作方法)」を文字通りに解釈し、Lotus 判決のように著作物性を否定することが、コンピュータプログラムに著作権の保護を及ぼすものとした議会の意思を否定することとなるため、裁判所はそのような解釈を採るべきではない、という点にあった。Lotus は、102 条(b)はそのような特殊な除外事由を設けたものではなく、アイデア／表現の二分論を単に条文化したものに過ぎないと主張した。なお、この上告受理申立ては受理されたものの、裁判官の意見が分かれた結果、Lotus 判決が是認された点は注 26 で述べた。

⁴² Mark Lemley et al., *Software and Internet Law* (Third edition), at 73 は、Lotus 判決によれば、API の著作物性は完全に否定されることになるが、ほとんどの裁判所はこうした絶対的なアプローチを採っていないと評価する。

⁴³ 本判決は、SSO の選択肢があったことを極めて重視し、フィルタリングにおける除外事由を詳細に検討した様子がないため、Whelan 判決の復活ともいえるように思われる面もあるが、CAFC は、自らの判決を Whelan 判決ではなく Altai 判決の系譜に位置づけている。判決 22 頁以下。

⁴⁴ Oracle は、102 条(b)の列举が特別な除外事由を定めたとする Lotus 判決の解釈が不当であると主張する際、元著作権局長の Ralph Oman の見解や Altai 判決ほかの裁判例を援用するだけでなく、現在 Google の Copyright Counsel を務める William Party 氏が（皮肉なことに）過去に（今回 Google が大きく依拠した）Lotus 判決を、全てのコンピュータプログラムを保護されない「method of operation」とするものだと批判していた（まさに本件における Oracle の主張そのものである）ことに言及している。See, William F. Patry, *Copyright and Computer Programs: It's All in the Definitions*, 14 *Cardozo Arts& Ent. L.J.*1, 5-8, 13, 59-63(1996),

複数ある以上、仮に SSO が機能を提供するというだけでは、102 条(b)によって著作権保護を否定されないとしたのは前述のとおりである。しかし、他の選択肢があるというだけで SSO の保護を認めるのは行き過ぎではなかろうか。ある機能を実現するために SSO の選択肢など複数あるのが通常であり、それだけでは保護される表現とアイデアとを区別できるものではないし、ほとんど全てが表現と評価されてしまう結果（その結果は Whelan のような表現部分を非常に広くとる見解と軌を一にするように思われる。）を招来する。⁴⁵

かかる CAFC の判断は、Altai 判決の AFC テストのフィルタリングのテストの中身が不明瞭であるという弱点⁴⁶を浮き彫りにした。実際に AFC テストの濾過テストを通過して残る SSO の創作性とはどのようなものか、想起しにくいように思われる。

今後は、再審理を求められたフェアユースの成否とともに、Lotus 判決の際に最高裁で決着のつかなかった 102 条(b)に関する CAFC の解釈が是認されるか否かも議論的となると思われる。

この点、Samuelson 教授は、CAFC の採った 102 条の狭い（列挙されている語に特別な意義を見出さず、アイデア／表現二分論を採用したに過ぎないとする）解釈が、Melville B. Nimmer による Baker 判決⁴⁷の誤った解釈⁴⁸に起因することを、102 条(b)の legislative history（制定経緯）に遡って丁寧に論証し、改正法前から存在した、コンピュータプログラムを著作権法の下で保護することへの懐疑を背景に、102 条(b)があえて「システム」、「動作方法」等を除外事由と定めたものであると主張している⁴⁹。

CAFC は、102 条(b)の列挙事由に特別な意味を見出さない立場を採用したが、その過程で立法経緯が詳細に検討された様子は伺えない。Google は当然のことながら、Baker 判決の正当な評価に依拠して 102 条(b)を解釈すべきであることを主張しているが、CAFC は 102 条(b)の立法経緯を詳細に検討することなくその主張を退けている。

なお、かかる Patry の見解については、Karjala, *supra* note 3, at 72 footnote 54 参照。

⁴⁵ Karjala, *supra* note 3, at 89 footnote 124 参照。

⁴⁶ Altai 判決も「率直に言って、非文字的なプログラム構造に対する著作権の保護の正確な境界は完全にははっきりしていない」と明言する。また、Karjala, *supra* note 3, at 80, 81 は、Altai 判決のフィルタリングテストの結果、ほとんどの非文字的要素が創作的表現として重要な要素として残る余地がないのではないかと指摘し、AFC テストの結果は、コードのデッドコピーまたはそれに近い実質的な複製のみを保護する結果となるはずだが、Altai 判決がその点を明確しなかったため、その後の Altai 判決に従おうとする裁判所がまごつく結果を生んでいると述べる。今回の CAFC による判断は Karjala 教授の懸念が顕在化した結果というべきであろう。

⁴⁷ Samuelson, *supra* 35, at 1928-1936, 連邦最高裁は、「科学上の真実又は技術上の方法は世界共通の財産であり、いかなる著作者も自分独自の方法でそれを表現し、説明し、利用する権利を有する。」と判断し、端的に、「未記入の会計帳簿は、著作物の対象足り得ない」として Selden の主張を退けた。

⁴⁸ Lloyd L. Weinreb, Copyright For Functional Expression, 111 Harv. L. Rev. 1149, 1173 (1998) は、この判断は「システム又は実用性のある技芸 (art) は、それがどのような方法で表現されていようとも、著作物の対象足り得ない、という簡潔かつわかりやすいものである」と評価する（下線は筆者）。これに対して Melville Nimmer は、Baker 判決から著作物性を判断するための基準を見出すことに反対し、Baker 判決によって、いかなる著作物の著作物性も否定されてはならないとした。Nimmer は、あるシステムの表現がシステムそれ自体と区別できる以上、著作権の対象であると主張した。しかし、かかる Nimmer の解釈は、Baker 判決が、システムの表現方法が他にあるか否かさえ検討せずに著作権による保護を否定していることと整合しない。

⁴⁹ Samuelson, *supra* 35, at 1953-1961,

よって、仮に本件が最高裁で審理されることになれば、この点が詳細に争われる可能性も相当程度あると思われる。⁵⁰⁵¹

(2) コンピュータプログラムの法的保護の在り方

CAFC は、判決の中で、独立して「Google' Policy-Based Argument」という項目を立て、Google がその立場をサポートするものとして大きく依拠した Samuelson 教授を批判的に採り上げている。ソフトウェアの著作権による保護は制限し、むしろ特許による保護を求めべきであるという教授の見解⁵²に疑問を投げかけ、むしろ著作権法による保護はもともとソフトウェアに適したものであり⁵³、かつ、著作権の保護と特許法の保護が重複することは連邦最高裁も認めているところであるとし⁵⁴、議会又は最高裁が立場を変えない以上、ソフトウェアプログラムの保護は特許法に限られるべきであるという主張は支持できないと結論づけている。⁵⁵

Samuelson 教授は、102 条(b)の Nimmer による解釈が誤っていることを指摘し、Baker 判決の正当な理解と 102 条(b)を定めた議会の意思からすれば、コンピュータプログラムの著作権による保護は謙抑的であるべきとすべきもので、CAFC の指摘は、同教授が立法論としてそのような主張をしているかのごとき理解をするもので、同教授の主張を誤解するものと言わざるをえない。

⁵⁰ なお、102 条(b)をアイデア／表現二分論の確認にすぎないという Nimmer の解釈は、Lotus 事件の原審が採用する立場であり、その解釈には次のような批判が加えられている。そのような解釈に従えば「Lotus が行ったように、Selden がコンピュータプログラムによって帳簿システムを実行していたら、原判決の分析の下では、(筆者注、Baker 判決の判示内容とは全く異なり) その『規則に従った線と会計項目』及びシステムの体系化の方法を擁護する権利を与えられることになろう。」が、それは、Baker 判決を無視するものと批判する。See, Dennis S. Karjala and Peter S. Menell, *Applying Fundamental Copyright Principles to Lotus Development Corp. v. Borland International, Inc.*, 10 Berkeley Tech. L. J 177, 189 (1995)

⁵¹ Samuelson 教授の私信によれば、今秋にも上告受理申立ての可能性があるのである。ちなみに、Copyright minimalist (著作権の強化に懐疑的な論者) と評価されている Justice Stephan Breyer は、tenure を取得した論文において、著作権によるコンピュータプログラム保護に懐疑的な姿勢を見せている。Stephen Breyer, *The Uneasy Case for Copyright: A Study of Copyright in Books, Photocopies, and Computer Programs*, 84 Harv. L. Rev. 281(1970).

⁵² See also, Dennis S. Karjala, *Distinguishing Patent and Copyright Subject Matter*, 35 Conn. L. Rev. 439(2002).

⁵³ CAFC は、ソフトウェアの特許による保護に疑問を呈する見解として、次の 2 つの記事を紹介している。See *Technology Quarterly, Stalking Trolls*, ECONOMIST, Mar. 8, 2014, <http://www.economist.com/news/technology-quarterly/21598321-intellectual-property-after-being-blamed-stymying-innovation-america-vague>; Timothy B. Lee, *Will the Supreme Court save us from software patents?*, WASH. POST, Feb. 26, 2014, 1:13 PM, <http://www.washingtonpost.com/blogs/the-switch/wp/2014/02/26/will-the-supreme-court-save-us-from-software-patents/>.

⁵⁴ *Mazer v. Stein*, 347 U.S. 201, 217, 74 S. Ct. 460, 98 L. Ed. 630, 1954 Dec. Comm'r Pat. 308 (1954).

⁵⁵ Oracle は、Google の依拠する Samuelson 教授がコンピュータプログラムを著作権法で保護するのは間違いであると主張した論文に言及し、Google の主張が立法論である(議会の意図とは別のことを言っている)かのような主張をしている。CAFC の判断を見る限り、Oracle の戦略は成功したというべきであろう。しかし、同教授は言うまでもなく現行法を無視しているのではなく、Baker 判決と 102 条(b)の正確な理解に基づいて(Nimmer の解釈の誤りを正すべく)議論していることは、Samuelson, *supra* 35 の論文を読めば明らかであり(Google はこの論文を引用しているが、Oracle がその主張に用いたのは、Pamela Samuelson, *CONTU Revisited: The Case Against Copyright Protection for Computer Programs in Machine-Readable Form*, 1984 Duke L.J.663(1984)である。)、CAFC の批判は当たらないように思われる。

コンピュータプログラムの法的保護の在り方は 80 年代～90 年代に大きく議論されたテーマであり、著作権法と特許法という異なる法律を用意する知的財産法体系の下でそれをどのように位置づけるのかは重要な視点である。

その技術的性格に付随する社会の恩恵ということを考えて場合、後行者による改良が徐々に加えられていくことが他の著作物とは大きく異なる特徴⁵⁶であり、かかる特徴うえに、他の伝統的著作物に比べ、コンピュータプログラムの著作権による保護はより制限的・謙抑的に解釈されるべきである。具体的には、文字的要素（ソースコードとオブジェクトコード）のデッドコピー又はそれに類する複製に限るべきであろう。CAFC は、機能性の保護が特許ではなく著作権で保護されることの意味を十分に咀嚼していないようである。かかる解釈は、特許法において保護されない機能⁵⁷が著作権法によって最小限度の創作性を充足するだけで特許より長い法的保護を享受することを認めるものだが、コンピュータプログラムの機能（特に nonliteral な部分）がそこまで強固な法的保護を受けることを正当化する事情を提示していない。

前述のとおり、Whelan 判決は、コンピュータプログラムのアイデアに当たるものにまで、特許法の求める厳格な要件を充足しないにも関わらず特許以上の長期間にわたり独占権を付与する（法的保護を与える）結果を生じさせるものであったが、Altai 判決及び Lotus 判決によって、コンピュータプログラムの著作権による法的保護の領域が極めて限定的なものになったと理解されたため、ソフトウェアの SSO について保護を求める者は特許権を取得することが一般的なものとなっていったと評価されていた。すなわち、著作権による保護の強化傾向は Whelan 判決以後一時強まったものの、90 年代半ばまでには適正なレベルまでに弱まり、その取扱いが安定した。^{58,59}

本決定は、そのような理解を根底から覆す判断である。CAFC の判断が米国のソフトウェア産業にとってどのような影響を及ぼすのか、この事件の進展とともに、米国内での議論の推移を見守る必要がある。

また、各論的には、後発者の互換性を実現するための SSO の複製がフェアユースを構成

⁵⁶ Karjala, *supra* note 3, at 61, 「文芸の著作物に関しては、最終章だけが異なっている 100 冊の「戦争と平和」よりも、100 の異なった戦争小説を持つことを我々は望む。その結果、小説に対する幅広い長期間の保護が、後行者を不当に妨害することなく、また、社会から望ましい作品を奪うことなく創作者を評価するという目的を達成する。しかし、技術は、消費者にとってより望ましい又は有用なものとするために後行発明者がわずかな改変を加えることにより徐々に改善されていくものである。」。かかる根本的な相違が著作権法と特許法のポリシーにあることを指摘し、このような基本的な視座から、コンピュータプログラムが著作権法の下で保護されることの意味を考える必要を説く。

⁵⁷ *Alice Corp. Pty v. CLS Bank Int'l*, 573 U.S. ___ (2014) によって、ソフトウェア特許の法的保護の基準は従前よりも厳格なものとなると推測される。同判決について SLN137 号(2014.7)参照。

USPTO のメモランダムも参照。 http://www.uspto.gov/patents/announce/alice_pec_25jun2014.pdf CAFC のような解釈は、そこからこぼれた特許に値しないコンピュータプログラムの機能を著作権法によって保護する道を開くものとなる。

⁵⁸ Pamela Samuelson, *The Uneasy Case for Software Copyrights Revisited*, 79 *Geo. Wash. L. Rev.* 1746, 1775-1782, 1980 年代～90 年代のコンピュータプログラム著作権をめぐる動向を総括し、コンピュータプログラムに対する著作権による保護が弱まる形で安定したことを肯定的に評価する。特に 90 年代の裁判例の動向がソフトウェア産業にマイナスの影響を与えたとは評価できないとする。

⁵⁹ 原決定も同様の理解をしている。原決定 32 頁。

するのか、つまり、互換性・技術の標準化という事情に基づきフェアユースが認められるのか、という点の差戻し後の原審の判断も注目される。⁶⁰

なお、本稿の作成にあたり梶山敬士弁護士より様々な助言及び示唆を頂いた。最後に記し、お礼を申し上げたい。

以上

⁶⁰いわゆる変容的利用（transformative use）法理では、直ちにフェアユースとの判断が導き出せないように思われる。互換性を実現するためのリバースエンジニアリングについてフェアユースとするのが第9巡回区控訴審裁判所の判例ではあるが、Google の行為は、リバースエンジニアリングをして新たな物を創作したものではないため（当然 Oracle は Sega 事件等との相違を強調する。）、その点がどのような判断がなされるのかも注目される。互換性実現に対する需要を考えれば、フェアユースとの判断が出る可能性も相当程度あるように思われる。米国が 107 条の柔軟な解釈によって産業を発展させてきた面は否定できず、このような米国特有の社会背景も加味する必要がある。See, Pamela Samuelson, *Unbundling Fair Uses*, 77 *Fordham L. Rev.* 2537, 2605 (2007), Fred von Lohmann, *Fair Use as Innovation Policy*, 23 *Berkeley Tech. L.J.* 829 (2008).