

SLN No.53 1994.2.4

フィルトレーション・テストを使って差戻

—ゲイツ対バンドー事件で第10巡回区—

1. はじめに	1
2. 事実関係	1
3. 著作権侵害について	2
4. トレード・シークレットについて	4
5. 若干のコメント	4

1. はじめに

産業機械用ベルトの選択に関するプログラムにつき、1993年10月19日、米国第10巡回区控訴裁判所は、著作権侵害を認めたコロラド地方裁判所の判決を破棄差戻した。この一審判決については、SLN40号で紹介したが、CA対アルタイ判決（1992年6月22日、SLN38号参照）の出されれたわずか2日後の1992年6月24日に出されており、その影響を受けなかったものとみられる。本控訴審判決は、CA対アルタイ判決と同様、abstraction - filtration - comparison の3段階テストを用いて、著作権侵害を否定した。

なお、トレードシークレット侵害については、原審を維持した。

2. 事実関係

詳しくは、前記SLN40号を参照されたい。控訴人、被控訴人会社のいずれも産業用機械のゴムベルトの製造販売会社であるが、客先で適当なベルトを選択するためにセールス・エンジニアがプログラムを利用する。バンドーの社長以下多数の者が、ゲイツから転職した者であった。

3. 著作権侵害について

A.

(1) 本判決は、ファイスト判決等を引用しつつ、著作権侵害の一般論を述べ、著作物に含まれる「保護される要素」を奪われたときのみ侵害になるという考え方を確認する。そして、どのような要素が保護されるのかを決定するテストとして、abstraction - filtration - comparison（抽象化-濾過-比較、以下本稿でa f cテストと略す。）を採用する。本判決に従って、a f cテストを記述しておく。

第1段階（抽象化）……一般性のレベルに応じてプログラムを分ける。

第2段階（濾過）……上記各レベルを検討し、保護されない要素を取り除く。アイデア、プロセス、事実、公有情報、scenes a faireに属するもの、その他の要素である。

第3段階（比較）……然る後、残った保護される要素を比較して、被告プログラムが原告プログラムの実質的要素を奪うものか否か決定する。

(2) この段階に従って、判決は検討を進める。

i) 抽象化

ラーニド・ハンド判示のニコルズ判決やニマーを引用した後、オギルビー論文に従った分析を展開しており、この点本判決の（多少の）新味がある。すなわち、プログラムを、以下のように分ける。

- 〈1〉 主たる目的
- 〈2〉 アーキテクチャー又はストラクチャー
- 〈3〉 モジュール
- 〈4〉 アルゴリズムとデータ構造
- 〈5〉 ソース・コード
- 〈6〉 オブジェクト・コード

本判決は、こうした分析はそれ自体が目的ではなく、次の濾過手続きを容易にするための道具だとしている。

ii) 濾過

以下のように、分説している。

(a) アイデア-表現二分法

主たる目的は常に保護されない。各モジュールの基本機能ないし目的もほとんど常に保護されない。逆に、ソース・コードやオブジェクト・コードは、マージャーや scenes a faire が登場しない限り、常に保護される。中間のSSO等については、一般論では論じ難い。

(b) プロセス-表現二分法

実用目的の作品においては、プロセス排除の方がより重要である。前記抽象化レベルのうち、〈1〉主たる目的の部分を除く各レベルでプロセスは見つけられるが、システム・アーキテクチャー又はアルゴリズムの中に見つけられる。

(c) 事実

事実は著作者にオリジンがないから保護されない。事実は、抽象化の各レベルで見つかるが、データ構造やソース・コード、オブジェクト・コードの中によく見つけられる。

(d) 公有

公有物も著作者のオリジナルではないから保護されない。このようなものは各抽象化レベルで見つけられる。

(e) マージャー

アイデア、プロセス、発見と分けられない表現は保護されない。

(f) scenes a faire

特定のトピックにとって標準的、陳腐、共通する表現又は共通のテーマ若しくはセッティングから必然的に出てくる表現は保護されない。

外部的要素に規定されるものも保護されない。それには、ハードウェア標準と機械仕様、ソフトウェア標準と互換の要請、コンピュータ・メーカーの設計標準、対象産業の慣行と要求、コンピュータ産業のプログラミング慣行が含まれ得る。

iii) 比較

これは質的評価であり、ケース・バイ・ケースにならざるを得ない。

B. 憲法論と基礎理論

連邦憲法 I、§ 8、cl. 8 を引きつつ、著作権法の政策目的は、著者に保護すなわち適正な報償を与えることによりインセンティブを与えることと、学習、進歩、発展を求める伝播との間にバランスをとることにある、とする。

過去の主要判例について、次のようなコメントを付している。

ウェーラン判決は広く批判されているが、基本的には伝統的理論にのっており、ストラクチャーを保護した結論も健全である。

ロータス対ペーパーバック判決には、a f c テストの萌芽が見られる。

CA 対アルタイ判決は a f c テストをとっており、本判決もこれを採用し、若干展開した。

フェイスト判決は電話帳の事件だが、基礎理論を述べ、額の汗理論を否定した。

オートスキル判決は予備的差止命令の事件だったので、当審は a f c テストをとるのを控えたが、本件ではとることとした。

C. 本件への a f c テストのあてはめ

原審は、メニュー、定数、ソート基準、コントロール・フロー、データ・フロー、工学計算モジュール、設計モジュール、共通のエラー、基本タスク、インストール・ファイルが保護される要素としたが、濾過手続きを誤っている。

(1) 定数

事実であるから保護されない。その発見に何千時間かかっても、額の汗に過ぎない。濾過しなければならなかった。

(2) メニューとソート基準

この概念がはっきり規定されていないので、原審は検討せよ。

(3) コントロール・フローとデータ・フロー

ここでも概念が定かでないので、再考せよ。特に、アイデア表現二分法とプロセス表現二分法に照らして。

(4) 工学計算モジュールと設計モジュール

あるアルゴリズムがプロセスにあたるなら保護されないが、表現にあれば、マージャー論を検討する。これらに照らして再考せよ。

(5) 共通のエラー

共通のエラーは、事実問題として、コピー行為の強力な証拠となるが、どのような要素が著作権法上保護されるかを決定するのを助けるものではない。エラーそれ自体は保護されるものではないが、エラーを含む表現は他の要件をみたまし保護され得る。その分析をせよ。

(6) 基本タスク

この語は、通常、抽象化の最も高いレベル、すなわちアイデアや目的をいう。原審はもっと具体的な内容を考えているようだが、何を意味しているかわからないので分析せよ。

(7) インストール・ファイル

これはフロッピーからハードディスクにインストールするための別個のユーティリティであり、OSに属するから、原告プログラムの一部ではない。従って、ゲイツがインストール・ファイルに著作権を持っているか明らかでない。この点を判断し、さらに本判決で述べたテストを行え。

4. トレード・シークレットについて

原審は「定数」のトレード・シークレット侵害を認めた。当審は、本件において州法が連邦法でプリエンプトされないと考えるし、ゲイツは財産性、秘密管理性の十分な立証をしたと考えるので、原審判断を維持する。

5. 若干のコメント

本判決により、フェイストの流れの下にCA対アルタイのa f cテストを採用した巡回区が一つ増えた。従って、このテストが全国的に定着しつつある、といえよう。ただ、このテストも一義的な基準を提供するものではないから、その運用の仕方をさらに研究していく必要がある。

(了)