平成２４年７月２６日判決言渡

平成２３年（行ケ）第１０３０２号　審決取消請求事件

口頭弁論終結日　平成２４年５月３１日

判　　決

原　　告	ｲﾝｶ ｲﾝﾀｰﾈｯﾄ ｶﾝﾊﾟﾆｰ ﾘﾐﾃｯﾄﾞ
訴訟代理人弁理士	林　　佳　　輔
訴訟代理人弁護士	高　　橋　　雄　一　郎
同	大　　堀　　健　太　郎
同	北　　島　　志　　保
訴訟代理人弁理士	望　　月　　尚　　子
同	坂　　場　　紀　　雄
同	中　　山　　秀　　明
同	本　　田　　昭　　雄
同	納　　戸　　慶　一　郎
訴訟復代理人弁理士	荒　　尾　　達　　也
被　　告	特　　許　　庁　　長　　官
指　定　代　理　人	田　　中　　秀　　人
同	樋　　口　　信　　宏
同	長　　島　　孝　　志
同	田　　村　　正　　明

主　　文

１　原告の請求を棄却する。

２　訴訟費用は原告の負担とする。

３　この判決に対する上告及び上告受理の申立てのための付加期間を３０日と定める。

事　実　及　び　理　由

第１　請求

特許庁が不服２００８－１７３７３号事件について平成２３年５月１１日にした審決を取り消す。

第２　争いのない事実

１　特許庁における手続の概要

⑴ 原告は，発明の名称を「オンライン上での有害情報遮断システム及び方法，並びにそのためのコンピュータで読出し可能な記録媒体」とする発明について，平成１２年１１月２８日（パリ条約による優先権主張外国庁受理１９９９年１２月３１日，韓国）を国際出願日とする国際出願をした（以下「本願」という。）。

⑵ 本願については，平成１７年５月１２日付けで拒絶理由通知がされ，これを受けて，原告は，同年８月１７日付けで，特許請求の範囲を変更する旨の手続補正（甲３。以下，これによる補正後の明細書及び図面を「本件明細書」という。）をしたが，平成２０年３月３１日付けで拒絶査定がされた。

⑶ これに対し，原告は，同年７月７日，拒絶査定不服審判請求（不服２００８－１７３７３号事件）をしたが，特許庁は，平成２３年５月１１日付けで「本件審判の請求は，成り立たない。」旨の審決（以下「審決」という。）をし，その謄本は，同月２４日，原告に送達された。

２　特許請求の範囲の記載

本件明細書に記載の特許請求の範囲の請求項１に記載の発明（以下「本願発明」という。）は，次のとおりである。

「実行対象のファイルにおける有害情報をリアルタイムで遮断する方法において，

（ａ）ウェブサーバーとクライアントシステムが相互連結されたコンピューターネットワークにおいて，前記ウェブサーバーがコンピューターネットワークを通じて前記クライアントシステムからの接続要請を受信するステップ；

（ｂ）前記ウェブサーバーが前記クライアントシステムに有害情報遮断コードモジュールを伝送するステップ：及び

（ｃ）前記有害情報遮断コードモジュールの伝送が完了すると，前記クライアントシステムが前記有害情報遮断コードモジュールを自動的に駆動して，コンピュータウィルスを含む有害情報をリアルタイムで遮断するステップを含み，

前記ステップ（ｃ）が，

（ｃ１）ファイルＩ／Ｏルーチンを奪うことにより前記クライアントシステム上におけるファイル入出力（Ｉ／Ｏ）を監視するステップ；

（ｃ２）前記ステップ（ｃ１）で監視されたファイル入出力（Ｉ／Ｏ）と関連している実行対象のファイルの有害有無を判断するステップ；及び

（ｃ３）前記ステップ（ｃ２）で有害と判断されたファイルの治療が可能な場合は適切な処理を行い，前記ステップ（ｃ２）で有害と判断されたファイルの治療が不可能な場合は該当ファイルの実行を中止させるステップを含むことを特徴とする方法。」

３　審決の内容

⑴ 審決の内容は，別紙審決書写しのとおりである。要するに，本願発明は，「山口英ほか３８名，“ｂｉｔ別冊　情報セキュリティ”，共立出版株式会社」（甲７。以下「引用文献１」という。）及び国際公開第９８／４１９１９号（甲８。日本語訳は甲９。以下，「引用文献２」という。）に記載された各発明（以下，引用文献１に記載された発明を「引用発明１」と，引用文献２に記載された発明を「引用発明２」ということがある。）に基づいて容易に発明できたものであるから，特許法２９条２項の規定により特許を受けることができないとするものである。

⑵ 審決が認定した引用発明１の内容

ウィルスに感染した実行形式のプログラムファイル及びウィルスに感染したマクロファイルをもつ文書ファイルにおけるウィルス部の実行をリアルタイムで未然に防ぐウィルスの解析・検出方法であって，

パーソナルコンピュータ上で動作するワクチンソフトウェアによって，

（ｃ１）ＯＳの機能であるファイルＩ／Ｏをフックすることにより，前記パーソ

ナルコンピュータ上におけるファイルの起動や当該ファイルのオープンといったファイルＩ／Ｏを監視するステップ；

（ｃ２）前記ステップ（ｃ１）で監視されたファイルＩ／Ｏと関連している実行形式のプログラムファイル及びマクロファイルをもつ文書ファイル（以下「実行ファイル」という。）の実行前に，前記実行ファイルのウィルス感染の有無を判断するステップ；

（ｃ３）前記ステップ（ｃ２）で前記実行ファイルがウィルスに感染していると判断された場合に，当該ウィルスを特定し駆除可能であれば，当該ウィルスの駆除を行い，当該ウィルスが未知のウィルスであってもユーザに警告を発し，当該ウィルスの危険な動作を未然に防ぐステップ；

を含むことを特徴とするウィルスの解析・検出方法。

(3) 審決が認定した本願発明と引用発明１との一致点

実行ファイルにおける有害情報をリアルタイムで遮断する方法において，

クライアントシステム上で，ウィルス対策プログラムがコンピュータウィルスを含む有害情報をリアルタイムで遮断するステップを含み，

前記ステップが，

（ｃ１）ファイルＩ／Ｏルーチンを奪うことにより前記クライアントシステム上におけるファイル入出力（Ｉ／Ｏ）を監視するステップ；

（ｃ２）前記ステップ（ｃ１）で監視されたファイル入出力（Ｉ／Ｏ）と関連している実行ファイルの有害有無を判断するステップ；及び

（ｃ３）前記ステップ（ｃ２）で有害と判断されたファイルの治療が可能な場合は適切な処理を行い，前記ステップ（ｃ２）で有害と判断されたファイルの治療が不可能な場合は該ファイルの危険な動作を未然に防ぐステップを含むことを特徴とする方法。

(4) 審決が認定した本願発明と引用発明１との相違点

ア　相違点１

「ウィルス対策プログラム」について，本願発明の「有害情報遮断コードモジュール」は，「（ａ）ウェブサーバーとクライアントシステムが相互連結されたコンピューターネットワークにおいて，前記ウェブサーバーがコンピューターネットワークを通じてウェブブラウザーが実行された前記クライアントシステムからの接続要請を受信するステップ」と「（ｂ）前記ウェブサーバーが前記クライアントシステムに，有害情報遮断コードモジュールを伝送するステップ」によって，前記有害情報遮断コードモジュールの伝送が完了すると，前記クライアントシステムによって自動的に駆動されるものであるのに対して，引用発明１の「ワクチンソフトウェア」は，パーソナルコンピュータ上への実装駆動方法が不明な点。

イ　相違点２

「（ｃ３）」のステップにおいて，本願発明は「ステップ（ｃ２）で有害と判断されたファイルの治療が不可能な場合は該当ファイルの実行を中止させる」のに対して，引用発明１は，「ステップ（ｃ２）で実行対象のプログラムファイルがウィルス感染していると判断された場合に，…当該ウィルスが，未知のウィルスであってもユーザに警告を発し，当該ウィルスの危険な動作を未然に防ぐ」ものである点。

第３　審決取消事由に関する当事者の主張

１　原告の主張

(1) 取消事由１（一致点の認定の誤り）

審決には，本願発明と引用発明１との一致点の認定において，以下のとおりの誤りがある。すなわち，

ア　「モニタリング法」について

引用文献１には「モニタリング法」が自動でウィルスを防ぐとの記載はなく，引用文献１の「３．２　動的解析による対策手法」の記載全体からみて，「モニタリング法」は手動で行われる手法であると考えるのが自然である。「モニタリング法」は「メモリ常駐」することが記載されているからといって，「リアルタイム」とはいえないし，「モニタリング法」が自動化されている必然性も蓋然性もない。引用

文献１に記載された「モニタリング法」は手動で行う手法であって，本願発明のようにリアルタイムで実行対象のファイルの問題ある動作を未然に防ぐことはできない。

イ　「ウィルスの発病」について

引用文献１には「実行形式のプログラムファイル」と「マクロをもちうる文書ファイル」とが別々に列挙されており，両者は感染場所が異なり発病メカニズムも異なる。しかるに，審決は，「マクロをもちうる文書ファイル」と「実行形式のファイルプログラム」を同一視し，いずれについてもファイルＩ／Ｏをフックすることを適用できるとして一致点を認定している。

ウ　「ファイルＩ／Ｏをフックすること」について

ウィルスに感染したファイルが読み込まれることはウィルス発病の必要かつ十分な条件ではない。現に，シフトキーを押しながら文書をオープンすると，マクロは実行されないから，「マクロを含む文書」をオープンすることと，マクロの実行とは関係がなく，「ファイルのオープン」は，発病の契機でもない。ファイルＩ／Ｏとマクロの実行は直接関係ないから，ウィルスを検出するためには，ファイルＩ／Ｏではなく，むしろ「マクロの実行」を監視することに動機付けられるというべきである。

また，ＯＳにはファイルＩ／Ｏ以外にも多数の機能があるから，多数のＯＳの機能からファイルＩ／Ｏの機能に着目して選択するにはそれなりの根拠が必要である。したがって，格別の根拠なく，当業者に自明であるとして，「ファイルＩ／Ｏを監視する態様を含む」と認定することは誤りである。

エ　以上のとおり，引用文献１の「モニタリング法」は人が精神活動を発揮させながら手動で行う手法であってリアルタイムで行うものではないし，また，「実行形式のファイル」と「マクロをもちうる文書ファイル」とでは，ウィルスの発病のメカニズムが異なるし，引用文献１の「ＯＳの機能をフックする」との記載をもってファイルＩ／Ｏをフックすることを認定することもできないのであるから，審決

の一致点に関する認定は誤っている。

(2) 取消事由２（相違点１に係る容易想到性判断の誤り）

引用発明１は「モニタリング法」であり，引用発明２は「パターンマッチング法」であるから，引用発明１と引用発明２とでは技術的に共通の分野に属するとはいえない。引用発明２は，イベントが発生してからウィルス検出オブジェクトを実装駆動するものであって，リアルタイムで監視を行う本願発明とはタイミングが逆になっており，引用発明１と引用発明２とを組み合わせることはできない。また，引用文献２だけにＡｃｔｉｖｅＸコントロールやＪａｖａアプレットを用いてワクチンソフトウェアを実装することが記載されていても，これが当時の技術常識であったとは到底いえない。

このように，審決の相違点１に係る容易想到性の判断には誤りがあり，この誤りは審決の結論に影響を及ぼす。

２　被告の反論

(1) 取消事由１（一致点の認定の誤り）に対し

ア　引用文献１の記載によれば，モニタリング法とは，ワクチンソフトウェアがメモリに常駐してプログラムのファンクションコールやＡＰＩの呼出しを監視し，プログラムの実行前に危険な動作を未然に防ぐものであることが理解できるから，当該監視はリアルタイムに行われるものであって，審決に誤りはない。

イ　審決には，「実行形式のプログラムファイルの実行は，当該ファイルの起動といったファイルＩ／Ｏを契機とすること」，及び，「マクロファイルをもつ文書ファイルの実行は，当該ファイルのオープンといったファイルＩ／Ｏを契機とすること」が述べられている。プログラムファイルを起動するためには，通常，ＯＳが当該プログラムファイルをメモリ上にロードすることが必要であり，ロードするに際して，前記プログラムファイルのＩ／Ｏが伴うことは当業者にとって自明の事項であるから，審決の認定に誤りはない。

ウ　原告が主張する引用文献１のシフトキーを押しながら文書をオープンすると

の記載は，単なるなお書きであり，「データファイル（文書）のマクロ部分に感染するウィルス（マクロウィルス）」は，当該ファイル（すなわち，マクロを含む文書）のオープンといったファイルＩ／Ｏを契機として発病する。

エ　ウィルスに感染したファイルを開くと，ウィルスが発病（活動開始）するのであるから，審決において「すなわち，前記「ＯＳの機能をフックすることにより，ファンクションコールやＡＰＩの呼出しを監視する」態様が，ＯＳの機能であるファイルＩ／Ｏをフックすることにより，前記パーソナルコンピュータ上におけるファイルの起動や当該ファイルのオープンといったファイルＩ／Ｏを監視する態様を含むことは，当業者にとって自明である。」と認定した点に誤りはない。

オ　以上のとおり，審決には，引用発明１の内容についての認定の誤り及び一致点についての認定に誤りはない。

(2) 取消事由２（相違点１に係る容易想到性判断の誤り）に対し

プログラムの実装方法として，ＡｃｔｉｖｅＸコントロールやＪａｖａアプレットを用いることは，本願出願前に当業者にとって技術的常識であったこと，及び，ウィルス対策プログラムのクライアント上への実装駆動方法として，ＡｃｔｉｖｅＸコントロールやＪａｖａアプレットを用いることが当業者にとって公知の発明（引用発明２）であったことからすると，ウィルス対策プログラムのクライアント上への実装駆動方法として，ＡｃｔｉｖｅＸコントロールやＪａｖａアプレットを用いることは，当業者であれば容易に想到し得たものである。

引用文献２に記載されたワクチンソフトウェアは，いつ発生するか分からない常時監視が必要なイベントをもトリガとするものであるから，モニタリング法に非常に近接した技術をも包含する。引用発明１の「ワクチンソフトウェア」と引用発明２の「ウィルス検出オブジェクト」とは，ともにウィルス対策プログラムである点において技術的に共通の分野に属する。

以上のとおりであり，原告主張の取消事由２に理由はない。

第４　当裁判所の判断

当裁判所は，審決には原告主張の取消事由はなく原告の請求は棄却されるべきと判断する。その理由は次のとおりである。

１　認定事実

(1) 本願発明に係る特許請求の範囲の記載は，前記第２の２に記載のとおりである。

(2) 本件明細書には，以下の記載がある（甲１）。

「【発明の詳細な説明】

（技術分野）

本発明は，保安システムに関するもので，特にクライアントとウェブサーバーが連結されたコンピューターネットワークにおいて，オンラインでコンピュータウイルスなどの有害情報を診断，治療及び遮断するシステム及び方法に関するものである。」

「【０００４】

このような各種有害情報に対する従来の対処方法としては，基本的に先被害／後復旧方式であった。このような保護政策は，コンピュータシステムが識別されていない有害情報によって被害を被ってから初めてその対処方案（例えば，ワクチンプログラムの開発）を模索する手動的な方式である。このような保護政策における他の短所は，有害情報に対処するための各種ワクチンプログラムなどを各パーソナルコンピュータに手動でインストールしなければならないということであり，コンピュータ利用者には煩わしさがあった。さらに，各種有害情報は絶えず新しく考案されてインターネットを通じて速いスピードで配布されているため，常に最新バージョンのワクチンプログラムを備えるのは容易なことではない。

【０００５】

したがって，現在インストールされているワクチンプログラムでは対処できない新しいコンピュータウイルスのような新種有害情報が利用者のコンピュータシステムに浸透した場合は，これを遮断する方法がなく，このような新種コンピュータウイ

ルス等によるコンピュータシステムの機能麻痺または個人情報の流出被害は不回避なものと認識されている。また，コンピュータ利用者は，確認されていないコンピュータウイルスが発見される度に，最新バージョンのワクチンプログラムを確保するために，有害情報関連専門業者またはオンライン通信会社にアクセスしなければならなかった。しかも，このような最新バージョンのワクチンプログラムをダウンロードした後，手動でインストールしなければならないため，無駄な時間が費やされるという煩わしさがあった。」

「【０００７】

（発明の開示）

本発明は上記問題点に鑑みてなされたものであり，クライアントシステムがコンピューターネットワークを通じてウェブサーバーに接続することで，前記クライアントシステムに有害情報遮断プログラムが自動的に伝送及びインストールされ，クライアントシステムのファイル及び通信パケットの入出力をリアルタイムで監視し，有害情報を能動的に遮断できるオンライン有害情報遮断システム及び方法を提供することを第１の目的とする。」

「【０００９】

上記第１の目的を達成するために，本発明はコンピュータウイルスを含む有害情報を遮断する方法において，（ａ）ウェブサーバーとクライアントシステムが相互連結されたコンピューターネットワークにおいて，前記ウェブサーバーがコンピューターネットワークを通じて前記クライアントシステムからの接続要請を受信するステップ；（ｂ）前記ウェブサーバーが前記クライアントシステムに有害情報遮断コードモジュールを伝送するステップ：及び（ｃ）前記有害情報遮断コードモジュールの伝送完了後，前記クライアントシステムにおいて前記有害情報遮断コードモジュールが自動的に実行され，コンピュータウイルスを含む有害情報をリアルタイムで遮断するステップを含むことを特徴とする。」

「【００１２】

前記ステップ（ｃ）で実行された有害情報遮断コードモジュールは，現在の有害情報遮断コードモジュールの実行状態を別途のウィンドウに表示し，前記ウィンドウを閉じれば，前記有害情報遮断コードモジュールの実行が終了されることが好ましい。前記ステップ（ｃ）で実行される有害情報遮断コードモジュールは，前記クライアントシステムが他のウェブサーバーに接続しようとする場合にも，前記クライアントシステムでそのまま継続して動作することが好ましい。前記ステップ（ｂ）で伝送される有害情報遮断コードモジュールは，Ａｃｔｉｖｅ－Ｘ™またはＪａｖａ™プログラムであることが好ましい。」

「【００２４】

適切には，有害情報遮断コードモジュールは，クライアント１３０で駆動される実行可能なアプリケーションプログラムである。例えば，マイクロソフト社のウィンドウ環境における使用のためのＡｃｔｖｉｅ Ｘ™制御，及びウェブブラウザーで実行され得るジャバアプレット（Ｊａｖａ™ ａｐｐｌｅｔ）及びジャバスクリプト（Ｊａｖａ Ｓｃｒｉｐｔ™）がある。また，高級言語で作成され，オブジェクトコード化されたプログラムをウェブブラウザーとリンクさせて，該当プログラムを実行させてもよい。」

「【００２９】

この実施形態において，図２ｂを参考にすると，ステップ２１０及びステップ２２０が，図２ａを参考にして説明した前記第１実施形態（中略）のような方法で行われる。次いで，クライアント１３０が主に前記第２ウェブサーバー１２０にアクセスする（ステップ２３０）。

【００３０】

前記第２ウェブサーバー１２０は，自分が提供するオンラインサービス情報の他に，有害情報管理サーバー１１０への接続に用いられるハイパーリンク（ｈｙｐｅｒｌｉｎｋ）情報をクライアント１３０に提供する（ステップ２３５）。前記ハイパーリンク情報は，有害情報管理サーバー１１０のフロントホームページ用リンク情報

ではなく，クライアント１３０が別途のウィンドウを通じて前記有害情報管理サーバー１１０から有害情報遮断コードモジュールを直接受信することができるようにするリンク情報であることが好ましい。

【００３１】

次に，クライアント１３０は，第２ウェブサーバー１２０からの前記ハイパーリンク情報に従って，有害情報管理サーバー１１０にＨＴＴＰ要請を提供する（ステップ２４５）。

前記有害情報管理サーバー１１０は，前記クライアント１３０からの前記ＨＴＴＰ要請に対するＨＴＴＰ応答として有害情報遮断コードモジュールを伝送する（ステップ２５５）。

【００３２】

有害情報遮断コードモジュールの伝送が完了すると，前記有害情報遮断コードモジュールはクライアント１３０において自動的に実行され（ステップ２６０），コンピュータウイルスを含む有害情報をリアルタイムで遮断する（ステップ２７０），というのは第１実施形態におけると同様である。

【００３３】

前記有害情報遮断コードモジュールについてより詳しく説明する。図３は，本発明に適用される有害情報遮断コードモジュールの一例の構成を示し，図４は，図３に示した有害情報遮断コードモジュールの動作を説明するフローチャートである。

【００３４】

図３に示すように，有害情報遮断コードモジュールは，入出力管理ユニット３１０，有害情報遮断ユニット３２０及び情報伝達ユニット３３０を含む。また，有害情報遮断コードモジュールは，現在の有害情報遮断コードモジュールの実行状態を表示する別途のウィンドウ３４０と関連し，前記ウィンドウ３４０を閉じれば，有害情報遮断コードモジュールの実行が終了されるのが好ましいというのは上述のとおりである。

【００３５】

前記入出力管理部３１０は，クライアント１３０上におけるファイル入出力（Ｉ／Ｏ）を監視する。前記ファイルＩ／Ｏの監視とは，ファイルＩ／Ｏルーチンを奪って（ｈｏｏｋｉｎｇｕｐ）該当ファイル情報を得ることを意味する。前記入出力管理ユニット３１０は，ネットワークからの有害情報を遮断するために，クライアント１３０上におけるネットワークパケットＩ／Ｏも監視することが好ましい。バックオリフィスウイルスと不法個人情報を流出させることの可能なコンピュータウイルスは，ファイルＩ／Ｏを点検したり，プロセスを点検することによって遮断されもするが，その内容は後述する。適切には，入出力管理ユニット３１０は，クライアント１３０が接続しようとするインターネットアドレスをモニターする機能も持っているため，コンピュータ利用者が猥褻サイトに接続することを遮断する。

【００３６】

有害情報遮断ユニット３２０は，ファイルまたはパケットの有害有無を診断し，そのファイルまたはパケットが有害である場合は，適切な治療を行う。情報伝達ユニット３３０は，有害情報であると判断されたファイルまたはパケットの情報を有害情報管理サーバー１１０に通知する。」

(3) 引用文献１には，次の記載がある(甲７)。

「２．１ 活動開始」

「ＰＣ内に侵入したウィルスは，そのプログラムコードが実行されるまでは何もすることができない。」

「ＣＯＭやＥＸＥなどの実行形式のプログラムに感染するウィルスの場合は，ユーザによって，もしくはＯＳ（Ｏｐｅｒａｔｉｎｇ Ｓｙｓｔｅｍ）やその他のプログラムによって起動されるのを待っている。」

「アプリケーションソフトウェアのデータファイル（文書）のマクロ部分に感染するウィルス（マクロウィルス）の場合は，そのマクロを含む文書が開かれるのを待っている。」

「２．２　感染・増殖」

「ウィルスを感染場所によって分類すると，

（１）ブートセクタ

（２）実行形式のプログラムファイル

（３）マクロをもちうる文書ファイル

の大きく３種類に分けられる。」

「２．２．２　実行形式のプログラムファイル

実行形式のプログラムは，プログラムのスタートアドレスから実行が開始する。」

「ウィルスは感染可能なファイルを検索し，対象ファイルにウィルスコードを追加し，ウィルス部が実行されるようにスタートアドレスのコード（もしくはスタートアドレス自体）を書き換える（図２）。このとき，ウィルス部の実行を終えた後で元のプログラム（宿主）を正しく実行するために，スタートアドレスのコード（もしくはスタートアドレス自体）を別の場所に保管している。その保管場所が特定できれば，感染ファイルからウィルスコードを取り除くこと（ウィルスの駆除）が可能となる。」

「上記のとおり，少なくとも，実行形式のファイルを開いて書込みを行うものは疑うべきであり，さらに常駐終了するならば要注意である。」

「２．２．３　マクロをもちうる文書ファイル

文書ファイルのマクロプログラムは，本来，文書の編集作業をサポートしたり定型業務プログラムを作成するためのものだったが，アプリケーションソフトのマクロ機能に制限がない場合は，バイナリのウィルスと同様，悪意をもったコードを書いて実行することができる」

「ウィルスは，感染対象となる文書ファイルを検索する場合もあるが，多くの場合は，まずアプリケーションが起動時に読み込む特殊なファイルのマクロ部分に感染する。」

「これらのファイルに感染した場合は，ウィルスはアプリケーションの起動時に読

み込まれ，コマンドマクロの機能をフックし，アプリケーションに常駐する。」

「３．対策

これまでも随時対策のポイントを述べてきたが，ここでは改めてウィルスの解析・検出手法を述べる」，「これらのうち，いくつかはワクチンソフトウェアに実装され，自動化されている。」

「３．１　静的解析による対策手法」

「パターンマッチング法（スキャン法）は，解析によって得られた特徴的なコードをブートセクタやファイルから検索する手法である。もちろんこれは既知のウィルスにしか効果はないが，ウィルスを特定し，駆除可能であればそれを行う，ほとんどのワクチンで利用されている基本的な対策手法である。」

「静的ヒューリスティック法は，ウィルスの行動パターンを小さな要素に分解し，検査対象ファイルにそれらの要素がどれだけ含まれているかをチェックする手法である。」

「３．２　動的解析による対策手法」

「モニタリング法は，メモリ常駐型ワクチンともいうべきもので，ＯＳの機能をフックすることにより，プログラムのファンクションコールやＡＰＩの呼出しを監視する。これにより，プログラムの実行前にパターンマッチング法で既知ウィルスの検査を行ったり，未知ウィルスであってもその危険な動作を未然に防ぐことが可能となる。

動的ヒューリスティック法は,モニタリング法に静的ヒューリスティック法を応用したもので,実行中のプログラムの危険度をより正確に判定することができる。」

(4) 引用文献２（日本語訳）には，次のとおりの記載がある(甲９)。

「好適な実施例では，ブラウザ３３０はＭｉｃｒｏｓｏｆｔ社製のインターネットエクスプローラである。クライアント３００はウィルス検出サーバ４００の生成するウィルス検出オブジェクトの形で供給されるプログラムを実行できるエンジンを含み得る。なお,クライアント３００にはエンジンを他の方法でも形成できるが,

この発明ではエンジンをブラウザ３３０と連携して構成し，したがってブラウザ３３０はウィルス検出オブジェクトの形のプログラムの実行に備えて「イネーブルされた」状態にあると考えられる。この好適な実施例ではＭｉｃｒｏｓｏｆｔ社製のプログラミングツールＡｃｔｉｖｅＸをウィルスの反復検出用のウィルス検出オブジェクトの生成に用いている。このＡｃｔｉｖｅＸツールはＡｃｔｉｖｅＸ用語で「ｃｏｎｔｒｏｌｓ」と呼ばれるオブジェクト，すなわちサーバに常駐できクライアントからアクセスできる実行可能なコードを含むオブジェクトの生成の手段である。このｃｏｎｔｒｏｌｓはクライアントにも転送され，クライアントがそのための手段を備えている場合は,そのクライアントで実行される。好ましい実施例では，ブラウザ３３０はＡｃｔｉｖｅＸでイネーブルされ，ウィルス検出オブジェクトはＡｃｔｉｖｅＸ　ｃｏｎｔｒｏｌｓである。ブラウザ３３０には，ウィルス検出オブジェクトの実行可能部分の生成用に種々の代替手段を利用できる。例えば，ブラウザ３３０はＮｅｔｓｃａｐｅ社製のネットスケープナビゲータで構成することもできる。また，ウィルス検出（および処置）オブジェクトは例えばＪａｖａアプレットなどのアプレットで構成できる。Ｊａｖａプログラミングツールは Ｓｕｎ　Ｍｉｃｒｏｓｙｓｔｅｍｓ社から市販されている。」

２　判断

⑴ 取消事由１（一致点の認定の誤り）について

ア　引用発明１の内容

引用文献１に記載された引用発明１は，審決の認定のとおり，次の内容であると認められる。

ウィルスに感染した実行形式のプログラムファイル及びウィルスに感染したマクロファイルをもつ文書ファイルにおけるウィルス部の実行をリアルタイムで未然に防ぐウィルスの解析・検出方法であって，

パーソナルコンピュータ上で動作するワクチンソフトウェアによって，

（ｃ１）ＯＳの機能であるファイルＩ／Ｏをフックすることにより，前記パーソ

ナルコンピュータ上におけるファイルの起動や当該ファイルのオープンといったファイルＩ／Ｏを監視するステップ；

（ｃ２）前記ステップ（ｃ１）で監視されたファイルＩ／Ｏと関連している実行ファイルの実行前に,前記実行ファイルのウィルス感染の有無を判断するステップ;

（ｃ３）前記ステップ（ｃ２）で前記実行ファイルがウィルスに感染していると判断された場合に，当該ウィルスを特定し駆除可能であれば，当該ウィルスの駆除を行い，当該ウィルスが未知のウィルスであってもユーザに警告を発し，当該ウィルスの危険な動作を未然に防ぐステップ；

を含むことを特徴とするウィルスの解析・検出方法。

イ　本願発明と引用発明１の対比

そこで，引用発明１と前記第２の２のとおりの本願発明とを対比する。

引用発明１の「実行ファイル」（実行形式のプログラムファイル及びマクロファイルをもつ文書ファイル）は，本願発明の「実行対象のファイル」に相当する。引用発明１の「ウィルス部」は，本願発明の「有害情報」及び「コンピュータウィルスを含む有害情報」に相当する。引用発明１の「未然に防ぐ」及び「ウィルスの解析・検出方法」は，それぞれ，本願発明の「遮断する」及び「方法」に相当する。引用発明１の「パーソナルコンピュータ」は，本願発明の「クライアントシステム」に相当する。引用発明１の「（ｃ１）」のステップは，本願発明の「（ｃ１）」のステップに相当する。引用発明１の「実行ファイルのウィルス感染の有無」は，本願発明の「実行対象のファイルの有害有無」に相当する。引用発明１の「（ｃ２）」のステップは，本願発明の「（ｃ２）」のステップに相当する。引用発明１の「ステップ(ｃ２)で前記実行ファイルがウィルスに感染していると判断された場合に，当該ウィルスを特定し駆除可能であれば，当該ウィルスの駆除を行い」は，本願発明の「ステップ（ｃ２）で有害と判断されたファイルの治療が可能な場合は適切な処理を行い」に相当する。引用発明１の「ワクチンソフトウェア」と本願発明の「有害情報遮断コードモジュール」とは，ともにウィルス対策プログラムである点で共

通する。引用発明１の「ステップ（ｃ２）で前記実行ファイルがウィルスに感染していると判断された場合に，…当該ウィルスが未知のウィルスであってもユーザに警告を発し，当該ウィルスの危険な動作を未然に防ぐ」において，「当該未知のウィルス」は自動的に駆除（すなわち，治療）できないことは当業者にとって自明であるから，引用発明１の「ステップ（ｃ２）で前記実行ファイルがウィルスに感染していると判断された場合に，…当該ウィルスが未知のウィルスであってもユーザに警告を発し，当該ウィルスの危険な動作を未然に防ぐ」ことと，本願発明の「ステップ（ｃ２）で有害と判断されたファイルの治療が不可能な場合は該当ファイルの実行を中止させる」こととは，ともに，「前記ステップ（ｃ２）で有害と判断されたファイルの治療が不可能な場合は該ファイルの危険な動作を未然に防ぐ」ものである点で共通する。

ウ　本願発明と引用発明１の一致点

以上によれば，本願発明と引用発明１の一致点は次のとおりと認定され，審決の認定に誤りはない。

実行ファイルにおける有害情報をリアルタイムで遮断する方法において，

クライアントシステム上で，ウィルス対策プログラムがコンピュータウィルスを含む有害情報をリアルタイムで遮断するステップを含み，

前記ステップが，

（ｃ１）ファイルＩ／Ｏルーチンを奪うことにより前記クライアントシステム上におけるファイル入出力（Ｉ／Ｏ）を監視するステップ；

（ｃ２）前記ステップ（ｃ１）で監視されたファイル入出力（Ｉ／Ｏ）と関連している実行ファイルの有害有無を判断するステップ；及び

（ｃ３）前記ステップ（ｃ２）で有害と判断されたファイルの治療が可能な場合は適切な処理を行い，前記ステップ（ｃ２）で有害と判断されたファイルの治療が不可能な場合は該ファイルの危険な動作を未然に防ぐステップを含むことを特徴とする方法。

エ　原告の主張について

(ｱ)「モニタリング法」について

原告は，「モニタリング法」は，本願発明とは異なり，手動で行われるもので，リアルタイムでの監視はできないと主張する。

しかし，原告の主張は，以下のとおり失当である。

すなわち，①引用文献１には，主に手動で行うことを主眼としているものの，「これらのうちいくつかはワクチンソフトウェアに実装され，自動化されている。」との説明がされており，同説明部分は，自動化がされていることを前提とした記述であると解され，また，②引用文献１には，「モニタリング法」について，「メモリ常駐型ワクチン」であり，「プログラムのファンクションコールやＡＰＩの呼出しを監視する」と説明がされており，同説明部分も，自動でリアルタイムの監視を行う場合を前提とする記述であると解され，さらに，③引用文献１には，「動的ヒューリスティック法」について，「モニタリング法に静的ヒューリスティック法を応用したもので，実行中のプログラムの危険性をより正確に判定する」と説明がされており，同説明部分も，モニタリング法が「実行中のプログラムの危険性」を判定する自動でリアルタイムでの監視であることを前提とする記述であると解される。

そうすると，引用文献１の「モニタリング法」は実行中のプログラムを自動でリアルタイムに監視するものと解されることになり，原告の主張は採用できない。

(ｲ)「ウィルスの発病」について

原告は，「実行形式のファイルプログラム」と「マクロをもちうる文書ファイル」を同一視して一致点と認定した点には誤りがあると主張する。

しかし，原告のこの点の主張も採用できない。

確かに，引用文献１は，前記１(3)記載のとおり，「実行形式のファイルプログラム」と「マクロをもちうる文書ファイル」とを区別し，それぞれのコンピュータウィルスにより，感染や増殖の機能や機構が異なる趣旨の記載がある。しかし，感染や増殖の機能や機構において異なるコンピュータウィルスが存在する旨の記述は存

在するが，引用文献１の「３．対策」の項では，「実行形式のファイルプログラム」と「マクロをもちうる文書ファイル」を区別して論じていない。異なるコンピュータウィルスのいずれであっても，プログラム又はマクロの実行前にはメモリ上にロードすることが必須であり，ロードに際してファイルＩ／Ｏを伴うことに関して相違はない。そうすると，両者について格別の区別をせずに一致点とした審決の認定に誤りはなく，原告の主張は採用できない。

(ｳ) 「ファイルＩ／Ｏをフックすること」について

原告は，引用文献１の「ＯＳの機能をフックする」との記載から「ファイルＩ／Ｏをフックする」と認定することは，誤りであると主張する。

しかし，原告のこの点の主張も，以下のとおり採用できない。

すなわち，一般に，「ＯＳの機能」は多数の機能を有するが（甲１４，１７），これらの「ＯＳの機能」の中に「ファイルＩ／Ｏ」を含むことは，自明であると解される。そして，プログラムファイルを実行するためには，通常，ＯＳが当該プログラムファイルをメモリ上にロードすることが必要であり，ロードするに際して，当該プログラムファイルのＩ／Ｏを伴うことは当業者にとっては自明である。この点は，引用文献１に「ＰＣ内に侵入したウィルスは，そのプログラムコードが実行されるまでは何もすることができない。」，「ＣＯＭやＥＸＥなどの実行形式のプログラムに感染するウィルスの場合は，ユーザによって，もしくはＯＳ（Ｏｐｅｒａｔｉｎｇ　Ｓｙｓｔｅｍ）やその他のプログラムによって起動されるのを待っている。」，「アプリケーションソフトウェアのデータファイル（文書）のマクロ部分に感染するウィルス（マクロウィルス）の場合は，そのマクロを含む文書が開かれるのを待っている。」と記載されていることからも，明らかである（これはシフトキーを押しながら文書をオープンする際も変わらない。）。そして，モニタリング法が「プログラムの実行前に」プログラムのファンクションコールやＡＰＩの呼出を監視するとされていることからすれば，引用文献１でいう「ＯＳの機能をフックする」との記載は，ＯＳの機能のうちプログラムの実行前に行われる機能をフッ

クすることを含む趣旨と理解するのが自然であり，上記の当業者に自明の事項を前提とすると，「ＯＳの機能」には「ファイルＩ／Ｏ」が含まれることが自明であるといえる。以上のとおり，引用文献１の「ＯＳの機能をフックする」との記載に接した当業者は，「ファイルＩ／Ｏをフックする」ことを含むとものと理解するから，「ＯＳの機能をフックする」との記載を「ファイルＩ／Ｏをフックする」ことを含むものとした審決の一致点の認定に誤りはない。

オ　小括

以上によれば，取消事由１についての原告の主張は，採用することができず，本判決と同趣旨の審決における本願発明と引用発明１との一致点の認定に，原告主張の誤りはない。

(2) 取消事由２（相違点１に係る容易想到性判断の誤り）について

ア　審決の認定した相違点１の内容は，前記第２の３(4)アのとおりであり，相違点１に係る構成について，プログラムの実装方法としてＡｃｔｉｖｅＸコントロールやＪａｖａアプレットを用いることは，前記１(4)のとおり，引用文献２に記載されているから，この引用発明２を引用発明１に組み合わせることは，当業者にとって容易であったといえる。したがって，審決の判断に誤りはない。

イ　原告の主張について

(ｱ) 原告は，引用発明１と引用発明２では技術的に共通の分野に属しないから，相違点１に係る構成に至ることは容易とはいえないと主張する。

しかし，原告の上記主張は，以下のとおり失当である。

すなわち，引用発明１の「ワクチンソフトウェア」は，監視対象となるコンピュータで実行され，ウィルスを検出するものであり，引用発明２の「ウィルス検出オブジェクト」も，同様に監視対象となるコンピュータで実行され，ウィルスを検出するものである点で共通である。

引用発明１と引用発明２とでは，具体的なウィルスの検出手法が，「モニタリング法」と「パターンマッチング法」とで異なるとしても，引用発明１の「ワクチン

ソフトウェア」と，引用発明２の「ウィルス検出オブジェクト」は，いずれも監視対象となるコンピュータで実行され，ウィルスを検出するソフトウェアである点で一致しており，技術的に共通の分野に属するものといえる。そして，ウィルスを検出するソフトウェアの実装駆動方法が，その検出手法の相違により適用できないとする格別の事情も存在しないから，引用発明１の「ワクチンソフトウェア」に，引用発明２の「ウィルス検出オブジェクト」の実装駆動方法を適用することは，当業者にとって困難とはいえない。

(ｲ) 原告は，引用発明１と引用発明２とはタイミングが逆で両者を組み合わせることはできないと主張する。

しかし，原告の主張は，以下のとおり失当である。すなわち，引用文献２に記載されたウィルス検出オブジェクトは，クライアントでトリガリング・イベントが発生した後に送信されるものではあるが，その点を根拠として，引用文献２に記載されているウィルス検出オブジェクトの実装駆動方法を引用発明１に適用することが困難になるとも認められない。

(ｳ) 原告は，引用文献２だけにＡｃｔｉｖｅＸコントロールやＪａｖａアプレットを用いてワクチンソフトウェアを実装することが記載されていても，これが当時の技術常識であったとはいえないと主張する。

しかし，原告の主張は，以下のとおり失当である。すなわち，本件明細書の発明の詳細な説明（【００２４】，【００１２】段落）において，ＡｃｔｉｖｅＸコントロールやＪａｖａアプレットを用いて有害情報遮断コードモジュールを構成する実施例が，これらを用いることについて特段の説明を伴わずに掲げられていること等に照らすならば，ＡｃｔｉｖｅＸコントロールやＪａｖａアプレットを用いたプログラムの実装方法が本願の出願当時技術常識であったと解するのが相当である。

３　結論

以上によれば，原告主張の取消事由１及び２については，いずれも理由がない。原告はその他縷々主張するがいずれも理由がなく，他に審決を取り消すべき違法は

ない。

よって，原告の請求は理由がないから，原告の請求を棄却することとして，主文のとおり判決する。

知的財産高等裁判所第１部

裁判長裁判官　飯　村　敏　明

裁判官　八　木　貴　美　子

裁判官　小　田　真　治