# Introduction to Software Patents

Japan Patent Office
Asia-Pacific Industrial Property Center, JIII

# CONTENTS

## 1. Introduction

More than thirty years have passed since the Japan Patent Office (JPO) examination guidelines were revised in 1975 to allow computer program–related inventions to be patented as “processes.” It was not until the 1990s that software-related patents became widely recognized in Japan. As such, software-related patents in Japan have a history of less than twenty years.

One of the factors behind the recognition of software-related patents was the transformation of the nation’s machine industry into an “information and knowledge” industry. The practice of patenting began with the advent of the Industrial Revolution. While patents were initially issued for mechanical inventions, the scope was expanded to protect inventions in the chemical and electrical fields as well. In response to the information and knowledge industry boom of the 1990s, the scope was again expanded to cover software-related inventions.

Currently, information and telecommunications systems produced by the information and knowledge industry serve a fundamental role in many other industries, and this role is expected to become increasingly important in coming years. Consequently, it is extremely important from a social perspective to adequately protect software-related inventions—the core components of these information and telecommunications systems—in an effort to promote the development of software and, ultimately, the development of the aforementioned systems. At the same time, however, there are those who point out that the exercise of software-related patent rights, which can potentially affect a wide range of information and telecommunications systems that are prevalent in society and industry, may actually hinder technological innovation. In light of this situation, it is important to acquire an accurate understanding of 1) the meaning of software-related patents with respect to the protection and utilization of software-related inventions, 2) what discussions and debates on software-related patents have taken place in the past, 3) how patent protection systems were developed, and 4) what software-related patent issues and challenges exist.

This introduction to software patents provides commentary on software and software-related inventions while offering a broad overview of the history of software since the invention of the computer. It explains the history of the patent protection of software-related inventions and provides details concerning Japanese examination guidelines for software-related patents.

Next, readers will learn about recent patent application and registrations trends, litigation trends, trends in open source, which is closely linked to software-related patents, problems and challenges posed by the status quo, and recent patent protection system trends aimed at resolving these problems and challenges.

Lastly, this guide will discuss what kind of corporate approach is necessary for the creation, protection, and utilization of software-related inventions.

## 2. The Advent of the Computer & Software Trends

Cybernetic research began in the 1940s and resulted in the development of the famous ENIAC (Electronic Numerical Integrator And Computer) and other computers. At the time, however, cybernetics was considered to be a “mental step” (mental process) that closely resembled the exercise of human intellect. Consequently, computer algorithms and other software-related inventions were not deemed eligible for patent protection; the fact that computers rely heavily on mathematics was another factor that prevented patent eligibility.

After IBM lost the antitrust suit filed by the United States Department of Justice in 1969, forcing the company to stop bundling software with its hardware, the software industry was distinctly positioned as a new industry. This “unbundling” of software made it possible for competitors to begin selling rival software products in the market. At the time, however, no copyright laws or patent laws for the protection of software products existed, and this posed a significant challenge for the software industry. In light of this situation, the U.S. Copyright Law was amended in 1980 to include protection for computer software. Japan followed the example of the U.S. by amending its own Copyright Law in 1985 to include computer software protection.

Meanwhile, another field emerged that followed a path separate from that of mainframe computers that originated from cybernetic research: the “device control” field. Originally, mechanical devices were controlled using relay and switch circuits; large-scale control, however, requires a much larger number of circuits, which makes it difficult to accurately grasp the actions of the circuits involved. In order to resolve this problem, a transition was made to electrical control using relay devices. Relay devices were in time replaced by semiconductor devices, which in turn were replaced by integrated circuits; these were ultimately replaced by microcomputers that are the mainstream today. Software makes microcomputer control of mechanical devices possible.

Ideas (inventions) pertaining to microcomputer control are truly software-related inventions. Discussions in Japan on how to protect such inventions sparked debates on the patent protection of software-related inventions; these debates resulted in the creation of patent examination guidelines and the development of a patent protection system.

Mainframe computer application software programs (e.g., applications used in accounting and information systems), which are an extension of cybernetics, were not initially considered patentable for the reasons mentioned above. However, microcomputer control ideas (inventions) came to be recognized as “apparatus” (i.e., product) inventions at an early stage. Because microcomputer control was perceived as a mere replacement of mechanical control circuits, which had theretofore been afforded patent protection, microcomputer control inventions, too, were eventually granted patent protection.

## 3. Hardware & Software

### 3-1. What is Software?

Unlike hardware, software can be easily changed. This easily changeable software is used to control hardware in order to achieve various processing and control objectives. In order to provide desired functionality, computer systems rely on software to control physical hardware, including input devices (e.g., keyboards, mice), central processing units (CPUs), and output devices (e.g., printers), and software used to operate the systems. Software, which is stored in the hardware’s memory device, controls other hardware devices by activating control circuits and mediating control signals.

Based on data that is entered via an input device, CPUs and memory devices are used repeatedly to perform the desired process in an efficient manner. Naturally, however, repeated use of the CPU can have an adverse effect on processing time. For example, the extraction of a high-resolution image from a compressed file takes time to process and therefore requires a different means of processing. In other words, when it is absolutely imperative to process a single set of data (an image in this case) within a certain period of time, repeated use of a single set of hardware is not suitable. In such cases, additional sets of hardware must be installed. In extreme cases, all processing is handled by the hardware rather than the software. This is referred to as large-scale integration (LSI), and it is used widely in image processing and other fields. In this way, there is a complementary relationship between the hardware and software that comprise a computer system; in order to realize the desired processing speed, new hardware must be added in order to reduce the CPU processing load.

While hardware can be defined as something that is physical and used repetitively, video and music data and other digital content fall into the “software” category. Software can be broken down into three categories: 1) programs (procedures), 2) procedures that include data (program/data combination), and 3) data only (this category is excluded from software-related patent consideration). While the broad definition of software includes data, the narrow definition refers to “computer programs.” In this guide, the terms “software” and “computer program” are considered to be equivalent.

- Operating Systems (OS)

An operating system is software that, unlike other types of software, does not need to be easily changed; instead, it is possible to think of an OS as being integrated with the hardware. Based on the above definition of software as an “easily changeable” thing, it is also possible to consider an OS, which does not need to be changed, as a hardware component. Software-related patent protection naturally applies to operating systems, and numerous OS-related patents have already been granted.

### 3-2. Replacing Hardware Functionality with Software

When compressing music data, for example, it is possible to replace the compression functionality of physical hardware (compression circuit) with a computer system that can perform the same task. Running a music data compression program on a personal computer, which is a general-purpose computer system used in daily life, makes it function as a compression circuit.

In this way, functionality that is traditionally provided by hardware is increasingly being realized through personal computer systems. In other words, the functionalities that have hitherto been performed by hardware have been realized through the repeated use of CPUs, memories, and other functions of the personal computer in accordance with the predetermined procedure. However, as mentioned above, jobs that involve numerous parallel processing tasks (e.g., image processing) require more time to complete due to the vast number of procedures (steps) involved. Special LSI hardware is frequently used to solve this problem.

Example: Train fare collection systems

Contactless smart card (IC card) technology is being rapidly implemented in train fare collection systems. These systems use the DES (Data Encryption Standard) cipher to prevent the counterfeiting and improper use of train tickets. When passengers swipe their cards over the automatic wicket (ticket gate) reader, the system must be able to process the smart card data rapidly in order to prevent human traffic jams from forming behind the gates.

Generally speaking, every single transaction must be processed within approximately 200 milliseconds. However, when processing is handled by software, the CPU load is so large that even fast CPUs take an extremely long time to complete the required tasks; this is expected to become a major problem in the future as the volume of data to be processed increases and authentication processes become more complex.

However, according to some reports, it is possible to realize a processing speed of roughly 300 microseconds by using a field programmable gate array (FPGA) to handle DES encryption processing. This hardware-based solution is eighty times faster than the time required for software-based processing (approx. 25 milliseconds).

(Source: “The Realization of Hardware-based DES Encryption/Decryption Technology” (Omron Technics, Vol. 43, No. 3, Serial Issue No.143; 2003))

To summarize the above, 1) hardware and software work in concert to realize the desired functionality, 2) most functionality that can be realized by hardware alone can be replaced by a hardware-software combination, and 3) most functionality realized through a hardware-software combination can be realized by hardware alone. In other words, we have realized the required functionalities easily and efficiently through the combined use of hardware and software, since we have found that those approaches can make our lives more convenient and more efficient in terms of cost performance.

Based on all of the above, one may find illogical the notion that hardware-related inventions are patentable while inventions related to software, which has a complementary relationship with hardware, are not patentable. Nevertheless, there are various challenges associated with the protection of intangible software-related inventions, including those posed by the relationships with mental steps, mathematics, and prior art.

### 3-3. Computer Programs & Algorithms

As the name suggests, mathematical algorithms were traditionally used in the world of mathematics. Currently, however, the term “algorithm” is used to refer to a broad range of logical problem-solving procedures.

Computer programs, based on procedures (i.e., algorithms), are sequences of commands. When creating a computer program, it is necessary to determine what algorithm(s) will be used in advance. In publishing terms, this is similar to laying out a basic scenario before writing the actual novel. Creating the scenario is the most important phase; if the original scenario is high quality, then the final novel (computer program) will be high quality as well. In this example, “scenario” is the equivalent of a hardware design blueprint or technological concept. Consequently, ideas related to algorithms (scenarios) are eligible for patent protection as inventions.

### 3-4. Fields in Which Software Is Used

It would not be an exaggeration to say that computer systems are used in all types of industries. Therefore, it can also be said that software—a vital component of computer systems—is also used in most industrial sectors. The following data was acquired by performing a search for Japanese FI-Section unexamined patents published between January and June 2007 that included the keyword “program” in the “invention title” field (see Table 3-1).

Table 3-1. Software Patent Applications Categorized by Major FI-classes

| SECTION (FI) | Class (FI) | Number |
|---|---|---|
| SECTION A — Human necessities | A61B | 142 |
| | A63F | 358 |
| SECTION B — Performing operations; transportation | B41J | 822 |
| | B60R | 30 |
| SECTION C — Chemistry; metallurgy | C23C | 8 |
| SECTION D — Textiles; paper | D05B | 4 |
| SECTION E — Fixed constructions | E02D | 6 |
| | E05B | 7 |
| SECTION F — Mechanical engineering; lighting; heating; weapons; blasting | F24F | 14 |
| | F24H | 14 |
| SECTION G — Physics | G03G | 114 |
| | G05B | 115 |
| | G06F | 3,084 |
| | G06K | 147 |
| | G06T | 653 |
| | G09B | 205 |
| | G10L | 129 |
| | G11B | 144 |
| SECTION H — Electricity | H04B | 100 |
| | H04L | 365 |
| | H04M | 152 |
| | H04N | 1,132 |

The largest number of patent applications is in Section G (Physics), followed by Sections H and B. As the table shows, applications for software-related invention patents have been recognized in all FI-class sections (A through H). This helps to substantiate the previous statement that software is used in all industrial sectors.

Table 3-2 shows the fields in which software is used based on the purpose of use.

Table 3-2. Fields in Which Software Is Used

| Field | | Specific Examples |
|---|---|---|
| Standalone computer systems | ・Computing | ・OS-related<br>・Game software–related |
| | ・Business systems (Systematization, etc.) | ・Production management<br>・Logistics management |
| Distributed computer systems (multiple computers linked by a communication network) | ・Common elements | ・Data compression<br>・GUI<br>・Security-related |
| | ・E-commerce | ・B-to-B<br>・Online shopping |
| | ・Payment/settlement | ・Electronic money–based settlement |
| | ・Finance/insurance | ・Financial asset management |
| | ・Management | ・Management information systems |
| | ・Service industry | ・Online auctions |
| Computer systems used to create communication networks | ・Communications services | ・Mobile phones<br>・E-mail<br>・VoIP |

As shown in Table 3-2, there are standalone computer systems and distributed, multi-computer systems in which computers are linked together via a communication network; the latter is currently the most prevalent. The reason for separating computer systems into these categories is to distinguish "computing" technologies used to expedite the execution of computer programs (e.g., applications) from other technologies. The majority of computer systems used in many industrial fields are positioned as "business systems" used to streamline and expedite business tasks associated with the industry in question. Consequently, many business systems comprise multiple computer systems that are connected via internal corporate telecommunications networks (e.g., LANs). A simple example of this is a corporate network that comprises multiple workstations (standalone systems) that are connected via a computer system that functions as a server.

As the table also shows, business systems and common elements (graphical user interface [GUI] software, security software, and other types of software) apply to both standalone and distributed computer systems.

Computer systems used to provide mobile phone services, e-mail services, and other telecommunications services can also be considered as telecom industry “business systems (Systematization, etc.).” However, because such systems are used to create telecommunications business, and because they comprise multiple distributed computers connected via networks, a third category was created to distinguish them from “standalone” and “distributed” systems.

With the exception of “business system (systematization, etc.)” and “common elements,” all of the specific examples listed in Table 3-2 are business method–related. Based on this, it is possible to say that software-related patents are closely tied to business. Therefore, business method–related patents became a hot topic in Japan in 2000 as a means of generating profit. At the time, numerous patent applications for Internet business method–related inventions (e.g., new business models) made possible by distributed computer systems were submitted. Business method-related inventions will be covered in more detail in Section 7-4.

## 4. Protection of Computer Programs under Copyright Law & Examples of Copyright Disputes

### 4-1. Protection of Computer Programs under Copyright Law

When explaining software-related patents, it is important to discuss the protection of computer programs under copyright law. Computer programs were given protection under the U.S. Copyright Law in 1985. After that, the U.S. strongly urged Japan and other nations to introduce similar protection systems. Discussions on the development of computer program protection systems began around the time that IBM stopped bundling software with hardware, and because neither the Patent Law nor the Copyright Law were thought to provide adequate protection, special legislation was given consideration.

The eventual decision to protect computer programs under the U.S. Copyright Law can be attributed to the declining competitive strength of the nation’s industries in the early 1980s. At the time, Japanese and German industries were becoming increasingly competitive, and the U.S. believed that boosting the competitive strength of its robust software industry would be an effective way to quickly revive the competitive strength of other national industries. Based on this view, the U.S. decided to follow a strategy whereby it would request that many other nations implement systems for protecting computer programs in an expeditious manner; it also determined that such systems should be capable of providing long-term protection. As a result of this movement, Japan revised its own Copyright Law in 1985 to include computer program protection, which came into force in 1986.

Japan's Copyright Law defines a [copyrightable] "work" as "a production in which thoughts or sentiments are expressed in a creative way, and which falls within the literary, scientific, artistic, or musical domain" (Article 2(1)(i)). Additionally, the law stipulates various rights for the purpose of protecting the economic benefits associated with works; some representative examples of these are the "right of reproduction," the "right of public transmission," and the "right of screen presentation." These rights are referred to as "author's intellectual property rights" and are transferable.

The Japanese Copyright Law also stipulates non-transferable rights (moral rights) that are specific to the author. These include the right of authorship, the right of making the work public, and the right to maintain integrity. Japanese copyright protection of author's works extends fifty years beyond the death of the author (seventy years for cinematographic works) or fifty years beyond the public disclosure of a company-owned work (seventy years for cinematographic works).

If a computer program displays creativity as a work that is based on academic concepts, then it is eligible for protection under Japan's Copyright Law. The law defines a computer program as "an expression of combined instructions given to a computer so as to make it function and obtain a certain result." However, considering the fact that the Copyright Law is a system for protecting works that appeal to human emotions, computer programs seem to be "out of place." Consequently, the following special provisions were added.

### (1) Right to Maintain Integrity

As one of the moral rights of an author, the right of integrity protects a work from being altered, distorted, or mutilated against the will, and without the permission, of the author. However, the right of integrity is meaningless if the computer program in question cannot be used by the hardware for which it is intended. Therefore, a provision was added to Article 20 to stipulate modifications to which the right of integrity does not apply.

> "A modification which is necessary to enable the use on a particular computer of a computer program work that is otherwise unusable on said computer, or to make more efficient use of a computer program work on a computer." (Article 20(2)(iii))

### (2) Reproduction, Etc. by the Owner of a Copy of Computer Program Work

Computer programs are rendered unusable when the media in which they are stored (e.g., CD-ROMs, floppy disks) are lost or damaged. Furthermore, in some cases, computer programs must be modified in different ways to make them usable on users' computers. In short, there are cases in which computer programs must be backed up or adapted. In order to make provisions for such cases, the following addition was made:

> "The owner of a reproduction of a computer program work may make reproductions or adaptations (including reproductions of derivative works created by means of such adaptation) of said work if and to the extent deemed necessary for his own exploitation of said work on a computer." (Article 47-2)

Article 10(3) of Japan's Copyright Law stipulates that copyright protection shall not be extended to any computer programming language, rule, or algorithm. A "programming language" (e.g., FORTRAN, BASIC, C, C++) is defined as "letters and other symbols, as well as the systems for the use thereof, which are used to express a computer program." A [computer program] "rule" is defined as "a special rule that applies to how a programming language, as defined in the preceding item, is used by a particular computer program;" expressions based on rules that are essential to the creation of a computer program are not eligible for copyright protection. A [computer program] "algorithm" is defined as "a method of combining, in a computer program, the instructions given to a computer;" as such, algorithms are not deemed eligible for copyright protection. However, it should be noted that algorithms are the equivalent of the aforementioned "scenarios" and are considered patentable as software-related inventions under Japan's Patent Law.

## 4-2. Representative Examples of Computer Program Copyright Disputes

As a result of various representative examples of disputes related to computer program copyrights in the U.S., a broader interpretation of the Copyright Law developed in the 1980s; the law eventually evolved into today's standard in the 1990s.

As mentioned above, copyright law was originally meant to protect "expressions," not ideas. In the 1985 case of Whelan v. Jaslow (U.S.), the court ruled that copyright law protection extends to the "structure, sequence, and organization (SSO)" of the computer program; however, this precedent was widely criticized. To improve upon this precedent, in the 1991 case of Computer Associates vs. Altai, the court proposed the use of a three-step (abstraction, filtration, and comparison) test to determine substantial similarity between computer programs in copyright infringement cases. As a result of adopting this type of test, many computer program–related copyright infringement cases have been handled as a matter of protecting "expressions," instead of protecting "ideas." The 1991 ruling set a precedent for many cases that followed, and is still used as a guideline in infringement cases today.

(1) Whelan vs. Jaslow (1986)

Whelan Associates, the author of a dental clinic business application written in the EDL programming language called Dentalab, sued Jaslow Lab, the author of a similar computer program written in BASIC called Dentcom, for copyright infringement. The U.S. Court of Appeals ruled that copyright law protection extends to the "structure, sequence, and organization (SSO)" of the computer program.

(2) Computer Associates vs. Altai (1992)

This was a case to determine whether or not Altai's OSCAR 3.5, a job scheduling program for IBM mainframe computers, was substantially similar to Computer Associates' ADAPTER program. The U.S. Court of Appeals proposed the use of a three-step (abstraction, filtration, and comparison) test to determine substantial similarity. The court ultimately found that OSCAR 3.5 did not infringe upon the ADAPTER copyright.

(3) Lotus vs. Borland (1995)

Lotus sued Borland for copyright infringement, claiming that the menu structure of the latter's Quattro Pro spreadsheet software infringed upon the copyright of its own Lotus 1-2-3 spreadsheet software. In the first trial, a district court ruled that the menu structure of Lotus 1-2-3 was copyrightable and found Borland guilty of copyright infringement. The U.S. Court of Appeals found that menu structure is not eligible for copyright protection and reversed the district court's ruling.

(4) Apple vs. Microsoft (1994)

Apple Computer sued Microsoft for copyright infringement based on the similarity of the Windows GUI (Graphical User Interface) to the Apple Macintosh GUI. The U.S. Court of Appeals stated that user interface copyright infringement is recognized only when the two interfaces in question are substantially identical. As such, the court upheld an earlier district court decision, which found that no infringement had taken place.

## 5. The History of the Protection of Software-Related Inventions under Patent Law

As the limitations of computer program protection under copyright law became more evident, the appeal of patent protection as a viable solution grew. In Japan, patent protection was first applied to ideas (inventions) related to the microcomputer control of mainframe computers before it was applied to mainframe computer applications. As discussed previously, this was because microcomputer control was perceived as simply replacing mechanical control circuits, which had theretofore been granted patent protection. This made it easier to patent microcomputer control ideas as “apparatus” (product) inventions.

In this section, we will take a look at the history of the protection of software-related inventions under patent law in Japan, the U.S., and Europe.

### 5-1. The History of Patent Protection in Japan

In Article 2 of Japan’s Patent Law, an “invention” is defined as “a highly advanced creation of technical ideas utilizing the laws of nature”; patents are not granted for inventions to which this definition does not apply. Unlike hardware-related inventions, there is no direct link between software-related inventions and the laws of nature; this made it difficult to determine whether or not software-related ideas actually utilize the laws of nature.

In 1975, the Japan Patent Office (JPO) published its “Examination Guidelines for Computer Software–Related Inventions – Part 1” and revealed that software comprising methods that utilize the laws of nature would be eligible for patent protection as “process” inventions. Then, in response to the increased use of microcomputers in numerous devices, the JPO in 1982 published “Implementing Guidelines for Inventions Related to Microcomputer-Applied Technology” based on the understanding microcomputers utilize software to serve multiple functions; under this new policy, the means used to realize each microcomputer function were deemed eligible for patent protection as “apparatus” inventions.

Next, in 1993, the JPO published Part VII, Chapter 1 of its Examination.

Guidelines for Patents and Utility Models entitled “Computer Software-Related Invention Examination Guidelines.” Under the new revision, patent eligibility was extended to include not only cases in which software-based information processing per se utilizes natural laws but also cases in which hardware resources are used in processing—even if the information processing per se does not utilize natural laws.

In 1997, the JPO published Chapter 1 of its Implementing Guidelines for Inventions in Specific Fields entitled "Computer Software–Related Inventions," which provided patent eligibility for recording media as "product" inventions. These implementing guidelines were intended for use in determining whether or not software-based processing utilizes natural laws. The following are examples of cases where utilization of natural laws is recognized:

1. Hardware resource control or processing involved in control
2. Information processing based on the physical or technical character of the subject matter
3. Processing that involves the use of hardware resources

As indicated above, it may be said that Japanese patent law evolved over the years as a result of debates on whether or not software-related inventions should be deemed patentable under the Patent Law. Let us next take a detailed look at the treatment of software-related inventions under JPO examination and implementing guidelines.

(1) Examination Guidelines for Computer Software–Related Inventions – Part 1

Under these guidelines, published in 1975, the subject matter is considered to be an invention only if the "cause and effect relationships" of the "techniques" needed to make a computer perform desired tasks are based on the laws of nature. Additionally, only computer program "processes" could be patented; the computer programs per se and computing actions were deemed ineligible. In other words, computer programs per se were deemed ineligible for patent protection because they were "extremely abstract" subject matter. Eligibility was not extended to media claims either as media inventions were perceived as mere storage for computer procedures.

(2) Implementing Guidelines for Inventions Related to Microcomputer-Applied Technology & Examination Treatment of OS-Related Technology (1982)

These additions were meant to supplement the aforementioned "Examination Guidelines for Computer Software–Related Inventions – Part 1." The new implementing guidelines were created in response to a rapid increase in the number of claims involving microcomputer-applied technology. While the "Part 1" guidelines limited the scope of patent claims to "processes," the 1982 revisions extended the scope to include "apparatus" claims. In other words, information processing and control that is realized by a microcomputer is also considered to be realized by an aggregation of various functions. Consequently, it is clearly stated that microcomputer-related inventions are eligible to make a claim as an apparatus that has constituent features based on the means used to realize the functions.

### (3) Examination Guidelines for Inventions in Specific Fields (1993)

Chapter 1 of the Examination Guidelines for Inventions in Specific Fields Part VII, 1993 Revision of the Examination Guidelines for Patent and Utility Manuals, published by the JPO in 1993, was created to cover computer software–related inventions. The guidelines stipulated in this chapter were intended to replace past guidelines as consolidated versions thereof. The subject matter is considered to utilize the laws of nature only if the conditions in (I) and (II) below are fulfilled.

(I) Inventions that feature software-based information processing that utilizes laws of nature

(i) Control of hardware resources or processing involved in hardware resource control

(For example, when software is used to control an engine [hardware resource], the control and processing involved is recognized as utilizing the laws of nature.)

(ii) Information processing based on the physical or technical character of the subject matter

(For example, when software is used to process [e.g., enhance] image data that is procured using an image scanner, the processing [enhancement] is based on the physical properties of the image data procured using the scanner, thus utilizing the laws of nature.)

(II) Inventions that feature the utilization of hardware resources

In cases where "mere use of hardware resources" does not apply, it is deemed that the invention utilizes hardware resources (or utilizes the laws of nature). Even in cases where software-based information processing per se is not recognized as utilizing laws of nature, inventions that feature the utilization of hardware resources are recognized as utilizing natural laws. Even in cases of mathematical processing, for example, processing based only on economical properties (e.g., sales forecasts) and playing video games, etc. using software, utilization of the laws of nature shall be recognized if the ways in which the computer hardware resources are used by the computer and how the processing is conducted have been made clear.

In cases where software-based information processing does not utilize the laws of nature (e.g., information processing based on economic laws, commercial methods, artificial agreements, or mathematical formulas) and hardware resources are not utilized (including the "mere use of hardware resources"), the invention is not recognized as utilizing the laws of nature.

(4) Implementing Guidelines for Inventions in Specific Fields (1997)

These guidelines, published by the JPO in 1997, recognized claims related to "computer-readable storage media." Under the guidelines, a medium is considered to be a patentable invention for hardware control even if the medium is not bundled with a device or machine; programs stored on floppy disks, CD-ROMs, and other recordable media are also considered eligible for patent protection (i.e., "medium patent").

(5) Revised Examination Guidelines for Inventions in Specific Fields (2000)

In December 2000, the JPO published the revised version of its Examination Guidelines for Patents and Utility Models. These revised guidelines were included in Part VII, Chapter 1 of the Examination Guidelines for Patents and Utility Models entitled "Computer Software–Related Invention Examination Guidelines," which was updated in January 2001. They expand on the "medium patent" eligibility stipulated in the 1997 guidelines, removing the requirement that a program must be stored on a computer-readable medium to be considered patentable; as a result, computer programs could now be patented as "product" inventions (or could be stated as "product invention" in the claim).

Furthermore, based on the concept of a software-related invention as "the creation of technical ideas that utilize the laws of nature," software-related inventions were now recognized in cases where software-based information processing is concretely realized using hardware resources (e.g., CPUs, memory). This refers to cases involving the following steps: 1) the software is read by the computer, 2) the software and hardware resources work in concert to perform mathematical calculations or processing in accordance with the intended purpose, and 3) as a result, an information-processing device (machine) or method of operation is created for the intended purpose.

(6) Revisions of the Japanese Patent Law

In conjunction with the revision of the Japanese Patent Law in 2002, computer programs became statutory "product" inventions (Article 2(3)(i)).

## 5-2. The History of Patent Protection in the United States

Title 35, Section 101 ("Inventions Patentable") of the U.S. Code (35 U.S.C. 101) states that "whoever invents or discovers any new and useful process, machine, method of manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title." Consequently, those related to abstract ideas, the laws of nature, and natural phenomenon are not patentable.

Around the time when debates concerning the protection of computer programs began to take place in the U.S., sentiment regarding the patent protection of computer programs was negative. That changed, however, following the case of Diamond vs. Diehr (1981) in which the U.S. Supreme Court found that software-related inventions were patentable. In 1996, the U.S. Patent and Trademark Office (USPTO) published its Examination Guidelines for Computer-Related Inventions, which provided clear guidelines for patent protection. Then, in the case of State Street Bank vs. Signature Financial Group (1998), the U.S. Court of Appeals ruled that software-related inventions that are related to business methods are eligible for patent protection.

Main events in the history of U.S. patent protection:

(1) In 1966, a report submitted by President Johnson's Commission on the Patent System recommended against granting patents for computer programs. The commission put forth four reasons, including one that a computer program is not a "process" stipulated in the Patent Law.

(2) In the 1981 case of Diamond vs. Diehr, the court found that software-related inventions that utilize mathematical formulas or algorithms are patentable if they work in concert with other devices or processes. This ruling paved the way for the granting of patents for software-related inventions.

(3) In 1981, the USPTO published its Examination Guidelines for Computer Software Patent Application. As such, protection under the law is not necessary. By switching the focus of examinations from computer programs themselves to "computer-related inventions," the basis for computer software patents was established.

(4) In the Alappat case (1994), the U.S. Court of Appeals ruled that a software-related invention is patentable in combination with a computer in cases where the mathematical algorithm involved produces a useful, concrete, and tangible result.

(5) In 1996, the USPTO published its Examination Guidelines for Computer-Related Inventions. The main points are as follows:

- Patentable subject matter must have a practical application and be in the technological arts.
- A mere abstract idea or a process that manipulates a mathematical algorithm does not constitute an invention. However, if the idea or algorithm is practically applied, it constitutes a process claim.
- While a data structure is non-statutory subject matter, it does constitute an invention if it is stored on a computer-readable medium.

(6) In the case of State Street Bank vs. Signature Financial Group (1998), the U.S. Court of Appeals ruled that software-related inventions that are related to business methods are patentable if they produce a useful, concrete, and tangible result. This ruling established the patentability of business method inventions.

### 5-3. The History of Patent Protection in Europe

Article 52(2) of the European Patent Convention (EPC) stipulates that the following shall not be considered as inventions:

(a) Discoveries, scientific theories, and mathematical methods;
(b) Aesthetic creations;
(c) Schemes, rules, and methods for performing mental acts, playing games, or doing business, and programs for computers;
(d) Presentations of information.

Additionally, Rule 29(1) of the Implementing Regulations to the Convention on the Grant of European Patents states that an invention must have technical character to be considered patentable.

Main events in the history of European patent protection pertaining to software-related inventions:

(1) In the 1985 Vicom case, the European Patent Office (EPO) Board of Appeal recognized the patentability of systems that include computer programs. Based on this, the EPO established examination guidelines for software-related inventions in the same year, establishing patent protection eligibility for software that provides a technical contribution to the conventional art.

(2) In the Sohei case (1995), the EPO Board of Appeal introduced a new judgment criteria that technical considerations have to be involved in solving the particular problem. Based on this, if a technical contribution to the art is found either in a technical effect or a technical problem (to be) solved, the invention may not be excluded from patentability under Article 52.

(3) In 1997, the European Commission published its “Green Paper on the Community Patent and the Patent System in Europe.” Based on the realization that adequate protection of software-related inventions is crucial to the development of European industries, this Green Paper stated that the Commission would consider excluding computer programs from EPC Article 52(2) in order to allow the patentability of computer programs.

(4) In the 1998 IBM case, the EPO Board of Appeal introduced the concept of “further technical effects”—criteria for evaluating the technical character of a computer program; this made it possible to patent computer programs that have technical character. Furthermore, with respect to claim form, the board ruled that computer programs could be considered patentable regardless of whether the claim is for a computer program per se or for a computer program stored on a computer-readable medium.

(5) In the Pension Benefit System Partnership case, the EPO Board of Appeal recognized the patentability of a claim pertaining to a pure financial, economic process used in pension management (September, 2000 ). It found that, while claims for an economic concept or business method per se are not considered patentable under EPC Article 52(1), if the claim is presented as a concrete apparatus (e.g., a computer) that utilizes said business method, it can be considered eligible for patent protection.

(6) The Diplomatic Conference on the Revision of the European Patent Convention (November 2000) was unable to reach a consensus among major nations to remove computer programs from the list of non-inventions in EPC Article 52(2).

(7) In July 2005, the European Parliament overwhelmingly rejected the Computer-implemented Inventions (CII) Directive, which removed computer programs from the list of non-inventions in EPC Article 52(2). After the initial version was proposed in 2002, the legislation was repeatedly revised to reflect the opinions of both those who agreed with and opposed it. Reportedly, there are no plans to submit a further revised draft in the future.

In these ways, the European Union continues to work toward resolutions concerning 1) whether or not software-related inventions should remain non-inventions under EPC Article 52(2) and 2) what kinds of inventions should be considered patentable. Under the status quo, software-related inventions must be endowed with technical character and exhibit further technical effects; otherwise, many believe that it is unlikely they will be considered eligible for patent protection.

[Technical Character]

The following criteria are used to determine whether or not an invention possesses a technical character:

(a) A technical problem to be solved;
(b) Technical features (characteristics) for solving said technical problem, and/or;
(c) Technical effect of the invention;
(d) Technical considerations or technical knowledge is required to implement the invention by computer.

[Examples of Further Technical Effects]

(1) Acceleration of computer program execution
(2) Image resolution improvement in image-processing programs, etc.
(3) Acceleration of data transmission
(4) Improved efficiency of data filtering or other filtering
(5) Improved screen interfaces in business management systems, etc.
(6) Simplified image (e.g., computer graphics) manipulation
(7) Improved efficiency of data compression algorithms
(8) Improved memory utilization

## 6. Japanese Examination Guidelines for Software-Related Inventions

This section discusses main points concerning examination guidelines for software-related inventions in accordance with Part VII ("Implementing Guidelines for Inventions in Specific Fields"), Chapter 1 ("Computer Software–Related Inventions") of the JPO's Examination Guidelines for Patents and Utility Models.

These examination guidelines comprise three main sections: 1) "Detailed description requirements of the specification,” 2) “Requirements for patentability,” and 3) “Examples.” More specifically, these three sections are intended to explain matters that require special judgments or treatment in examinations of inventions that require software for working. In this section we will take a look at 1) and 2) above.

## 6-1. Description Requirements

### 6-1-1. Scope of Claim

A software-related invention can be expressed as a chronologically connected sequence of processes or operations (procedures); it can also be expressed as the multiple functions that the invention serves. This means that a software-related invention can be expressed as a “process” or “product” (apparatus). Additionally, as a product invention, a software-related invention can also be expressed as a “program” that specifies multiple functions performed by a computer and as a computer-readable “storage medium” on which a program is stored. In this way, software-related inventions can be expressed in many forms. Consequently, it is necessary to express such an invention in multiple forms based on the way in which the invention is exploited. In order to ensure that patent rights are adequately enforceable, patent applicants express software-related inventions in the four different forms specified above: process, apparatus, program, and storage medium. In addition to these four, applicants also frequently express inventions as a “method” or “system;” in these cases, however, such expressions are considered as “product” inventions.

Some examples of expressing an invention as a program:

A. “The program makes the computer execute procedure A, procedure B, procedure C, and so on…”

B. “The program makes the computer function as means A, means B, means C, and so on…”

C. “The program makes the computer realize function A, function B, function C, and so on…”

Article 36(6)(ii) of the Japan Patent Law stipulates that “the invention for which a patent is sought is clear.” Inventions in the following categories are considered to be in violation of this provision.

- "The invention for which a patent is sought is unclear resulting from the description of the claim itself being unclear."
- "The invention for which a patent is sought is unclear resulting from the technical meaning of matters defining the invention being incomprehensible."
- "The invention for which a patent is sought is unclear resulting from matters defining the invention not being related technically."
- "The category of an invention for which a patent is sought is unclear, or something that falls in neither products, processes, nor the process of manufacturing the product is stated in a claim."
- "When the scope of the invention is unclear as a result of using an expression where the standard or degree of comparison is unclear."
- "Where 'an intended result to be achieved' is used to define an invention for which a patent is sought whereas nothing concrete (concrete means, concrete articles, concrete processes, etc.) can be conceived even if taking into consideration the common general knowledge as of the filing."

### 6-1-2. Detailed Description of the Invention

With respect to the description of the invention, the Japan Patent Law stipulates "the statement shall be clear and sufficient as to enable any person ordinarily skilled in the art to which the invention pertains to work the invention" (Article 36(4)). In other words, the detailed description must be clear enough so that any person who is able to use ordinary technical means and exercise technical skills in the field that applies to the invention is able to implement said invention based on the description including the part other than the claims, matters specified in drawings, and technical knowledge that exists at the time of filing. This is commonly referred to as the "enablement requirement." In the following cases, descriptions are considered to be in violation of the enablement requirement; patents are not granted in these cases.

- When uncommon technical terms, abbreviations, symbols, etc. are used in the specification without definition, thus preventing the implementation of the invention.

- When the procedure or function associated with the invention is merely described in an abstract manner, thus preventing the implementation of the invention.
- When hardware or software that realizes the function of the invention is merely explained using functional block diagrams or general flowcharts, thus preventing the exploitation of the invention.

In addition to the enablement requirement above, a "ministerial ordinance" requirement also exists. It is defined as follows:

> "Statements of the detailed description of the invention which are to be in accordance with an ordinance of the Ministry of Economy, Trade, and Industry under Patent Law Section 36(4) shall state 'the problem to be solved by the invention' and 'its solution,' or other matters necessary for a person having ordinary skill in the art to understand the technical significance of the invention. (Section 24bis of Regulation under Patent Law)"

Consequently, patents are not granted in cases where the aforementioned "person" is unable to understand what the problem to be solved (and the solution) is; such cases are found to be in violation of the ministerial ordinance.

### 6-2. Patentability Requirements

In an examination of patentability requirements, the claimed invention is identified on the basis of the description in a claim. The significance of matters (terms) used to define the invention is interpreted while taking into account the descriptions of the specification, drawings, and the common technical knowledge that exists at the time of filing.

Among the various patent requirements for software-related inventions, the "statutory invention" and "inventive step (nonobviousness)" requirements, discussed below, are very important.

#### 6-2-1. Statutory Invention Requirement

To be considered as a "statutory invention" under the Patent Law, the claimed invention must be "a creation of technical ideas utilizing the laws of nature." The basic concepts used to determine whether the software-related invention constitutes "a creation of technical ideas utilizing the laws of nature" are as follows:

(1) "Where information processing by software is concretely realized by using hardware

resources, said software is deemed to be “a creation of technical ideas utilizing the laws of nature.”

(2) “Where (1) above is satisfied, the information-processing device (machine) and operational method thereof, which work in concert with said software, and the computer-readable storage medium on which said software is recorded are also deemed to be ‘creations of technical ideas utilizing the laws of nature’.”

This means when 1) the software is read by the computer, 2) the software and hardware resources work in concert to perform mathematical calculations or processing in accordance with the intended purpose, and 3) as a result, an information-processing device (machine) or method of operation is created for the intended purpose, said software is considered to constitute “a creation of technical ideas utilizing the laws of nature.”

The aforementioned 2) is crucial here. The information processing by software must be concretely realized using hardware resources (e.g., CPU, memory). Mere use of a computer by software is not deemed as “working in concert.” Caution must be exercised to ensure that the invention is considered in this light. In the following case, for example, the invention is not deemed to constitute “a creation of technical ideas utilizing the laws of nature” because information processing by software is not concretely realized using hardware resources.

> “A computer comprising an input means to input document data, a processing means to process the entered document data, and an output means to output the processed document data; wherein said computer prepares a summary of the entered document by using said processing means.”

6-2-2. Inventive Step (Nonobviousness)

(1) Determination of Inventive Step

Whether or not an invention involves an inventive step is determined by taking into consideration what a person 1) with an accurate grasp of the technical knowledge that exists in the field associated with said invention at the time of filing and 2) skilled in said field would do, and if the person could have conceived of the claimed invention based on cited inventions (prior art).

Specifically, after finding the claimed invention and cited invention(s), a comparison of the claimed and cited invention(s) is made and their common points and differences in the matters that specify the inventions are clarified. Next, based on the content of the cited invention(s) (including well-known or commonly used art), the examiner considers whether or not the claimed invention incorporates the optimum composition, modified design, or simple aggregation of features of a cited invention(s); the examiner also considers whether or not the content of cited invention(s) may disclose a motivation to arrive at the claimed invention. Based on these results, a determination of the existence of absence of an inventive step in the claimed invention is made.

Unlike some other technologies, software technology is commonly used in multiple fields of industry. As such, parties in a particular field normally make attempts at combining methods and means that are already used in other fields to achieve desired goals; attempts at applying methods and means in other fields are also common. As a result, combining technologies used in one field and applying them to another field is usually considered to be within the scope of the ordinary creative activities of a person skilled in the art; therefore, in cases where there are no technical difficulties (technical hindrances) pertaining to the combination or application of existing technologies (e.g., there are no substantial technical effects), then no inventive step is recognized.

(2) A Person Skilled in the Art

A "person skilled in the art" with respect to software-related inventions is defined as one who:

- Possesses common technical knowledge in the applied field of the said software-related invention and common knowledge (including significant facts), soft, and common technical knowledge in the computer field (e.g., systemization technology);
- Is able to use ordinary technical means for research and development;
- Is able to exercise ordinary creative skills in design modification, etc.;
- Is entirely familiar with the technology that exists in the field associated with the claimed invention at the time of filing (i.e., technologies in the applied field of the claimed software-related invention and computer technology).

It should also be noted that the Japanese equivalent of "a person skilled in the art" (*togyosha*) can also apply to "a group of persons" (e.g., a team of software specialists) rather than a single individual. As an example, let us say that a team has been formed to develop a new accounting system. The team is composed of members who possess common technical knowledge and common knowledge in the field of accounting and members who possess common technical knowledge in the field of computer technology. In such a case, this team would be referred to as “a person skilled in the art” in Japan.

(3) Examples of Exercising Ordinary Creative Skills

As mentioned above, the determination of nonobviousness (inventive step) of a software-related invention involves consideration based on the perspective of a person skilled in the art—a person who is able to exercise ordinary creative skills in the field of technology in question. The following cases are recognized as examples of the exercise of ordinary skills; as such, an inventive step is not recognized.

(i) Application to other fields

If a cited “file search system” invention exists, and the common means (i.e., concrete search configuration) used by the file search system is applied to a medical information system in order to create a “medical information search system,” this activity constitutes the exercise of ordinary creative skills.

(ii) Addition of a commonly known, commonly used means or replacement by equivalent

If, in addition to a keyboard, a means for entering numerical codes by selecting items displayed on the screen using a mouse or by bar-code reader is added as a system input means, this activity constitutes the exercise of ordinary creative skills.

(iii) Use of software to handle functions otherwise performed by hardware

If a comparison circuit (hardware) used to compare code is replace by code-comparing software, this activity constitutes the exercise of ordinary creative skills.

(iv) Systematization of human transactions

There are cases of prior art that describe transactions performed by humans but fail to describe how to systemize them. Even in such cases, if such transactions are systemized and realized by a computer, the activity constitutes the exercise of ordinary creative skills if said transactions can be realized through ordinary system analysis techniques or system design techniques in daily work. For example, if the process of receiving orders via telephone or fax is switched to an online system, this activity constitutes the exercise of ordinary creative skills.

(v) Reproduction of a known phenomenon in virtual (computer) space

In a tennis game device, for example, configuring the game so that the tennis ball bounces faster on an asphalt court than on a clay court constitutes the exercise of ordinary creative skills.

(vi) Design modification based on known facts or customs

Expressing gratitude upon the closing of a business transaction is common sense; furthermore, the addition (to an electronic transaction device) of a means for displaying a message of gratitude falls into the category of 2) above. Therefore, adding the means to display a message of gratitude (e.g., “Thank you”) to an electronic transaction device that can display messages constitutes the exercise of ordinary creative skills.

## 7. Software-Related Patent Application Trends

This section primarily discusses Japanese software-related patent application trends in light of the creation and publication of examination and implementing guidelines.

### 7-1. Software-Related Patent Trends in Japan

Table 7-1 is based on reference materials created by the JPO for the second meeting of the Legislative Affair Subcommittee of the Intellectual Property Committee of the Industrial Structure Council. As the data indicates, Japanese software-related patents centered around calculator-type patents in the 1970s, microcomputer-type patents in the early ’80s, word-processor-type patents in the mid-’80s, and software-medium-type patents in 1996 and 1997. Currently, network-type patents are the most prevalent.

Table 7-1. Software-Related Patent Trends in Japan

| Period | Patent Type | Typical Examples | Remarks |
|---|---|---|---|
| Mid-1970s | Calculator-type patents<br>・Apparatus patents | Calculators, keyboards, logic circuits | Realized only by hardware (not programs) |
| Early 1980s | Microcomputer-type patents<br>・Apparatus, device patents (microcomputer control)<br>・Programs used for hardware control | Microcomputer-controlled electric rice cookers | Realization of temperature control by microcomputer circuits |
| Mid-1980s | Word-processor-type patents<br>・Apparatus patents (characterized by program functions)<br>・Programs not limited to hardware control | Word processors | Programs stored in word processor ROM used to realize Japanese character conversion |
| 1996–1997 | Software medium–type patents<br>・Media (e.g., CD-ROMs) patents (characterized by program functions)<br>・Programs not limited to hardware control | Japanese character conversion programs (CD-ROMs) | Programs recorded on floppy disks used to realize Japanese character conversion |
| Present | Network-type patents<br>・Patents for programs distributed via computer networks | Programs distributed via computer networks | Programs downloaded from a server via a network are used on personal computers |

## 7-2. Trends from the 1970s to the ’90s

The most important patent classification with respect to understanding software-related patent application trends in Japan is the G06G (FI) class: electric digital data processing. Table 7-2 below shows data for patent applications and publications that include the “Houhou (process/method)” and “Souchi (apparatus, machine, equipment, etc.)” in the title of the invention. The first year that such patents were published was 1973.

Table 7-2. Software-Related Patent Applications & Publications (FI＝G06F)

| Year | “Houhou” | | “Houhou” AND “Souchi” | |
|---|---|---|---|---|
| | Applications | Publications | Applications | Publications |
| 1973 | 51 | 7 | 11 | 1 |
| 1974 | 57 | 8 | 12 | 1 |
| 1975 | 58 | 26 | 9 | 3 |
| 1976 | 259 | 12 | 47 | 2 |
| 1977 | 343 | 11 | 53 | 0 |
| 1978 | 363 | 7 | 48 | 2 |
| 1979 | 454 | 19 | 68 | 1 |
| 1980 | 541 | 26 | 74 | 4 |
| 1981 | 815 | 49 | 93 | 7 |
| 1982 | 988 | 57 | 127 | 11 |
| 1983 | 1173 | 107 | 141 | 10 |
| 1984 | 1453 | 144 | 184 | 25 |
| 1985 | 1909 | 158 | 191 | 17 |
| 1986 | 2435 | 158 | 242 | 21 |
| 1987 | 2686 | 176 | 305 | 32 |
| 1988 | 3415 | 198 | 505 | 25 |
| 1989 | 4030 | 238 | 737 | 27 |
| 1990 | 4955 | 261 | 1057 | 36 |
| 1991 | 6233 | 322 | 1519 | 35 |
| 1992 | 6816 | 514 | 1609 | 68 |
| 1993 | 7561 | 673 | 1859 | 97 |
| 1994 | 7642 | 1049 | 2135 | 165 |
| 1995 | 8959 | 1621 | 2432 | 312 |

The following conclusions can be drawn from the data in Table 7-2.

(1) The significant increase in the number of software-related applications in 1976 can be attributed to the JPO publication of Examination Guidelines for Computer Software–Related Inventions – Part 1 in 1975.

(2) The significant increase in the number of software-related applications and publications between 1983 and 1984 can be attributed to the JPO publication of Implementing

Guidelines for Inventions Related to Microcomputer-Applied Technology in 1982.

(3) The significant increase in the number of patent publications after 1993 can be attributed to the JPO publication of Examination Guidelines for Inventions in Specific Fields in 1993.

Table 7-3. Examples of Published Software–Related Patents Circa 1973

| Publication No. | Title | Filing Date |
|---|---|---|
| Tokko Sho 48-6134 | Buffer memory control method and apparatus | Aug. 12, 1969 |
| Tokko Sho 48-6134 | Display method and apparatus | Sept. 25, 1969 |
| Tokko Sho 49-36156 | Recording an inspection method and apparatus | Feb. 20, 1970 |
| Tokko Sho 50-24816 | Array method and apparatus for encoding, detecting, and revising data | Oct. 23, 1970 |
| Tokko Sho 50-24819 | Apparatus for automatic display of logic elements and automatic change of status | Feb. 21, 1969 |
| Tokko Sho 50-28771 | Printing method and apparatus | Jun. 13, 1969 |
| Tokko Sho 51-10462 | Word storage system and apparatus used in a data-processing apparatus | Jul. 30, 1970 |
| Tokko Sho 51-26013 | Method and apparatus for X-ray image scanning | Dec. 19, 1972 |
| Tokko Sho 53-33876 | Method and apparatus for automatic symbol reading | Feb. 19, 1972 |
| Tokko Sho 54-2573 | Method and apparatus for editing printed material | Sept. 20, 1973 |
| Tokko Sho 55-12997 | Circuit apparatus and method for eliminating randomness | Jun. 5, 1972 |
| Tokko Sho 55-18394 | Data-processing method and apparatus | Apr. 24, 1975 |

## 7-3. Software-Related Patent Application Trends Since the 1997 Introduction of the Medium Patent

With the 1997 publication of Part VII (Implementing Guidelines for Inventions in Specific Fields), Chapter 1 (Computer Software–Related Inventions “) of the JPO’s Examination Guidelines for Patents and Utility Models, programs recorded on computer-readable storage media were deemed eligible for patent protection as “product” inventions; as a result, the term “program” came to be widely used in invention titles and scope of claim. As such, trends in software-related patent applications will be investigated based on the number of patent applications that include the term “program” in the invention title and/or the scope of claims.

### 7-3-1. Trend in the Number of Software-Related Patent Applications

Figure 7-1 below indicates the trend in the number of software-related patent applications from 1990 to 2006 based on the investigation method described above. The data is based on patents published in Japan (unexamined) that have the word "program" in the invention title and in the scope of claim. Based on this data, the following conclusions can be made.

(1) In both "Title" and "Claim" categories, the number of applications increased significantly in 1997 and 2000. It is possible to attribute the 1997 increase to the publication of implementing guidelines that allowed for "medium patents" and the 2000 increase to revised examination guidelines that recognized the patentability of programs as "products."

(2) The trend in the number of applications with the word "program" in the invention title mirrors that of applications with the word "program" in the scope of claim. Consequently, it is possible to acquire an understanding of software-related application trends by looking at data in either category.

(3) The number of applications in either category remained relatively constant from 1990 to 1996. This can be attributed to the fact that not all claims involved "programs" or "storage media." In particular, the fact that the number of patent applications including the word "program" in the scope of claim hovered around 5,000 during this period indicates that the word "program" was used as a modifying word of other products/processes in the description of the claim.

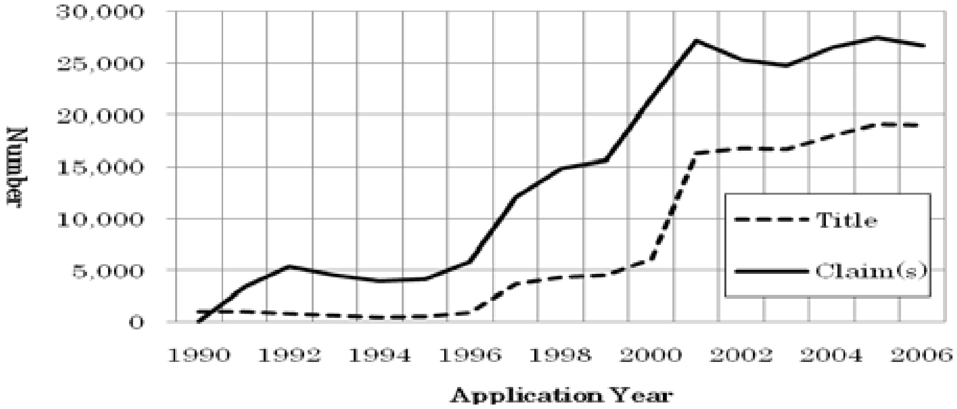

Figure 7-1. Number of Published Software-Related Patent Applications (unexamined) in Japan

7-3-2. Number of Software-Related Patent Applications in Specific Technical Fields

The data in Figure 7-2 below is based on patents published in Japan (unexamined) after the introduction of "medium patents" in 1997.

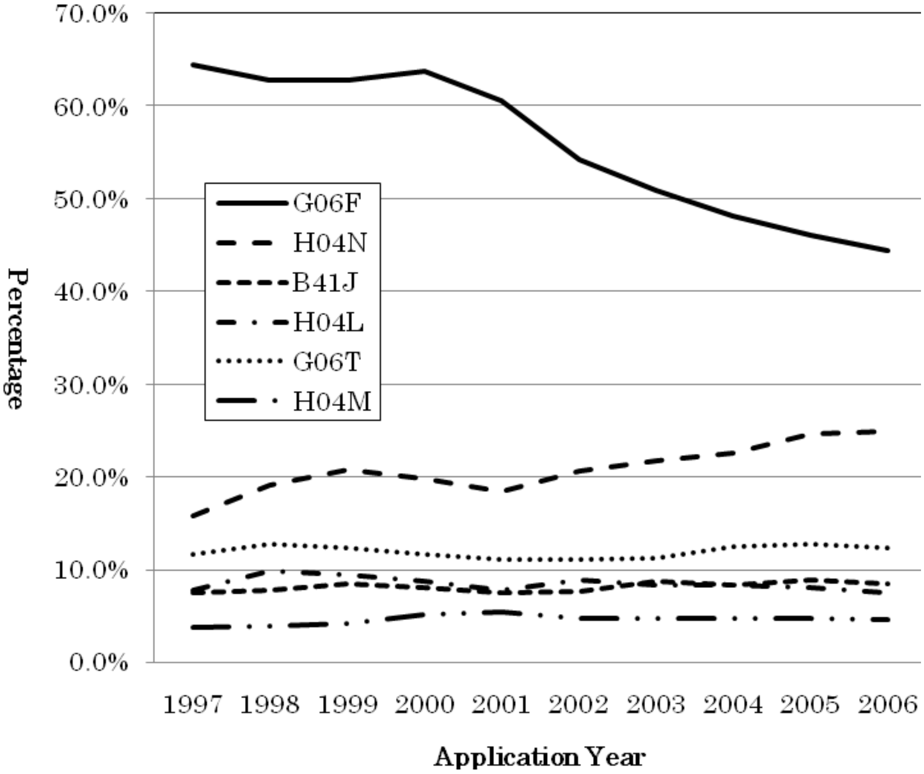

Figure 7-2. Percentage of Software-Related Patent Applications in Specific Technical Fields (FI Class)

Based on this data, the following conclusions can be made:

(1) The ratio of patent applications in the "electric digital data processing" (FI=G06F) category dropped sharply after 1999.

(2) The ratio of patent applications in the "pictorial communication" (FI=H04N) and other categories is increasing.

Figure 7-3 below is based on Japanese patents that were published (unexamined) in the first half of 2000, 2003, and 2005. These patents have been assigned to both the "G06F" (electric digital data processing) and other FI classifications.

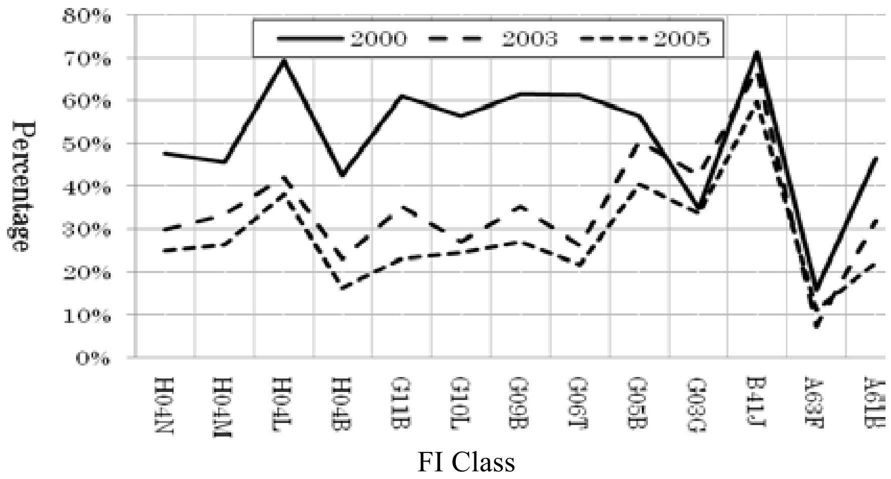

Figure 7-3. Percentage of Software-Related Patents Assigned to Both “G06F” and Other FI Classifications

As the figure above shows, in most FI classifications, the percentage of patents that were also assigned to the “G06F” (electric digital data processing) category declined sharply; up until 2000, most software-related patents were assigned to the G06F category. Based on this observation, one can say that software-related patents have become more diverse.

## 7-4. Recent Trends in Business Method-Related Inventions

### 7-4-1. Patent Application Trends Pertaining to Business Mehod-Related Inventions

According to survey data regularly released by the Japan Patent Office, the approximate number of applications for business method–related patents—which are a very important subcategory of software-related inventions—since 1999 is as follows:

1999: 4,100
2000: 19,600
2001: 19,000
2002: 13,000
2003: 10,000
2004: 9,000
2005: 8,000
2006: 7,000

Source: “Recent Trends in Business Method-Related Inventions” (http://www.jpo.go.jp/cgi/link.cgi?url=/tetuzuki/t_tokkyo/bijinesu/biz_pat.htm)

In the following section, patent application data is based on business method–related inventions.

As the data above shows, the number of business method–related patent applications has been on the decline since around 2001. In 2000, Japan experienced a business method–related patent boom once business methods came to be recognized as patentable subject matter. As a result of the boom, many companies and individuals who theretofore had comparatively little interest in patents were suddenly motivated to file applications, and this is the reason why the number of applications suddenly increased. However, the JPO subsequently published examination guidelines pertaining to business method–related patents, and this had a dampening affect on the boom, resulting in the subsequent decline in the number of applications.

Using a survey method slightly different from the method employed by the JPO mentioned above, the author of this guide conducted their own research (see Table 7-4 below). This data shows that the ratio of published Japanese patents (unexamined) belonging to individuals reached a peak between October 2001 and June 2002, and then began to decline. In terms of the number of applications filed by individuals, the peak occurred between April 2000 and December 2000. In 2000, the number of applications reached approximately 19,600; this data matches that released by the JPO.

Table 7-4. Number & Ratio of Applications Filed by Individuals

| Published | No. of Applications | Percentage of Total Applications |
|---|---:|---:|
| Jan-Mar 2000 | 18 | 1.9% |
| Apr-Jun 2000 | 27 | 2.6% |
| Jul-Sep 2000 | 32 | 2.5% |
| Oct-Dec 2000 | 59 | 4.4% |
| Jan-Mar 2001 | 91 | 5.0% |
| Apr-Jun 2001 | 128 | 5.4% |
| Jul-Sep 2001 | 223 | 7.9% |
| Oct-Dec 2001 | 638 | 11.3% |
| Jan-Mar 2002 | 827 | 11.2% |
| Apr-Jun 2002 | 722 | 10.0% |
| Jul-Sep 2002 | 622 | 8.7% |
| Oct-Dec 2002 | 507 | 7.3% |

The JPO data makes a distinction between "applications for business method–related inventions, where the primary feature is the business method–related invention itself" and "applications for business method–related inventions, where the primary feature is another technology." In the case of the former, the invention is assigned a primary FI classification, which specifies a business method–related invention. In the case of the latter, even though a primary FI classification is assigned, which specifies a business method–related invention, the invention is nevertheless considered to belong to another primary FI classification.

Of all the "applications for business method–related inventions, where the primary feature is the business method–related invention itself" that were filed since 2000, the numbers of applications in (1), (2), and (3) below indicate a downward trend; the rate of decline in (2) —electronic commerce—is particularly conspicuous.

(1) Business systems (i.e., computer systems adapted to industry-specific business tasks)
Examples: real estate management systems, medical information systems, etc.
(2) Electronic commerce
(3) Payment and settlement

The field of "electronic commerce" is characterized as one that involves new business models; more specifically, the field is characterized by ideas centered on the realization of business models that utilize Internet technology. In 2000, many individuals and companies who developed, and competed to file patent applications for, new business ideas involving the use of Internet technology encountered difficulty in acquiring patents because their inventions were not deemed to include an inventive step. Furthermore, it became increasingly difficult for individuals and companies to develop new ideas. These are considered to be contributing factors behind the sharp decline in the number of business method–related patent applications.

Using a survey method slightly different from the method employed by the JPO, the author of this guide conducted their own research (see Figure 7-4 below). In this figure, the upper graph (①) shows the trend in patent applications for business method–related inventions involving the use of Internet technology between November 1999 and December 2005. The lower graph (②) shows the trend in patent applications for business system–related inventions. Based on the data in both graphs, the following observations can be made:

- After exhibiting an upward trend, the number of patent applications for business system–related inventions (systematization, etc.) leveled out somewhat. (②)

- By contrast, the number of patent applications for business method–related patents dropped sharply after exhibiting an upward trend. (①)

- The reason for the overall decline in the number of patent applications for business method–related inventions can be attributed primarily to the decrease in the number of applications for business method–related inventions that use Internet technology.

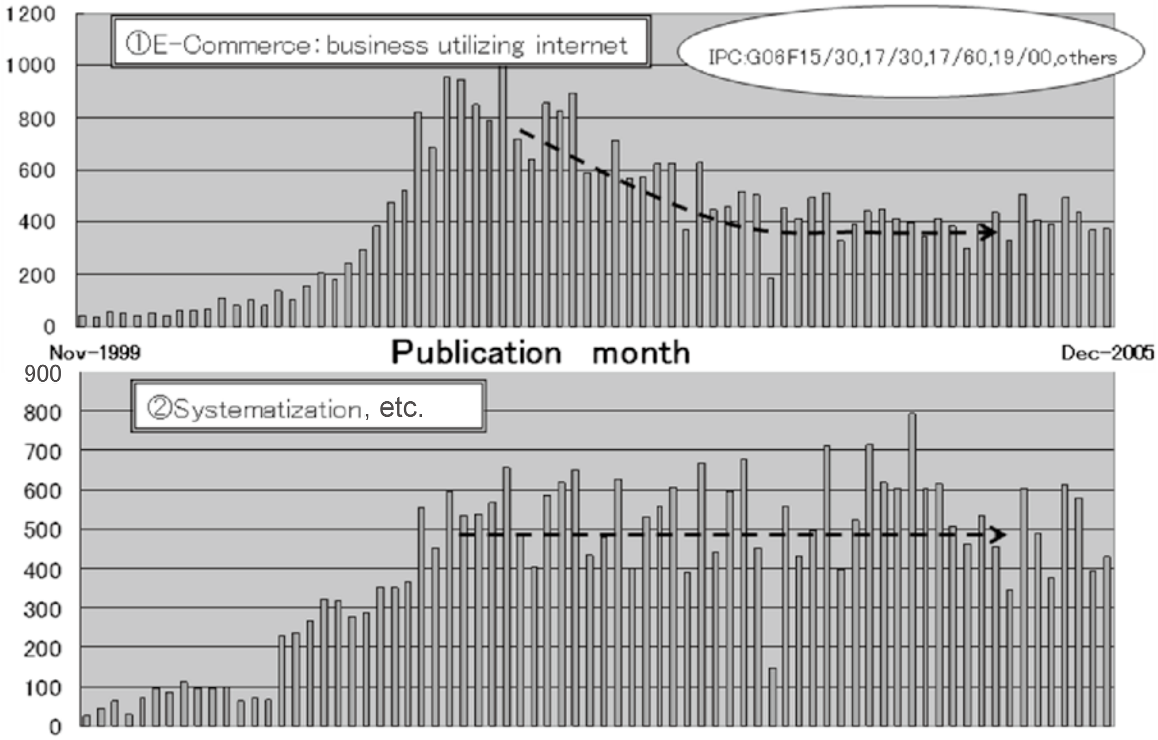

Figure 7-4. Monthly Number of Patent Applications for Business Method-Related Patents in Japan

7-4-2. Patentability of Business Method-Related Inventions

According to JPO data, the rate of decisions to grant patents for the aforementioned “applications for business method–related inventions, where the primary feature is the business method–related invention itself” has dropped significantly since 2000, hovering around 8% between 2003 and 2006. Compared to the average rate of 50% in all patent fields, “8%” is an extremely low value. At the same time, the rate of demands for an appeal against the examiner’s decisions of refusal (overall average of less than 20% since 2004) has declined since 2002, even though the rate of decisions to reject applications is high compared to overall examination results. Furthermore, with respect to “applications for business method–related inventions, where the primary feature is the business method–related invention itself," demands for an appeal against the examiner's decision of refusal are accepted is only 20% of the cases; this figure is extremely low compared to the overall average of 50%.

The JPO data above indicates that business method–related inventions are comparatively less patentable than others. This can be attributed to the following:

(1) Many companies and individuals, who traditionally expressed little interest in patents, created their patent specifications before acquiring an accurate understanding of examination guidelines and other matters.

(2) At the time, there were few case studies involving 1) decisions to reject patent applications and 2) decisions to accept appeal requests. This made it difficult for claimants (applicants) to acquire an accurate understanding of examination guidelines and other matters.

Although business method–related inventions are comparatively less-patentable than others at this time, this patentability problem will likely be solved in the future. In fact, the downward trend in the number of patent applications for business method–related inventions can be seen as indicating a decrease in the number of wasteful patent applications for inventions that would not have been deemed patentable.

Still, the business method–related patent boom at the turn of the century, which drove companies and individuals to file so many patent applications, can also be perceived in a positive light. First, it heightened public awareness of patents and, by extension, intellectual property. It also helped to dispel the traditional notion that business method–related inventions are only created in R&D environments, and expanded the definition of innovation to include nontechnological matters; for example, changes made to social systems are now considered to be innovations as a result. Finally, it is also thought to have played a vital role in boosting the public's desire to invent.

## 7-5. Examples of Intellectual Property High Court Rulings Pertaining to the Patentability of Software-Related Inventions

As discussed above, compared to other types of software-related inventions, both the rate of decisions to grant patents and the rate of decisions to accept demands for appeals are extremely low in the case of business method–related inventions. This same trend also applies to administrative lawsuits seeking to have the Intellectual Property High Court overturn examiners' decisions to reject patent applications. Table 7-5 below lists representative examples of Intellectual Property High Court rulings in recent cases. As shown, the Intellectual Property High Court upheld JPO examiners' decisions to reject patent applications in the overwhelming majority of cases. The most common reason given for upholding examiners' decisions was the absence of inventive step (no inventive step).

Table 7-5. Recent Examples of Intellectual Property High Court Rulings in Patent-Related Lawsuits

| No. | Case | Invention Title<br>(Publication No.) | Ruling |
|---|---|---|---|
| 1 | Heisei 19 (Gyo ke) 10194 | "GENERIC USER AUTHENTICATION FOR NETWORK COMPUTERS" (JP2000508153T) | **Upheld** (NO inventive step) |
| 2 | Heisei 19 (Gyo ke) 10226 | "METHOD FOR DETERMINING NUMBER OF RESERVED ARTICLES TO BE ORDERED" (JP2001142977 ) | **Upheld** (NO inventive step) |
| 3 | Heisei 18 (Gyo ke) 10239 | "METHOD TO GENERATE ABBREVIATED EXPRESSION OF BIT GROUP" (JP2000122538) | **Upheld** (NO Statutory Invention) |
| 4 | Heisei 18 (Gyo ke) 10564 | "ARTWORK SELLING SUPPORT SYSTEM" (JP2002203136) | **Upheld** (NO inventive step) |
| 5 | Heisei 18 (Gyo ke) 10511 | "METHOD FOR DISTRIBUTING DIGITAL CONTENTS, DISTRIBUTOR, REPRODUCTION DEVICE AND COMPUTER PROGRAM" (JP2003051797) | **Upheld** (violation of enablement requirements) |
| 6 | Heisei 18 (Gyo ke) 10173 | "INTERACTIVE DISPLAY SYSTEM" (JP6110608) | **Upheld** (NO inventive step) |
| 7 | Heisei 18 (Gyo ke) 10315 | "METHOD AND SYSTEM FOR DISTRIBUTING DIGITAL CONTENTS" (JP2001265937) | **Upheld** (NO inventive step) |
| 8 | Heisei 18 (Gyo ke) 10203 | "METHOD FOR PROVIDING A PLURALITY OF INFORMATION" (JP2002024461) | **Overturned** (inventive step) |
| 9 | Heisei 18 (Gyo ke) 10253 | "GUIDING METHOD FOR BRIDAL INFORMATION ON WEDDING HALL AND THE LIKE" (JP9269962) | **Upheld** (NO inventive step) |

# 8. Software-Related Patent Litigation Trends & Major Cases

## 8-1. Software-Related Patent Litigation Trends

Table 8-1 (Japan) and Table 8-2 (Overseas) show the number of “high-profile” cases involving patent litigation from 2003 to 2007 in the fields of “computing, electronic,” “commerce,” “systematization, etc.,” and “telecommunications”—all of which are closely linked to software-related patents. These “high-profile” cases are cases that the author of this guide have a general knowledge of based on print and online media coverage. The aforementioned fields are generally based on those specified in Table 3-2 above; however, “payment/settlement,” “finance/insurance,” “content delivery,” “online auctions,” and some other items executed by “distributed computer systems” have been included in “electronic commerce” below. Additionally, “data compression” and “GUI” have been included in “systematization, etc.” below.

Based on the information in Tables 8-1 and 8-2, the following points can be made:

(1) The number of patent litigation cases in Japan is extremely small compared to the number of overseas cases (primarily in the U.S.). Furthermore, the number of Japanese cases does not exhibit any particular trend.

(2) By contrast, the number of overseas patent litigation cases increased year-on-year, particularly in the telecommunications field.

(3) One may conclude, based on (1) and (2), that software-related patent litigation is rare in Japan because Japanese companies and individuals do not possess the kind of software-related patents that would invite litigation; in other words, they do not offer internationally accepted software products.

Table 8-1. High-Profile Japanese Software-Related Patent Litigation Cases

| Field | 2003 | 2004 | 2005 | 2006 | 2007 |
|---|---|---|---|---|---|
| Computing | 0 | 0 | 1 | 2 | 0 |
| Electronic commerce | 0 | 1 | 1 | 1 | 2 |
| Systematization, etc. | 2 | 4 | 8 | 0 | 2 |
| Telecommunications | 1 | 0 | 1 | 1 | 1 |
| Total | 3 | 5 | 11 | 4 | 5 |

Table 8-2. High-Profile Overseas Software-Related Patent Litigation Cases

| Field | 2003 | 2004 | 2005 | 2006 | 2007 |
|---|---|---|---|---|---|
| Computing | 6 | 16 | 8 | 17 | 22 |
| Electronic commerce | 13 | 15 | 13 | 20 | 12 |
| Systematization, etc. | 17 | 28 | 31 | 30 | 23 |
| Telecommunications | 13 | 14 | 30 | 30 | 56 |
| Total | 49 | 73 | 82 | 97 | 113 |

Table 8-3 shows the number of newly filed lawsuits including infringement warnings. These new litigation cases account for roughly half of the aforementioned "high-profile" cases, and the data exhibits the same trends indicated in Tables 8-1 and 8-2; in particular, the number of litigation cases in the field of telecommunications increase dramatically. One of the most notable cases in the telecommunications field, which involves mobile phone handsets, is that of Qualcomm vs. Nokia.

Table 8-3. New High-Profile, Software-Related Patent Litigation Cases

| Field | 2003 | 2004 | 2005 | 2006 | 2007 |
|---|---|---|---|---|---|
| Computing | 2 | 9 | 4 | 11 | 11 |
| Electronic commerce | 8 | 12 | 5 | 13 | 5 |
| Systematization, etc. | 4 | 11 | 14 | 6 | 11 |
| Telecommunications | 4 | 4 | 12 | 16 | 18 |
| Total | 18 | 36 | 35 | 46 | 45 |

## 8-2. Major Cases

### 8-2-1. Infringement Case Involving Justsystem's "Ichitaro" & "Hanako" Software

#### (1) History

In this high-profile case in Japan, Matsushita Electric Industrial Company sued Justsystem Corporation, claiming that the use of icons in Justsystem's "Ichitaro" word-processing software and "Hanako" graphics software infringed upon its patent rights. The lawsuit requested an injunction to prevent Justsystem from selling the two products and demanded that existing Ichitaro and Hanako inventory be disposed of. The following is a list of developments associate with the case.

<<From Filing to Registration>>

The patent in question was Japanese Patent Number 2803236, a patent for an information processor and information-processing method. Matsushita filed the patent application in October 1989 and the patent was registered in July 1998. The description of the scope of claim is as follows.

Claim 1

An information processor featuring:

- A display means for displaying on a screen a first icon, which executes a function for displaying descriptions of icon functions, and a second icon, which executes predetermined information processing functions;
- A specifying means for specifying the icons displayed on the screen by said display means;
- A control means for displaying the descriptions of the functions of the second icon on the screen according to the specification of the second icon subsequent to the specification of the first icon by said specifying means.

Claim 2

The information processor specified in Claim 1, wherein said control means executes a predetermined information-processing function of the second icon if the specification of the second icon by said specifying means does not immediately follow the specification of the first icon.

Claim 3

An information-processing method for controlling an apparatus that comprises a data input apparatus and data display apparatus. This method comprises the following steps:

- Displaying on a screen the first icon, which executes a function for displaying descriptions of functions, and a second icon, which executes predetermined information processing functions;
- Displaying on a screen the descriptions of the functions of the second icon in accordance with the specification of the second icon following the specification of the first icon.

As evidenced by the above claims, the Matsushita patent is an “apparatus” (i.e., product) and “process” patent.

<<Tokyo District Court Ruling>>

In this case, the points of contention were 1) whether or not the "Help Mode" and "Print" buttons displayed on personal computers in which Justsystem's products were installed could be considered "icons," 2) whether or not this constituted indirect infringement, and 3) whether or not clear grounds for invalidation existed. Matsushita's allegations regarding all three points were recognized, and instructed Justsystem to halt production of Ichitaro and Hanako and dispose of existing inventory.

<<Intellectual Property High Court Ruling>>

Justsystem rejected the ruling of the Tokyo District Court and subsequently filed an appeal with the Intellectual Property High Court. In addition to 1) and 2) above, two additional points of contention were added: 3) whether or not Matsushita's patent, and the exercise of the rights thereof, should be invalidated, and 4) whether or not additional allegations and evidence submitted by Justsystem should be dismissed as an unduly late offensive/defensive action. With respect to 3), the Court ruled that, based on examples of publicly known overseas technology submitted by Justsystem, Matsushita's patent did not meet the inventive step requirement and should therefore be invalidated. With respect to 4), the Court rejected Matsushita's argument that additional allegations and evidence submitted by Justsystem should be dismissed on the grounds of lateness. With respect to 1), the Court upheld the District Court's finding regarding the term "icon." However, with respect to 2), the Court disagreed with the District Court's finding regarding indirect infringement of a "process" invention patent.

In this case, the Intellectual Property High Court ruled against the Tokyo District Court, which recognized indirect infringement even in a case of a "process" invention. Instead, the High Court found that Justsystem was only manufacturing and selling products used in the manufacture of products (i.e., personal computers) rather than manufacturing or selling the products (i.e., computers) themselves. This is an important point that is discussed in some detail below.

<<Important Points in the Matsushita-Justsystem Case>>

While the Intellectual Property High Court recognized that "a computer on which Justsystem's products are installed" satisfies the constitute features of the invention described in Claim 3 above, which constitutes an invention of a process," it also found that Justsystem was not liable for indirect infringement under Patent Law Article 101(4) that stipulates that the “act of producing, assigning, etc. any product” constitutes patent infringement when the process described in the invention can be worked using the said product, because Justsystem was manufacturing and selling only Justsystem’s products used for manufacturing personal computers rather than manufacturing or selling said computers themselves.

Matsushita filed a patent application for this invention before the JPO recognized the patentability of “computer-readable storage media having a program recorded thereon” or “programs per se” as “product” inventions. This is thought to be the reason why indirect infringement was not recognized with respect to a “process” invention.

### 8-2-2. GIF Patent Controversy

Developed by CompuServe, the Graphics Interchange Format (GIF) is an image file format that employs the LZW (Lempel-Ziv-Welch) compression method (lossless data-compression algorithm). This became a high-profile controversy not only in the industry but also throughout society because, among other reasons, 1) a patent had already been granted for the LZW compression method and 2) Unisys dramatically changed its licensing policy.

<<Overview of GIF History>>

1985: U.S. patent granted for Unisys LZW technology.

1987: U.S. Internet service provider CompuServe (now owned by AOL) releases GIF as a recommended specification for exchanging images over its network.

1993: Unisys discovers that the GIF algorithm infringes on its LZW patent rights.

1994: CompuServe entered into a licensing agreement with Unisys. Unisys announced that it would “not require licensing, or fees to be paid, for non-commercial, non-profit GIF-based applications, including those for use on on-line services.” Consequently, GIF rapidly became a standard image format used in free Internet browsers.

1996: Unisys changed its policy regarding LZW patent licensing and began charging licensing fees on freeware and other types of software as well.

2003: The United States LZW patent expired on June 20th.

2004: The Japanese LZW patent expired on June 20th.

<<Japanese LZW Patent: Scope of Claim>>
Invention title: Apparatus and method for compressing digital signal stream
Application filed: June 20, 1984
Application no.: Tokkyo Shutsugan Sho 59-123473
Publication date: June 22, 1985
Publication No.: Tokkyo Kokai Sho 60-116228
Kokoku No.: Tokkyo Kokoku Hei 05-068893

<<Scope of claim>>

The digital signal stream compression method is characterized by the following:

"A data compressor compresses an input stream of data character signals by storing in a string table strings of data character signals encountered in the input stream. The compressor searches the input stream to determine the longest match to a stored string. Each stored string comprises a prefix string and an extension character where the extension character is the last character in the string and the prefix string comprises all but the extension character. Each string has a code signal associated therewith and a string is stored in the string table by, at least implicitly, storing the code signal for the string, the code signal for the string prefix and the extension character. When the longest match between the input data character stream and the stored strings is determined, the code signal for the longest match is transmitted as the compressed code signal for the encountered string of characters and an extension string is stored in the string table. The prefix of the extended string is the longest match and the extension character of the extended string is the next input data character signal following the longest match. Searching through the string table and entering extended strings therein is effected by a limited search hashing procedure. Decompression is effected by a decompressor that receives the compressed code signals and generates a string table similar to that constructed by the compressor to effect lookup of received code signals so as to recover the data character signals comprising a stored string. The decompressor string table is updated by storing a string with a prefix in accordance with a prior received code signal and an extension character in accordance with the first character of the currently recovered string."

<<Main Points>>

(1) At the time CompuServe released the GIF specification, it was not aware that a patent for the LZW compression method existed and therefore believed it was free for public use. This

wrong assumption eventually led to the controversy that developed. The first GIF specification, which was released in 1987 as GIF87a, was based on an article written by Terry A. Welch entitled “A Technique for High Performance Data Compression” (IEEE Computer, Vol. 17, No. 6 (June 1984)). Therefore, a simple patent search conducted at the time would have revealed that an LZW patent existed, and thus the relevant parties could have been notified that the GIF specification involved a patented invention. The controversy could likely have been avoided if this had been done.

In short, a company or individual who wishes to establish a standard specification should first conduct a patent search to determine whether or not the specification possibly infringes upon an existing patent. Based on the results, the company or individual can then decide to abandon the specification or proceed in a way to avoid controversy, such as by having the patent holder specify licensing conditions.

(2) When Unisys and CompuServe entered into a licensing agreement in 1994, Unisys announced that it would not require licensing, or fees to be paid, for non-commercial, non-profit GIF-based applications. As a result, freeware developers continued to develop software that incorporated the GIF specification. However, in and after 1996, Unisys adopted a hard-line stance against such developers and changed its policy; consequently, freeware that incorporated the GIF specification quickly vanished.

In this case, regardless of the intension of Unisys, big controversy was caused by the fact that the software developer was not aware of the existence of a patented process in relation to the software to be developed. The developer, however, continued development work with the belief that it could use the process without paying any fees. The developer was required to pay a huge amount of licensing fees in the end.

### 8-2-3. MP3 Patent Infringement Case

This was a case to determine whether or not the use, in a foreign country, of software programs that violate U.S. patents constitutes infringement. Developments in the case were closely followed by people in the software industry because, if the court had ruled that it did indeed constitute patent infringement, it would have led to numerous other infringement lawsuits and resulted in enormous damage awards and licensing fees in light of the fact that many software products developed in the U.S. are used overseas.

<<Case History>>

June 2006: AT&T sued Microsoft for patent infringement over speech-decoding technology (G.723.1 codec) used in its “NetMeeting 2.0” videoconferencing software.

February 2004: U.S. District Court rules in favor of AT&T.

March 2004: AT&T and Microsoft reached a settlement (terms are unclear). However, the two decided to ask the Court of Appeals to determine whether or not Microsoft Windows software sold outside of the U.S. also infringed upon AT&T’s patent.

July 2005: The Court of Appeals upheld the decision of the lower court and found Microsoft liable for infringement.

October 2006: The U.S. Supreme Court accepted a request from Microsoft to appeal the Court of Appeals’ ruling, which found that AT&T could claim licensing fees from Microsoft with regard to the manufacture and sale of software outside of the U.S.

April 2007: The Supreme Court found that Microsoft was not liable for patent infringement because the software that it exports on a master disk or by electronic transmission is not directly installed on PCs manufactured outside the U.S.; rather, it is the “copies” of the software that are installed. As such, it is not possible to say that the software is “supplied from the U.S.” Consequently, the Court overruled the previous court’s decision.

<<Main Points>>

Added to the U.S. Code in 1984, Section 271(f) states:

> “Whoever without authority supplies or causes to be supplied in or from the United States all or a substantial portion of the components of a patented invention, where such components are uncombined in whole or in part, in such a manner as to actively induce the combination of such components outside of the United States in a manner that would infringe the patent if such combination occurred within the United States, shall be liable as an infringer.”

AT&T claimed that cases in which software is exported on a master disk and installed in personal computers manufactured overseas constitute patent infringement under Section 271(f), and both district and appellate courts ruled in favor of AT&T, finding Microsoft liable for infringement. The U.S. Supreme Court, however, reversed those decisions by finding that only "copies" of the software were installed, and therefore the case could not be construed as "supplying" software from the U.S.

In this case, the Supreme Court found that Section 271(f) was created with hardware in mind and was not applicable to software. However, as indicated throughout this guide, a computer does not function based solely on a combination of tangible components; software code is essential. Furthermore, master disks can be copied anywhere in the world. Consequently, the court's interpretation that Microsoft did not "supply from the U.S." based on the fact that only "copies" of the software were installed does not seem to be based on reality. The fact that the Supreme Court justices have characterized this problem as a legislative issue suggests that, at the time Section 271(f) was established, no one predicted such a case would arise.

### 8-2-4. Drug Discovery/Design Software Patent Infringement Case

The owner, A and licensee (Institute of Medicinal Molecular Design) of the patent in question filed a lawsuit against Sumisho Electronics, claiming that a complex search process used by a module called "FlexX," which was used in programs imported to and sold in Japan on CD-ROMs, fell within the technical scope of the patent; the plaintiffs also claimed that the CD-ROMs constituted indirect infringement and requested an injunction on the sale thereof.

The Japanese Supreme Court found 1) that the patent holder, who had granted an exclusive license to the licensee, had the right to request an injunction under Article 100 ("Right to seek an injunction") of the Patent Law, 2) that the "process" in question (the "complex search process") applied to the technical scope of the patent, and 3) that, even though the software stored on the CD-ROMs incorporated various other tools in addition to the FlexX module, the CD-ROM in question was considered to be solely for the use of implementing the FlexX module, which falls under the technical scope of the patented invention in question. This is because there was not enough evidence to prove that, without using the FlexX module, there are any other economic, commercial, or practical uses for the CD-ROMs under normal social conventions. Based on these findings, the Supreme Court reversed the ruling of the District Court, which dismissed the plaintiffs' claims, and granted the injunction.

<<Case History>>

2001: A (patent holder) and the Institute of Medicinal Molecular Design (exclusive licensee) filed a lawsuit with the Tokyo District Court, claiming that the FlexX module used in its “SYBYL” molecular modeling software, which is imported to and sold in Japan, infringed upon the lawsuit in question.

2003: The Tokyo District Court dismissed the case, finding that the technical scope of the patent did not apply. The plaintiffs subsequently filed an appeal with the Tokyo High Court.

2004: The Tokyo High Court found that the technical scope of the patent applied to the complex search process in question used by the FlexX module, and that, even though the software stored on the CD-ROMs incorporated various other tools in addition to the FlexX module, the CD-ROMs incorporated various other tools in addition to the FlexX module, the CD-ROM in question was considered to be solely for the use of implementing the FlexX module, which falls under the technical scope of the patented invention in question, and granted the injunction. Sumisho Electronics subsequently appealed to the Supreme Court.

2005: The Supreme Court upheld the decision of the High Court and dismissed the appeal.

<<Scope of Claim of Patent Involved>>

Invention title: Method of searching the structure of stable biopolymer-ligand molecule composite
Application filed: March 26, 1993
Application no.: Tokugan Hei 5-517287
Publication date: October 14, 1993
Registration no.: Patent No. 2621842
Registration date: April 4, 1997

<<Abstract>>

“A method of searching the structure of a stable composite composed of a biopolymer and ligand molecules, the method of which comprises: (1) the first step of covering all modes of hydrogen bonding between a biopolymer and ligand molecules by covering all of the possible combinations of matching between dummy atoms positioned at the hydrogen-bonding heteroatoms of the hydrogen-bonding functional groups of the biopolymer and the hydrogen-bonding heteroatoms of the ligand molecules; (2) the second step of estimating the modes of hydrogen bonding between the biopolymers and the ligand molecules and the conformations of the hydrogen-bonding portions of the ligand molecules at the same time by comparing the distance between the dummy atoms with that between the hydrogen-bonding heteroatoms; and (3) the third step of finding the structure of a biopolymer-ligand molecule composite by substituting the coordinates of all the atoms of the ligand molecules on the basis of the relation of matching between the hydrogen-bonding heteroatoms of the ligand molecules and the dummy atoms for each of the modes of hydrogen bonding and the conformations estimated in the second step into the coordinate system of the biopolymer. This method permits the structure of a stable biopolymer-ligand molecule composite to be searched efficiently and accurately in a short time."

<<Main Points>>

(1) It is important to accurately describe the invention in the patent specification so that the courts can make proper rulings.

(2) Even though other software were stored on the CD-ROMs in addition to the FlexX software, the court deemed that there was not enough evidence to prove that, without using the FlexX module, there are any other economic, commercial, or practical uses for the CD-ROMs under normal social conventions; therefore, the CD-ROM was considered to be solely for the use of implementing the FlexX module, which infringed upon the plaintiffs' patent. This was considered to be a rational decision because multiple software products (modules) are recorded on a single medium since the multiple software products can function in concert.

(3) The High Court's interpretation in this case, which recognized the patent holder's right to request an injunction and prevent infringement under Patent Law Article 100, was upheld by the Supreme Court.

## 8-3. Indirect Infringement of Software-Related Patents

Over the years, systems for protecting programs per se and media on which programs are stored as "product" inventions have been implemented. The 1997 implementing guidelines specify that media on which programs are recorded are patentable as "medium" inventions (can be included in the claim); the 2000 revised examination guidelines specify that programs can be patentable as "product" inventions; the 2002 revision to the Patent Law specifies that programs are patentable as a “product” inventions.

When programs per se are protected as “product” inventions, the unauthorized distribution of such programs via a network or computer-readable storage medium constitutes direct infringement of patent rights. Consequently, the exercise of patent rights in such cases is just as easy as exercising hardware patent rights. However, software patents before 1997, when programs were still not deemed patentable as “product” inventions, and even software patents after 1996 that did not specify a “product” invention in the claim were deemed to be “apparatus” (i.e., “product” or “process”) patents, and patent rights had to be exercised accordingly. In such infringement cases, programs sold or distributed could not be considered as “directly” infringing upon “apparatus” or “process” patents; therefore, it was necessary to enable patent holders to file lawsuits based on “indirect infringement.” Consequently, the Patent Law (specifically Article 101) was revised in 2003 to recognize indirect infringement in a broader range of cases.

In the “Ichitaro/Hanako” case described previously, the Intellectual Property High Court ruled that Matsushita’s patent did not meet the inventive step requirement and should therefore be invalidated based on examples of common overseas technology. In its decision, the Court recognized that the manufacture and sale of a personal computer (referred to as “information-processing device” in the claim) on which Justsystem’s Ichitaro and Hanako software were installed could be considered as indirect infringement under Patent Law Article 101(2). At the same time, however, with respect to a “process” invention, the court found that the manufacture and sale of Justsystem’s own products used for manufacturing personal computers, rather than the manufacture or sale of the computers per se, did not constitute indirect infringement under Article 101(4). On the other hand, in the Sumisho drug discovery software case described above, the Tokyo High Court found that the sale of a medium on which the FlexX module was stored constituted indirect infringement upon the patent concerning the “Method of searching the structure of stable biopolymer-ligand molecule composite.”

The Intellectual Property High Court decision on the Ichitaro/Hanako case was handed down in October 2005, and the Tokyo High Court decision on the drug discovery method case was handed down in February 2004. Based on more recent Intellectual Property High Court rulings, the author of this guide feels that indirect infringement should be recognized in more cases. Up until now, the court’s rulings seem similar to the ruling that the U.S. Supreme Court made in the MP3 patent infringement case.

Patent Law Article 101 (Acts Deemed to Constitute Infringement)

The following acts shall be deemed to constitute infringement of a patent right or exclusive license:

1. Where a patent has been granted for an invention of a product, acts of manufacture, sale, etc. or import, or offering for sale, etc. any product to be used exclusively for the manufacture of said product as a business;

2. Where a patent has been granted for an invention of a product, acts of manufacture, sale, etc., importing or offering for sale, etc. any product (excluding those widely distributed within Japan) to be used for the manufacture of said product and indispensable for the resolution of a problem by said invention as a business, knowing that the said invention is a patented invention and that the said product is used for the implementation of the invention;

3. Where a patent has been granted for an invention of a process, acts of manufacture, sale, etc., importing or offering for sale, etc. any product to be used exclusively for the use of the said process as a business; and

4. Where a patent has been granted for an invention of a process, acts of manufacture, sale, etc., importing or offering for sale, etc. any product (excluding those widely distributed within Japan) to be used for the use of said process and indispensable for the resolution of a problem by said invention, knowing that said invention is a patented invention and said product is used for the implementation of the invention as a business.

## 9. New Trends in Software-Related Patents

### 9-1. Open Source Software & Software-Related Patents

### 9-1-1. Open Source Software

#### (1) Open Source History

The origin of “open source” can be traced back to Richard M. Stallman’s “free software movement.” (1983) Stallman advocated the development of free software that could be used by anyone and the creation of a world in which non-free software was not necessarily required. His movement resulted in the creation of such well-known software products as the “Sendmail” mail transfer agent and the “BIND” DNS server. Once use of the Internet became common in the 1990s, a more cooperative, organizational approach to free software development was rapidly adopted. One of the results of this trend was the Linux operating system that was developed by a team led by Linus Torvalds. Rather than following the conventional “Cathedral” model, where code is developed by a small group of elite programmers, the Linux OS was developed under the “Bazaar” model, in which anyone is allowed to make contributions, albeit in a cooperative manner.

The success of the Linux OS underscored the effectiveness of the Bazaar model and drew attention to this new style of software development. However, because the “free software” concept regards software as the collective property of humanity and is therefore inherently incompatible with business principles, the Open Source Initiative (OSI) was established to carry on the concept of free software while promoting the open source movement, which is centered on the Bazaar model of software development.

(2) The Meaning of “Open Source”

As described above, the most crucial part of “open source” is the realization of the Bazaar software development model; as such, licensing conditions for handover and distribution are important. A software product qualifies as open source software only when the licensing conditions associated therewith match open source licensing conditions. The “Open Source Definition,” which comprises ten criteria, is used by the Open Source Initiative to determine whether or not a software license can be considered “open source.” Of those ten criteria, the following are the most important:

- Free distribution (and reproduction) is allowed;
- Acquisition of source code (which is needed to modify software) is allowed;
- Modifications and creations of derivative works are allowed.

If these rights are guaranteed, then any individual can freely develop their own software based on a single program. As a result, these criteria have promoted community-based software development.

Various types of licenses, beginning with GPL (GNU General Public License), have been recognized as open source licenses. When exploiting open source software, various important conditions should be studied. They include those not mentioned above, e.g., whether or not the software added to the open source software satisfies the open source software licensing conditions.

### (3) Treatment of Patent Rights with Respect to Open Source Licenses

An open source license is basically a copyright license agreement. By clearly specifying the treatment of copyrights, the agreement attempts to establish an environment in which software can be freely modified or distributed. Such agreements, however, do not specify treatment of patent rights. Section 7, Paragraph 1 of the GPL (Version 2) stipulates:

> "If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement, or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License."

Then, it adds:

> "If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program."

In short, GPL specifies only passive defensive measures for the software covered by a license under the GPL against external attack based on the patent rights.

Furthermore, open source software code is inevitably disclosed to the public at some point, and this makes it possible for any third party to easily examine any software product that incorporates the code; consequently, this makes it easier for third parties to exercise their patent rights against open source software developers in some cases.

### (4) Open Source Software Market

According to research conducted by the various organizations below, the open source software market is rapidly expanding.

① A Gartner (U.S.) survey of 274 companies in North America, Europe, and the

Asian-Pacific region conducted between May and June 2008 found that 85% of the respondents had already installed open source software, and that the remaining 15% planned to install open source software within the following twelve months.
Source: http://www.gartner.com/it/page.jsp?id=801412

② A Forrester Consulting (U.S.) survey of 132 major European corporations that had already installed open source software, which was conducted in October 2008, found that 45% of the respondents used open source software in applications, services, and/or products. Among the various reasons given for installing open source software, "to reduce costs" was the most common (56%), followed by "to avoid relying on a single vendor" (45%); "because open source software is flexible" and "because open source software is innovative" were also popular responses.
Source: http://itpro.nikkeibp.co.jp/article/Research/20081202/320540/

③ IDC (U.S.) in May 2007 released the results of a survey on the global open source software market. The survey found that 2006 sales totaled USD 1.8 billion and that the market was expanding at an average annual rate of 26%. IDC predicted sales of USD 5.8 billion in 2011.
Source: http://itpro.nikkeibp.co.jp/article/Research/20070601/273284/

### 9-1-2. Recent Trends in Open Source Software & Software-Related Patents

As discussed in Section 9-1 above, as the open source software market expanded, it became a big concern for vendors and users to patent their open source software–related inventions. The issue of intellectual property rights with respect to open source software first arose in March 2003, when The SCO Group (U.S.) brought a civil lawsuit against IBM for allegedly "devaluing" its version of the UNIX operating system by contributing SCO's intellectual property to the codebase of its own Linux operating system, which is similar to UNIX. In response, SCO's warning that users of Linux may be found liable for infringement by using the Linux system without a license from SCO, Red Hat, Sun Microsystems, Hewlett-Packard, and other Linux OS distributors revealed measures for protecting their respective customers from liability. IBM subsequently filed a countersuit against SCO, claiming that the company's UnixWare, Open Server, SCO Manager, and Reliant HA software violated four of its own patents.

This case highlighted the issue of software-related patents with respect to open source software. Subsequent developments in this area are shown in Table 9-1 below.

Table 9-1. Developments Regarding Software-Related Patent Issues

| Month/Year | Overview of Issue | Source |
|---|---|---|
| January 2005 | Sun Microsystems announced plans to make its "Solaris 10" operating system available for free and release more than 1,600 patents associated with the OS | http://www.sun.com/smi/Press/sunflash/2005-01/sunflash.20050125.2.xml |
| November 2005 | IBM, Sony, Novell, Philips, and Red Hat jointly established the Open Invention Network, a company that acquires software-related patents and provides them royalty-free | http://www.openinventionnetwork.com/ |
| May 2007 | Microsoft alleged that free and open source software violated more than 230 of its patents. | http://japan.cnet.com/news/biz/story/0,2000056020,20348704,00.htm |
| May 2007 | Red Hat claimed that software-related patents actually hinder innovation | http://www.itmedia.co.jp/enterprise/articles/0511/10/news063.html |
| December 2008 | The Open Invention Network unveiled "Linux Defenders" to prevent relevant patents from being acquired by other companies by disclosing the developed technology as prior art | http://www.openinventionnetwork.com/press_release12_09_08.php |

As the developments shown above indicate, a greater focus is being placed on issues concerning open source software and software-related patents, and cooperation between companies that advocate open source (the open source faction) and companies that prioritize their own intellectual property (the proprietary faction) is becoming more common. Table 9-2 below lists partnerships that Microsoft—a representative example of a "proprietary faction" company—has entered into with open source companies.

Table 9-2. Microsoft Partnerships with Open Source Companies

| Month/Year | Description of Partnership | Source |
|---|---|---|
| September 2005 | Microsoft and Jboss announced that they had entered into an agreement whereby they would improve the interoperability of their server software | http://japan.internet.com/busnews/20050928/12.html |
| February 2006 | Microsoft announced plans to partner with SugarCRM to improve compatibility between Windows Server and the latter's open source software | http://www.itmedia.co.jp/enterprise/articles/0602/15/news020.html |
| September 2006 | Microsoft announced its "Open Specification Promise," pledging not to assert legal rights over certain Microsoft Web service patents | http://itpro.nikkeibp.co.jp/article/USNEWS/20060914/248035/ |
| November 2006 | Microsoft and Novell entered into an "historic" partnership under which they would collaborate on Windows and Linux interoperability | http://headlines.yahoo.co.jp/hl?a=20061122-00000095-myc-sci |
| June 2007 | Microsoft and Linux distributor Xandros entered into a technology and patent collaboration agreement | http://itpro.nikkeibp.co.jp/article/NEWS/20070605/273500/ |
| June 2007 | Microsoft and Linspire entered into an agreement to facilitate interoperability between Windows and Linux | http://www.itmedia.co.jp/enterprise/articles/0706/14/news087.html |
| October 2007 | Microsoft and Turbolinux extended a broad collaboration agreement | http://www.itmedia.co.jp/news/articles/0710/23/news048.html |
| November 2007 | Kyocera Mita and Microsoft entered into a broad cross-licensing patent agreement covering Linux technologies | http://japan.cnet.com/news/ent/story/0,2000056022,20361063,00.htm |
| February 2008 | Microsoft announced that it would fully disclose the code used in some of its software, including the Vista OS | http://itpro.nikkeibp.co.jp/article/COLUMN/20080314/296298/ |

As the developments in Table 9-2 indicate, one of the main reasons behind the remarkable increase in collaboration between open source and proprietary factions is the fact that ensuring interoperability between software products has become crucial. For example, say that Company A has a computer system that utilizes a certain operating system; if the company builds a second system which utilizes a different OS that is incompatible with the existing OS, the company may encounter serious efficiency-related problems. Therefore, when building the new system, the company will be strongly inclined to select software that is compatible with the existing system.

Increasingly, software providers are being forced to take this interoperability issue into account in order to meet the needs of their customers. Thus, collaboration between open source and proprietary faction companies, in addition to the public disclosure of system interface specifications, is becoming more common. This is one of the primary reasons behind the rapid expansion of the open source software market.

## 9-2. Legal System Reform Efforts

As we have learned, various patent examination guidelines for software-related inventions have been implemented and upgraded in Japan, the U.S., and Europe over the years. We learned that a greater focus has been placed on issues involving open source software and software-related patents. Furthermore, we learned that the number of cases involving patent litigation in the fields of computing, systematization (business systems), and telecommunications, which are closely related to software-related patents, is on the rise.

Based on these trends, the following section will take a look at the current status of efforts to reform legal systems with respect to patent protection in the U.S., Europe, and Japan.

### 9-2-1. U.S. Efforts

In October 2002, the Director of the United States Patent and Trademark Office (USPTO) acknowledged that many “problematic” business patents had been wrongfully awarded in the past. After this admission was made, the nation became engaged in a vigorous debate on the U.S. patent system, and more and more people called for the reform thereof. Some of the primary issues in the debate were 1) the enormity of the settlements involved in resolving patent litigation and 2) the worsening crisis involving patent trolls who, instead of manufacturing or selling anything on their own, simply buy up others’ patents and attempt to enforce them for the purpose of receiving compensation.

Table 9-3 below lists a number of software-related patent infringement cases in which huge settlements were reached. The enormous amounts of money involved in these settlements indicate the magnitude of the software market. For example, the case of Eolas vs. Microsoft (below) involved the Internet Explorer browser, which is used in the majority of personal computers that are sold, and the case of NTP vs. RIM (below) involved e-mail services that are used in many mobile phones; this is why settlements in cases involving products like these—products for which a huge market exists—are so large. Of the cases listed in Table 9-3, Eolas vs. Microsoft and NTP vs. RIM are the two recognized as involving lawsuits filed by patent trolls. Because the amount of the settlement in the former case was not publicly disclosed, the figure shown here is the amount of compensation awarded by the U.S. district court. In the case of Alcatel-Lucent vs. Microsoft, the figure shown is the amount of compensation awarded by the U.S. district court before its decision was overturned.

Table 9-3. Major Settlements & Compensation in Software-Related Patent Infringement Cases

| Year | Field | Plaintiff | Defendant | Settlement | Payment |
|---|---|---|---|---|---|
| 2003 | Patent for a browser plug-in | Eolas | Microsoft | ○ | ($520M) |
| 2004 | Computing technologies | Sun | Microsoft | ○ | $900M |
| 2004 | DRM and trusted computing technologies | InterTrust | Microsoft | ○ | $440M |
| 2004 | Pay-for-performance search technologies | Yahoo! | Google | ○ | $280M |
| 2005 | Information storage and management | EMC<br>(HP) | HP<br>(EMC) | ○ | $325M |
| 2006 | BlackBerry mobile e-mail service | NTP | RIM | ○ | $612.5M |
| 2007 | MP3-encoding technology | Alcatel-Lucent | Microsoft | | ($1520M) |

In April 2005, the United States House Judiciary Subcommittee on Intellectual Property announced a draft for patent law revisions that was subsequently submitted to Congress in June of the same year as the “Patent Reform Act of 2005.” The Act proposed a switch from “first to invent” to “first to file, expanded use of post-issuance reexamination and opposition proceedings, and other reforms; it also called for adjustment of compensation amounts.

Because the Act was never enacted, a very similar one – the Patent Reform Act of 2007 – was introduced in April 2007; although the House version of the bill passed, the Senate bill was not voted on and the bill was abandoned. This would seem to indicate that, although there are many opposing opinions regarding the reforms included in the Act, a consensus has formed within the U.S. Congress regarding the necessity of patent reform.

### 9-2-2. European Efforts

As discussed in Section 5-3 above, the European Parliamentin July 2005 overwhelmingly rejected the Computer-Implemented Inventions (CII) Directive, which removed computer programs from the list of non-inventions in EPC Article 52(2). After the initial version was proposed in 2002, the legislation was repeatedly revised to reflect the opinions of both those who agreed with it and opposed it. This section is devoted to discussing the gist of those opinions.

The group of those who agreed with the CII Directive was mostly composed of major IT vendors and other large corporations who believed that the protection of software-related inventions was vital to European innovation and to acquiring a competitive advantage. The group that opposed the legislation was composed of volunteers and companies that were open source advocates; they believed that software-related patents actually hinder software development and are thus unnecessary. As covered in Section 9-1-1 above, an open source license is basically a copyright license agreement that aims to realize the open source philosophy: to promote the evolution of software by promoting community-based software development. As such, the existence of software-related patents is incompatible with the open source philosophy, and this is why such strong opposition to the CII Directive emerged.

Although there are no plans to submit a further revised draft in the future, the European Patent Office (EPO) announced in October 2008 that it would refer questions regarding the patentability of computer programs to the Enlarged Board of Appeal (EBA)—the supreme ruling body of the European Patent Convention (EPC); this was done in an effort to standardize the treatment of computer program patentability under the EPC. It indicates that, because the CII Directive (which would have removed computer programs from the list of non-inventions in EPC Article 52(2)) was rejected, the EBA is now in charge of settling all issues pertaining to non-invention stipulations.

### 9-2-3. Japanese Efforts

As discussed in Section 5-1 above, the “Computer Software–Related Invention Examination Guidelines” that the Japan Patent Office (JPO) released in December 2000 included clear guidelines regarding the utilization of laws of nature, and the Patent Law revisions made in 2002 stipulated that computer programs could be patented as “product” inventions. As a result of these guidelines and revisions, the protection of software-related inventions in Japan was strengthened.

In 2005, the Justsystem icon infringement case (see Section 8-2-1) touched off a discussion in Japan about the relationship between software patents and innovation and what should be done in cases where such patents are found to hinder innovation. Subsequently, the Ministry of Economy, Trade, and Industry （METI） published “Rules Concerning Software-Related Intellectual Property Rights.” The history leading up to the release of these standards as well as recent trends is discussed below, with a focus on the relationship between software-related patents and innovation.

The emergence of open source software and the heightened awareness of the importance of software interoperability drove discussions on how to strike a balance between the protection and utilization of software-related intellectual property rights. As a result of those discussions, in June 2005, the METI formed the “Study Group on the Legal Protection of Software and Promotion of Innovation” to more closely examine the treatment of software-related intellectual property rights from the perspective of promoting innovation. In October of the same year, the Study Group released an interim report that summarized the issues at hand. Based on an analysis and organization of software characteristics, the Study Group proposed the following for the purpose of eliminating any hindering effects that software patent protection might have on innovation.

[Main Points of the Interim Report]

(1) A multilayered structure:
As seen in operating systems, middleware, and application software, software products have a multi-layered structure, in which the functions of software at upper levels are performed based on those at lower levels.

(2) Communication structure
A newly developed software component can fulfill its functions only by communicating

with other related components.

(3) Locked-in effect on software users

Users of information technology accumulate necessary data and install useful applications in their systems, and many different systems are networked for interoperation. A principle of behavior beyond product performance and price competition comes to rule the market if specific software vendors of these systems become dominant.

(4) As described above, the software sector is multilayered and communication-enabled and tends to have a locked-in effect on users. Because of such characteristics, the granting of patents may have created unduly powerful exclusive rights in this sector. This could generate adverse effects on innovation because of inhibited competition.

(5) Even in the software sector, with such characteristics as described above, the majority of patents are exercised according to the original intent of the patent system. Therefore, attention should be focused on establishing systems and conditions for securing innovation.

(6) Forthcoming legal response

“ Rules for Economic Transactions in the Market” should be established to specify the possibility of “abuse of right” for such conducts as exercise of rights that hinders the functions of software components to communicate with other components, including activities those made by patentees to restrict transactions of third parties or to exploit their patents against public welfare for the purpose of maximizing their monopolistic powers.

(7) Action by the industry

Action should be taken to propagate a concept along the lines of “Creative Commons,” and to popularize, through agreements among private enterprises, the business practices of mutual non-assertion of rights to such patented inventions as relating to certain categories of software, such as OSS, or to interoperability of software, thereby making this concept the standard in the industry, going beyond the previous patent system or standardization activities.

(8) Other issues that require further consideration include a compulsory license system and enhanced application of the Antimonopoly Law.

In March 2007 METI added the “Rules Concerning Software-Related Intellectual Property Rights,” which reflected public comments made in regards to the Interim Report and to the “Rules Concerning Electronic Commerce & Standards Pertaining to Electronic Commerce, Information Property Transactions, etc.”

Under these rules, with respect to the exercise of a software-related patent right, an abuse of right (as stipulated under Japan Civil Code Article 1(3)) may be found when the enforcement of said right is made as described below.

[Rules Concerning Software-Related Intellectual Property Rights]

With respect to the exercise of a software-related patent right, an abuse of right (as stipulated under Japan Civil Code Article 1(3)) may be found when the enforcement of said right is made as described below. To claim an abuse of right, one may file a lawsuit to make a plea against the claim of rights by the enforcing party or to affirm the non-existence of rights to claim an injunction by the enforcing party.

i. Where bad faith, such as the intention to inflict harm, is found in the subjective view of the enforcing party

ii. Where bad faith is found in the manner of enforcing said right, such as the intention to unreasonably cause a disadvantage to the parties affected by said enforcement of the right

iii. Where, in contrast to the benefit to be gained by enforcing said right, a considerably large disadvantage is caused to parties affected by said enforcement of said right as well as to society

Basically, the “Rules Concerning Software-Related Intellectual Property Rights” stipulate that the exercise of a software patent right may be restricted in cases where the result could have far-reaching implications for the software market.

In December 2007, the Japan Patent Office established the Policy Committee on Innovation and Intellectual Property (PCIIP) for the purpose of discussing national policy on intellectual property in order to promote future innovation. The results of these discussions were included in the draft policy recommendations and draft report that the PCIIP released in June 2008. The following are excerpts from the report that pertain to the patent troll issue and patent systems for promoting innovation.

(1) The U.S. and other nations have come to pay due attention to the patent troll problem because it is seen as a factor that hampers innovative activities. The patent troll issue encompasses a wide variety of means of intellectual property rights enforcement. It is

difficult, therefore, to clearly define this issue. A solution to this issue, therefore, will require careful consideration from diverse perspectives including IP systems, “abuse of right” principle in the civil law, and standardization.

(2) Vertical-integrated innovation is being replaced by open innovation in which an entity can make profits by allowing outside players to use the entity’s own technologies or can achieve speedier R&D and commercialization by utilizing outside technologies. Some people see intellectual property rights serving as a kind of currency in an open-innovation environment because it enhances the smooth flow of knowledge and technology. Under a closed-innovation environment, an intellectual property right holder can exclusively use a technology that he or she developed. In addition to such a role, intellectual property rights are expected to function as an infrastructure that will help facilitate the distribution of knowledge/technology.

In January 2009, the Nikkei Shimbun reported that the JPO was preparing to conduct a study on major revisions to the Patent Law, which would include the “addition of intangible properties such as software to the subjects of patent protection.” Specifically, the article stated that the JPO would review 1) revisions to the definition of the “invention” to be protected,” 2) revisions to stipulations pertaining to employee inventions, 3) legislating of the examination standards, 4) methods for resolving disputes in a timely and efficient manner, 5) expedition of examinations and meeting applicants’ needs, and 6) making the content of the Patent Law easily understandable. The article also stated that the study would be conducted for a period of one year by an unofficial study group reporting to the JPO commissioner. A discussion of the findings will be held by the Industrial Structure Advisory Council in 2010. There are plans to present a draft revision of the Patent Law or a new law in an ordinary session of the Diet in 2011 and enact the legislation in 2012. This movement appears to have arisen based on the recommendation stated above, and it is necessary to watch for future developments.

## 10. Corporate Software Patent Strategies

In a March 2007 METI survey on the awareness of software patent protection, the companies that responded gave the following reasons for procuring software patents:

- To establish exclusive rights with respect to their own inventions (24%)
- To generate profit through the licensing of their own inventions (27%)

- To enter into mutual licensing agreements with other companies (12%)
- To prevent other companies from patenting similar inventions (i.e., defensive applications) (31%)
- To defend against other companies attempting to exercise patent rights (19%)
- Other reasons (1%)

As indicated, the most popular reason among companies for acquiring patent rights is “to prevent other companies from patenting similar inventions (i.e., defensive applications”; only 24% of companies responded “to establish exclusive rights with respect to their own inventions.” In other words, the majority of companies do not acquire software patents for the purpose of establishing exclusive rights on their own software products or software-related businesses. Namely, the majority of companies adequately understand the difficulty of promoting their software-related businesses by relying solely on software patent protection.

Based on this notion, the following section discusses the “creative cycle” of corporate software patent strategies: creation, protection, and utilization.

### Creation Phase

In this stage, it is important for the company to identify future (including near-future) issues and create ideas for solving them. For example, a man named Charles Freeny acquired a patent (U.S. Patent No. 4528643) that specifies a system for delivering via a network game software, music, and other content at low prices; this patent was granted in 1985—before the Internet was widely used. Freeny identified a future issue (how to deliver content via a network) and created an idea for solving it.

Additionally, it is also important in this stage to properly search for prior art and other publicly known technology in order to determine the patentability of an invention. This is particularly true in the software field, where a vast number of people are developing software; there is always a good chance that a seemingly original idea has already been patented. Specifically, a company should first conduct a patent search to acquire an understanding of the prior art that exists in order to evaluate the patentability of its own invention. Next, the company should to the greatest extent possible research patent documents and non-patent literature, and again evaluate the patentability of its own invention. Compared to patentability evaluations based on patent documents, evaluations that are based on non-patent literature require much more work due to the fact that existing non-patent literature databases are inadequate; furthermore, research in such cases is frequently insufficient.

Therefore, rather than attempting to patent all of its inventions, a company should instead focus on patenting only certain inventions. That is to say, it should discriminate between 1) inventions that should be disclosed, 2) inventions that should be patented, and 3) inventions that should be kept confidential. In either case, however, a patent search must still be conducted to understand prior art trends; this is because attempting to develop an invention without first acquiring an understanding of prior art and inventions for which patent applications have already been filed could considerably diminish the value of the results.

### Protection Stage

Most of the preparations required for this stage should have been taken care of in the previous "creation" stage. In this stage, a company must research truly recent publicly known technology and conduct a subsequent patentability evaluation to fill in any gaps remaining from the previous stage.

What is most important in this stage is making adjustments to ensure that the invention in question is patentable in accordance with examination guidelines for software-related patents, patent laws, and other rules and regulations. Specifically, a company must 1) acquire an accurate understanding of enacted laws and revisions in addition to court judgments in software-related patent cases, and 2) properly reflect that knowledge in efforts to acquire its patent. This must be done to avoid the lessons learned in Japan's business method-related patent boom, where a large number of poorly prepared patent applications resulted in a high rate of rejections.

### Utilization Stage

With respect to the exercise of software-related patent rights, a company must give serious thought to whether the patent is to be used for licensing agreements or whether it is to be used to request injunctions. As mentioned above, an METI survey on the awareness of software patent protection found that the majority of respondents do not acquire software patents for the purpose of establishing exclusive rights on their own software products or software-related businesses. Based on the notion that—compared to hardware products—software products more commonly work in concert with other products in order to deliver the specified functionality, a company should act prudently if it decides to pursue and injunction.

According to a database of patent dispute case studies, which is based on publicly available information and managed by the author of this guide, nearly 240 new disputes arose between

2005 and 2007, and nearly 110 settlements were reached during the same period. Of the settlements reached, roughly 70% involved licensing agreements.

In other words, in the majority of disputes where settlements were reached, the patent holder was able to enter into a licensing agreement with the opposing party without having to enforce an injunction. Considering the fact that the dispute cases above include both hardware- and software-related cases, it is presumed that the 70% rate of settlements based on licensing agreements can be attributed to software-related cases.

Japan Patent Office

Collaborator: **Chozaburo MINAGAWA**
Professor,
Department of intellectual Property Strategy
Graduate School of Management of Science
and Technology
Tokyo University of Science