# Swiss Cheese Windows

## *Estimating Some Costs of the Nine State Remedy*

Authored by:

**Stan J. Liebowitz**

**Professor of Economics**
**Management School of the University of Texas at Dallas**

## Table of Contents:

# Executive Summary

A new wrinkle in the Microsoft antitrust trial is the separate track taken by nine states who have dissented from the settlement reached by the Department of Justice and nine other states. While these nine states have dropped the proposal to break-up Microsoft, they have instead proposed a remedy that pokes enough holes in the Windows platform to suggest the Swiss cheese analogy. One key element of their proposed remedy relates to the replacement of middleware components of the Windows operating system. These states would have Microsoft completely remove its middleware code from Windows at the request of a computer manufacturer or third-party licensee. The effects of this proposed remedy, if it were implemented, would be felt by many in the information technology (IT) industry--software developers, service firms, resellers, business users and consumers.

The aim of this report is to assess the direct impact on software companies who create, market, and support applications that already run on Windows. The impact is significant.

- **Tens of billions of dollars in extra costs over the first three years.** PC software producers would incur as much as $30 billion to $80 billion over the next three years in development, testing, marketing, and support costs. This extra cost is due to software developers having to adapt their software to new middleware they might not prefer because there is no safety net middleware they can always count on being available to all Windows computer users**.**

- **Consumers and producers will pay the price**. Software companies would deal with increased costs in two ways: pass costs along to consumers and business users in the form of higher prices; or elect not to support some middleware products. The effects of reduced margins and fragmented markets will drive some software producers to exit the business altogether, particularly the small producers.  Consumers pay either way--higher prices, fewer choices of software, less certainty that software they buy will run properly.

- **No balance of costs and benefits.** The magnitude of these costs and disruptions should be seen in relation to the settlement agreed upon by the Department of Justice, nine other states and Microsoft. That settlement also allows OEMs and users to remove any Microsoft middleware from the desktop and replace it with competing middleware products. The settlement does not, however, require that the actual code be removed. This allows a 'safety net' for developers who, if they so choose, can rely on the presence of "default" middleware from Microsoft.

# I. Introduction

Although the Department of Justice and nine states have reached a settlement with Microsoft in its antitrust case, nine other states remain opposed to that settlement and have proposed a remedy of their own [I use the term "proposed remedy" or "nine-states remedy" to refer to the proposal put forth by the nine dissenting states and the term "Settlement" to refer to the agreement reached by Microsoft, the Department of Justice, and nine other states]. There are many components of the proposed remedy that are worthy of scrutiny, but in this paper I wish to focus on one area that is central to their proposed remedy and also related to some prior analysis that I have conducted over the last several years.

A major component of the proposed remedy—the restriction on binding middleware—is based on the assumption that it will help consumers by instilling greater competition in the market for 'middleware' products—products that are deemed to work as a part of the operating system, but not integral to the operating system.[1] In fact, as described by the states, their overriding goal is to provide competition in the market for operating systems, and their hope is that one or several of the middleware products that their remedy is intended to benefit will grow to become viable operating system(s) on its own. That latter premise seems to be to be of dubious merit since it seems highly unlikely that an email program or an instant messaging program could evolve, even under the most wildly improbable circumstances, into a viable operating system.[2]

---

[1] The definition of middleware, and indeed the concept itself, is murky and imprecise. The nine states provide a definition which states: " 'Middleware' means software, whether provided in the form of files installed on a computer or in the form of Web-Based Software, that operates directly or through other software within an Operating System or between an Operating System (whether or not on the same computer) and other software (whether or not on the same computer) by offering services via APIs or Communications Interfaces to such other software, and could, if ported to or made Interoperable with multiple Operating Systems, enable software products written for that Middleware to be run on multiple Operating System Products. Examples of Middleware within the meaning of this Final Judgment include without limitation Internet browsers, network operating systems, e-mail client software, media creation, delivery and playback software, instant messaging software, voice recognition software, digital imaging software, the Java Virtual Machine, calendaring systems, Handheld Computing Device synchronization software, directories, and directory services and management software. Examples of software that are not Middleware within the meaning of this Final Judgment are disk compression and memory management software."

[2] Far more likely competitors to Windows are other operating systems. Programs such as SoftPc (on the Macintosh) or Lindows (promised software on Linux) directly address the application barrier to entry by allowing users to run Windows programs under different operating systems. These "solutions" have not fared all that well historically, probably in part because the software ran more slowly, and in part because most users of Windows software are happy to run it directly under Windows. Of course, programs written to run in Netscape using Java also would run more slowly than applications

Nevertheless, my main focus here is on a different issue. Discussions in the legal case have thus far ignored the costs to developers, consumers, and society at large of allowing OEMs to remove the safety net of Microsoft middleware program functionality from Windows. I am not arguing against allowing OEMs to choose third party middleware in place of Microsoft's, which the current Settlement requires, but merely that Microsoft should not be forced to remove its middleware code from Windows.

It is unreasonable (although not unlawful) to propose a remedy without some articulation and estimate of its costs and benefits. More importantly, it would be a grave dereliction to ignore the costs to consumers, the putative recipients of the remedy's benefits. This report is an attempt to examine some of the costs of the proposed remedy, and to contrast the benefits and costs of the proposed remedy with the Settlement.

The costs imposed by the nine state remedy depend upon which of three possible scenarios comes to pass. The first scenario is that envisioned by the nine states. The stated intent of their remedy is to encourage multiple versions of each middleware type to be distributed in the hope that these non-Microsoft products will achieve market success and themselves become serious challengers to the Windows computing platform.[3] The second scenario is one where significant costs are imposed upon Microsoft by forcing it to alter its operating system, but that in the end few OEMs and users actually switch away from Microsoft middleware products so the only impact of the remedy is to impose needless costs upon Microsoft by forcing it to alter its operating system for no purpose. A third scenario is one where some ISVs conclude that it's too much effort and uncertainty to rely upon *any* middleware APIs. Instead, they would incur costs to develop their own code to replace functions they had previously called from Microsoft's middleware APIs. Or, instead of writing their own replacement for middleware, ISVs might drop features from their software that relied upon current or planned Microsoft APIs, such as voice recognition.

---

written especially for a particular operating system. This further demonstrates the bankruptcy of the applications barrier to entry theory.

[3] As I explain below, I believe the notion that middleware products will transform themselves into serious competitors to Windows is without merit. Nevertheless, I treat it as a serious proposal here since the Courts seem to have accepted this theory.

This report will focus on the first scenario, since the nine-state remedy should be judged at least in part based on the outcomes it intends to create. My own belief is that only a small number of alternative middleware products will be chosen by OEMs.[4]

The costs that will receive the greatest focus in this report are the additional costs to software developers of creating, marketing, and supporting their programs in an environment where Windows is configured with middleware provided by firms other than Microsoft and *where the Microsoft middleware code is removed*. This third-party middleware will have the effect of Balkanizing Windows, raising incompatibilities between versions with different middleware components and making it more costly for developers to write programs that work seamlessly with all middleware-versions of Windows.

This Balkanization of Windows that is so fervently hoped for by the nine states (and the Microsoft competitors advising them), if it were to come to pass, will impose large costs on both software vendors and software consumers. It will hurt software producers because they will have to incur extra costs in developing and supporting programs to run under competing versions of Windows. It will hurt consumers because they will have to pay the resulting higher prices for this software, suffer under a less predictable software environment, and forgo the benefits of participating in a large network where compatibility could be taken for granted, one of the advantages historically bestowed by Windows and consciously promoted by Microsoft.

The most severely hurt are likely to be small software producers who will be less able to absorb the additional costs of creating and maintaining multiple versions of software and who will be forced by higher costs to leave the industry in disproportionate numbers.

---

[4] This depends, in part, on how much of a financial impetus is provided to OEMs to eliminate Microsoft middleware. The nine-state remedy requires that Microsoft give discounts for any middleware removed where the discounts are equal to the share of R&D on a middleware product relative to the R&D costs of Windows. It isn't clear how large these discounts will turn out to be.

## II. What might be the benefit from alternative middleware products in Windows?

This claim that absent Microsoft's anticompetitive actions, alternative operating systems would have seriously challenged Windows was never seriously examined in the court records, as far as I can tell. Instead, much was made of the "applications barrier to entry" a claim that the Windows OS was impervious to competition because it had so many more applications than its competitors did. There is validity in the claim that consumers are unlikely to switch to an operating system if they cannot use or find favorite programs, or if they need to repurchase a large percentage of their old library of programs. Certainly, a competing operating system that allowed consumers to keep their old hardware and software would have an advantage over an operating system that did not.

The story accepted by the Courts, however, is one that presumes that an item of middleware that exposes APIs to programmers will grow to become a platform on its own that would attract entire libraries of productivity, communication, and entertainment applications. The fact that this middleware already runs under Windows would mean, according to this theory, that software developers considering creating programs to run on the middleware-originated platform need not fear that their programs would have a limited potential audience since all Windows users could run their software as long as they had the middleware that went with it. Eventually, under this hypothetical scenario, the middleware platform would generate a sufficient library of programs that it would become a viable alternative to Windows in its own right. At that point, if this middleware could run on other operating systems, Windows would lose its program-library advantage over alternative operating systems and would only retain its market share and customers if Windows provided a superior computing experience for the money, exclusive of any advantage in the library of applications.[5]

---

[5] In the words of the nine-state remedy (my italics): "The applications barrier to entry, coupled with Microsoft's 90% plus market share, gave Microsoft the power to protect its "dominant operating system irrespective of quality" and to "stave off even superior new rivals." Id. at 56. During the mid-1990s, Microsoft was confronted with a potential threat to the applications barrier to entry, and thus to its monopoly power, in the form of two new products, Netscape's Internet browser, known as Navigator, and Sun Microsystems' Java technologies. Recognizing the threat posed by these middleware products (i.e., software that can itself be a platform for applications development), Microsoft aggressively and unlawfully prevented these rivals from achieving the widespread distribution they needed to attract software development and ultimately make other platforms meaningful competitors with Microsoft's Windows operating system…any remedy must prevent Microsoft from continuing the practices it used to artificially enhance and protect the applications barrier to entry…A meaningful remedy must ….be forward-looking with respect to technological and

Even if I were to accept this story as reasonable (and although the Court accepted it for Netscape and Java, to me it has always been a most fanciful fiction) one must ask whether the nine-state remedy would bring about the conditions under which additional operating systems might arise to challenge Windows. Common sense indicates that it is extremely far fetched that most of the items categorized as middleware could grow into alternative operating systems.

Here is a partial list of the Microsoft middleware that the nine-state remedy targets for "removal upon request":

- Internet Explorer
- Outlook Express
- Windows Media Player
- Microsoft Messenger
- Voice Recognition Software
- Digital Imaging Software
- Directories
- Exchange
- Calendaring Systems
- Systems and Enterprise Management Software
- Office
- Handheld Computing Device Synchronization Software
- Directory Services And Management Software

Take the case of handheld computing device synchronization. This is a program that allows a device, such as a Palm Pilot or PocketPC, to transfer data back and forth with a host computer using a docking station, serial cable, or some other mechanism to allow communications between the machines. This is a very specialized software task. Why would anyone ever imagine that such a narrowly defined program would turn into an operating system? It would seem to have about as much possibility of becoming an alternative operating system as would the solitaire game found in Windows.

---

marketplace developments, so that today's emerging competitive threats are protected from the very anticompetitive conduct that Microsoft has so consistently and effectively employed in the past. *Only then can the applications barrier to entry be reduced and much-needed competition be given a fair chance to emerge."*

It is so far-fetched as to be beyond the pale, yet the nine-states specifically include it as middleware that needs protection on the chance that it might become a challenger to Windows.[6]

Similarly, digital imaging software has the very narrow task of allowing computers to download, manipulate, share, and print images captured with a digital camera. It too has no realistic chance of becoming an alternative operating system. The nine-state remedy would appear to be to help promote alternative applications (for example, digital imaging software from Kodak, a company that complained loudly about Windows XP, or the Instant Messaging software from AOL Time Warner) even without any possibility of increasing competition in the operating system market. Since the Department of Justice/Microsoft settlement already includes provisions ensuring that APIs be made available to virtually all developers, the additional restrictions in the nine-state remedy serves no useful purpose. Instead, the nine-state remedy allows companies such as Kodak and AOL, in collaboration with OEMs, to add their applications to Windows while hiding their APIs from the rest of the world and potentially closing the market off to its competitors, including Microsoft.[7] This can hardly be to the advantage of consumers. It seems apparent that the nine-state remedy might be better understood by examining the efforts of the firms most heavily lobbying for "tougher" remedies, such as Sun Microsystems and AOL Time Warner.[8]

---

[6] Of course, the nine-state remedy includes it as middleware in order to 'protect' the market leading Palm operating system (and the hardware that runs it) that is currently competing with PocketPC products made by the likes of Compaq and HP that use Microsoft's PDA operating system. Under the (Department of Justice/Microsoft) Settlement, Microsoft would need to make its API's available to Palm on a timely basis, allowing Palm to create hardware and software that allows synchronization directly within the Windows environment and putting both companies on an equal footing. Under the nine-state remedy, by contrast, Palm need not make the API's found in its middleware available to Microsoft or other third parties. For those PCs that contain the Palm middleware in Windows, users of the PocketPC might not have an elegant synchronization solution. Microsoft would have to provide an alternative method for its PocketPC devices to synchronize with Windows computers, increasing its costs and putting it at a competitive disadvantage relative to Palm in a market where Palm is still the dominant force. This tilting of the market in Palm's favor does not benefit consumers who should be allowed to choose their PDAs based on their inherent advantages and not a court-rigged comparison.

[7] In principle, a digital imaging company would be in a position to pay the most to OEMs if it could monopolize the digital imaging market, since that would maximize the digital imaging profits.

[8] See for example, "Nine states stand firm in opposition to proposed Microsoft settlement" D. Ian Hopper, The Associated Press November 27, 2001 at http://seattletimes.nwsource.com/html/themicrosofttrial/134371532_webmicro27.html. Or this from ACT: http://www.actonline.org/getIT/Christmas%20for%20Rivals-ATL.doc.

## A.  The Disadvantages of the Nine-State Remedy.

A similar analysis can be made for virtually all the middleware products listed. On the one hand, the Settlement directly handles the question of Microsoft disadvantaging rivals by explicitly forbidding Microsoft from failing to provide timely access to Windows APIs. The nine-state remedy, on the other hand, allows a middleware producing company to gain advantages over its rivals by producing middleware that works only with its own products.[9] It also imposes additional costs on developers by forcing them to create multiple versions of their programs.

It is easy to imagine leaders in these middleware markets lobbying for and supporting the nine-state remedy since it could easily enhance their leadership positions, whereas the DOJ/Microsoft Settlement puts all middleware firms on an even footing. The nine-state remedy is not a result consistent with promoting competition.[10]

The Nine-State remedy has the potential to impose much larger costs on developers than does the Settlement. It does so by requiring Microsoft *to remove* its middleware code from Windows upon the request of an OEM or licensee.  In section 1 of the Nine-State remedy, titled "Restriction on Binding Microsoft Middleware Products to Windows Operating System Products," Microsoft is supposed to provide "*an otherwise identical version of the Windows Operating System Product that omits any combination of Microsoft Middleware Products*." The meaning of the term "omits" would appear to require that Microsoft remove the middleware code from Windows. Further, the previous paragraphs in the Nine-state remedy talk about preventing Microsoft's "commingling" of middleware code and

---

[9] Producers of competing middleware have far less incentive to make their APIs readily available than did the pre-settlement Microsoft. If Microsoft had limited its OS and not made APIs available to third parties, the OS itself would have suffered in the market since there would have been fewer applications written to use it. Middleware producers, on the other hand, have little incentive to make their APIs available since a decline in the Windows market brought about by their hiding APIs causes them little harm to them since they are only a small component of the Windows OS, yet they enjoy the full benefit from not exposing their APIs.

[10] One might argue that non-selected middleware producers still can produce their own standalone products that run under Windows, although not as a part of Windows. Although true, the selected middleware producer can try to prevent third party applications from running easily or properly, something that Microsoft had less incentive to do prior to the antitrust case and cannot legally do under the Settlement.

Windows code. Commingling only becomes a problem when the goal of the nine-states would be to have the middleware code physically removed.[11]

This is the crux of the problem with this portion of the nine-state remedy. If OEMs choose different middleware products, as the Attorneys General of the nine states hope, developers can no longer count on standard versions of Windows. They will now have to develop different versions of their products to run on the different middleware programs that might be contained in "Windows." These extra versions require separate coding, testing, installation routines, documentation, packaging, technical support, and so forth.

Writing programs to run under the different versions of Windows will be somewhat like 'porting' programs from one operating system to another, although the difference between the different 'versions' of Windows will be less dramatic than for completely different operating systems. Nevertheless, there will be potentially enormous costs due to the large number of middleware programs and the possible rewriting of the programs to work under this new environment. The details of estimating the fragmentation costs are reported in Section IV below.

The potential fragmentation problem can be illustrated with the example of sound playback software (one of the components that could be turned off under the Settlement). Under the Settlement, developers could count on all (Windows) users having access to the Microsoft media player, since it is normally installed with Windows. Even if an OEM removed the Microsoft media player, it would be available to programs that make calls to its APIs.

Under the Nine-state remedy, however, software developers could no longer count on users having access to Windows media player. Some OEMs might decide to include the Windows Media player, while others might prefer to reduce their costs by including an alterative, such as RealPlayer, WinAmp, or QuickTime, or perhaps not including any player at all. On Christmas morning, when little Johnny turns on the computer to play his new video game, there will be no sound if his parents purchased a

---

[11] Commingling the code makes it difficult and time consuming to remove middleware from Windows. Microsoft's critics viewed its commingling of Internet Explorer code as an attempt to impede any proposal to remove the code from Windows and still have Windows work. As a practical matter, Microsoft's critics were unwilling to risk the wrath of consumers, OEMs, and much of the computer industry, by shutting down the sale of Windows completely while the code was removed. Thus Microsoft's commingling of code was viewed by Microsoft critics as a tactic delaying the removal of Internet Explorer.

computer missing the needed media player.[12] The software developer now has one very unhappy customer.[13]

Multiply this problem by many potential middleware products, and it is easy to see how a fragmented market will harm consumers. This is a potentially enormous problem, with no likelihood of accomplishing its stated goal—to encourage new platforms that would compete with Windows as a personal computer operating system.

## B. How the Settlement Avoids These Costs While Still Allowing Choice.

The Settlement between the Department of Justice/nine-states and Microsoft also allows middleware producers to have their products used in place of Microsoft's products. But it does so in a far more reasonable way—OEMs or users get to remove the middleware programs from the desktop and system with a newly beefed-up add/remove feature.[14] The underlying middleware code still remains available to programs requiring it.

This seemingly small distinction makes a world of difference. The costs that developers face are dramatically less under the Settlement. First, developers can count on the Microsoft middleware product being there and can write to its APIs with the assurance that the APIs will reside in the operating system used by their consumers.

Of course, independent software developers, if they believe an alternative middleware product is superior or sufficiently popular, can write for that alternative middleware in addition to the Microsoft product, or exclusively for the non-Microsoft product. But if developers decide to write exclusively for

---

12 Many computer-game users have experienced this type of problem because the hardware (e.g., sound cards) in PCs is not fully standardized, and game developers generally write their games to work with only the leading sound cards since it would be too expensive to do otherwise. Although the packaging usually states the hardware requirements, many users are not sufficiently sophisticated to know whether the program will work on their machines.

[13] Defenders of this remedy might claim that the game developer could include on the distribution CD, along with the game, those components of Windows that are needed to run the game but which might have been removed by an OEM. In that case, however, the cost of the game would increase to cover the cost of buying these Windows components, needlessly raising prices for those customers who already have those Windows components installed. The game developer would also have higher support costs since his customers would come to him first if there problems with the middleware.

[14] This can be found in section H of the revised proposed final judgment (Settlement) between the Department of Justice and the nine states and Microsoft.

some third party middleware they will have made that decision consciously and will almost certainly take precautions to warn consumers that their product will work only if certain middleware products are installed. There will always be a safety net whereby the developer can rely upon Microsoft's default APIs if the preferred middleware is not found on the computer. The availability of such a safety net benefits both consumers and developers. The Settlement retains this safety net, whereas the nine-state remedy tears holes in the net.

## III.    What are the Incremental Software Production Costs from a Balkanized Windows?

Software producers incur extra costs by being forced to create multiple versions of their products to run under different operating systems. This will also be true for various flavors of Windows. The greater the number of non-Microsoft middleware products that are used in OEM machines, the greater the number of varieties of Windows developers potentially have to deal with. The more evenly distributed the market shares of competing middleware products, the greater the degree of Balkanization and the more costly it becomes for developers to ignore and write off relatively less popular middleware products.[15] These extra costs include building, testing, selling, and supporting their applications for specific operating systems. (A more detailed explanation of these extra costs can be found in Appendix 1.)  All these costs are incurred without a corresponding increase in revenue, since they are serving the same customers who were previously served with a single software version.

I calculate an estimate of the extra costs to a software developer for a platform-equivalent alteration to Windows. By platform-equivalent I mean a significant enough change in the APIs in

---

[15] If middleware product categories are dominated by a single producer, whether Microsoft or someone else, the Balkanization problem could go away. In that case a popular configuration of Windows would emerge with some of the middleware supplied by non-Microsoft vendors. Note, however, that I presume the nine states favor competition within middleware markets and not just competition between the Windows OS and some mythical middleware platform that arises to challenge it. Therefore, it seems fair to assume that their remedy envisions multiple competing versions of middleware and not just new monopolies in each middleware category. Further, if each middleware category becomes monopolized by some non-Microsoft firm, the cost of the Windows/middleware hybrid might rise significantly due to what is known in the economics literature as "double marginalization"—a confluence of multiple small monopolies charging higher prices than a single monopoly over the entire product would.

Windows that it becomes the equivalent of a different version of Windows such as a movement from Windows 3.1 to Windows 95. Of course, it is not possible to perfectly measure the number of platform-equivalent versions of Windows that might occur due to changes in middleware products configured with Windows. I will return to that issue later in the report.

## IV.   Estimating Incremental Software Production Costs

There is good deal of uncertainty about the actual future of middleware markets under either the Settlement or the nine state remedy. On the one hand, most consumers may prefer Microsoft middleware --or have no preference-- and there would therefore be no real change in the market or in the cost of developing software. In this instance, the sole costs will be those borne by Microsoft in redesigning its Windows product to fit either the nine state remedy or the Settlement.[16] The redesign costs of the Settlement require Microsoft to rewrite Windows to allow the removal of middleware functionality through an add/remove feature that is a beefed-up version of current Windows add/remove feature. I do not know how large these costs are, but Microsoft has already agreed to pay them. The rewriting of Windows required by the nine-state remedy, on the other hand, is far more intrusive. Microsoft's middleware code needs to be removed entirely from Windows, a task that is far more difficult, especially since the resulting version of Windows would have to "operate effectively and without degradation." [States remedy, p 5]. Nor is there any justifiable reason for doing so. These costs are likely to slow down improvements in Windows, reduce its utility, and appear to have no purpose other than to load Microsoft down with additional costs.

On the other hand, it is possible that OEMs will adopt a large number of non-Microsoft middleware alternatives. In this case, there will be many different versions of Windows that developers will need to consider when writing their software. It is this latter instance that seems to embody the vision of the nine states promoting it, and this is the one that I analyze below.

---

[16] There may also be some additional costs associated with uncertainty on the part of consumers and developers before the market clearly demonstrates that no change in middleware is going to be forthcoming.

## A. Procedure

I assume that spending on development, testing, marketing, and support would have to increase for each new variety of middleware that developers decide to support. Since higher costs will lead to higher prices,[17] unit sales of software would be expected to decrease from what they otherwise would have been. Depending on the elasticity of demand, revenues could either increase or decrease.

The estimate for the incremental software development and support costs involved with a new platform-equivalent version of Windows was based on a survey conducted in 1999 asking developers how much extra cost they expected to incur with additional versions of Windows, based on their prior experience in porting programs from one version of a platform to another version of the same platform. My instructions in that survey were: "I am investigating what the costs of some proposed remedies might be. Do you have any idea how much additional effort is required to port a product to different flavors of an operating system? (Win 3.1 and 95, or various flavors of Unix, say)."

These percentage estimates were then significantly scaled down so that they could be considered a minimum, or conservative, cost estimate and as such could be embraced by nonpartisans in the Microsoft controversy. The table below reproduces the survey results and the scaled down estimates:

| Table 1 | | | |
|---|---|---|---|
| Type of Cost | Average expected cost increase per new platform | Scaled down cost increase used in my estimates | Number of respondents |
| R&D | 78% | 25% | 11 |
| Support | 47% | 25% | 5 |
| Selling costs | 7.50% | 5% | 2 |

These scaled down estimates are then applied to estimates of R&D, Sales and Marketing, and Technical Support as a share of a typical software firm's total costs, as reported in a study conducted by KPMG and the Software Publishers Association (now known as The Software & Information Industry Association). These calculations are reported in the Appendix in table A1. After weighting these cost increases by the component share from the KPMG figures, the extra effort to developers for each full platform-equivalent version is seen to raise total costs for software developers by an amount

---

[17] It is standard economic analysis that increases in variable costs lead to higher prices.

equal to **6.46%** of revenues. That this number is so small is to a large extent a function of the fact that I have been so conservative in scaling down the estimated additional porting costs that were revealed by the survey.

This provides an estimate of the increase in costs for making a full platform version change. This increase in cost is then applied to the total Windows-based packaged software industry to estimate the costs of dealing with related-platform changes forced by the nine-state remedy. The data on current and predicted software revenues in the Windows market are found in Table 2. The numbers come from a 2001 IDC study representing expected revenues in Windows markets for various categories of packaged software.

There are three major categories of packaged software: tools, applications, and infrastructure. Infrastructure includes the sales of the operating system itself, and thus I remove an estimate of the sales of Windows operating systems from the sum of tools, applications and infrastructure revenues to derive the expected revenues from software in the Windows market. These estimates appear in the last row of table 2. These estimates are used to derive the dollar value of incremental costs of writing for multiple Windows platforms.

| Table 2: Predicted Revenue in Windows Markets (Millions) | | | | |
|---|---|---|---|---|
| | Forecast Period | | | 3-year |
| Software for Windows 32-bit Systems | 2003 | 2004 | 2005 | Total |
| Tools: spreadsheets, information access, programming environments, databases, components, objects, internet tools | $33,428.0 | $42,141.0 | $52,258.0 | $127,827.0 |
| Applications: consumer applications, personal productivity, games, accounting, office applications and other cross-industry packages, office automation, vertical industry applications | $54,206.0 | $66,999.0 | $80,244.0 | $201,449.0 |
| Infrastructure: system management, security, and networks | $30,030.0 | $35,346.0 | $41,142.0 | $106,518.0 |
| less: Microsoft's share of infrastructure revenue | -$5,193.7 | -$6,113.1 | -$7,115.5 | -$18,422.3 |
| Total Revenue for Windows software | $112,470.3 | $138,372.9 | $166,528.5 | $417,371.7 |
| Sources: IDC report "Worldwide Software Market, Forecast Summary, 2001-2005", Analysts: Richard V. Heiman, Dennis Byron, R. Paul Mason, and Melita Marks, June 5, 2001. Information on Tools comes from Table 16, Applications from Table 17 and Infrastructure from Table 18. Infrastructure revenues include sales of the operating system which were then removed based on sales figures reported in Table 13 of the IDC report. | | | | |

The predicted revenues from Table 2 make it easy to determine the predicted incremental production costs from porting to a related operating system. In table 3, the 6.46% increase in costs is applied to the revenues in the Windows market to arrive at a total dollar figure.

| Table 3: Increased Costs for one Extra Related Platform (in Billions) | | | | |
|---|---|---|---|---|
| Software for Windows 32-bit Systems | 2003 | 2004 | 2005 | Total |
| Total Revenue for Windows software | $112.47 | $138.37 | $166.53 | $417.37 |
| Incremental Cost Factor | 6.46% | 6.46% | 6.46% | 6.46% |
| Increased Cost due to one additional version | $7.3 | $8.9 | $10.8 | $26.9 |

In terms of dollar values, the absolute amounts are quite enormous because of the very large amount of resources that are devoted to producing products for the Windows market. From Table 3 we can see that the total increase in development costs involved with porting a program for a single related platform over the three year period is almost 27 billion dollars.

## B. Predicting the Development Costs of the Nine-State Remedy

Measuring the number of platform-equivalent varieties is probably the most difficult component of this exercise. The nine-state remedy enumerates twelve separate categories of middleware but does not limit the definition of middleware to just those categories. Nevertheless, if we restrict our attention to these twelve categories of middleware the number of potential varieties can be calculated under different scenarios. The pure number of feasible varieties of Windows is astronomical.[18]

There is little reason to believe that there would only be a single alternative middleware producer, particularly under the scenario developed by the nine states where they envision competition in the middleware markets. A real-world example of this would be the market for media creation software, where Microsoft (Windows Media Player), Real Networks (RealPlayer), Nullsoft (WinAmp), and Apple (QuickTime) all compete.

Developers do not have to worry about every possible combination of middleware combinations. For one thing, they do not necessarily all need to use every middleware category. Even if they did, it

---

[18] The theoretical maximum number of distinct varieties cannot be calculated without some assumption about the number of competing middleware products in each category. The formula is quite simple. If there are twelve middleware categories and each has a choice of two products, the number of different combinations is $2^{12}$. If there is a choice of three products, then the formula becomes $3^{12}$. And so forth. If we assume that there is only one middleware product alternative to Microsoft's in each of the twelve categories there would be 4096 different combinations of middleware products that developers would need to worry about. If a second alternative middleware producer existed in each category, the number of different combinations rises dramatically, up to 531,441 different combinations of middleware. If there is a chance that OEMs might leave certain middleware products out of the version of Windows they include with their machines, or if a third non-Microsoft middleware producer existed in each category, providing now 4 choices in each middleware category (including the possibility of leaving a category empty), there would be over 16,000,000 (sixteen million) combinations of middleware in Windows, all different from one another

would be easier to merely write programs for each middleware replacement than it would be to write a different version for each possible combination of middleware products. This means that if a developer writes software using six of the twelve middleware categories, and if there were four competing middleware products in each of those categories, that he would need to account for 24 middleware possibilities—in addition to the Microsoft middleware that may or may have been removed. This might require considerable rewriting of the original program, extra support personnel, and so forth. Of course, many programs will use only a small number of middleware components. Table 4 below provides some estimates of the increased cost to developers from having to deal with the more complicated Windows environment engendered by the nine-state remedy.

Table 4 allows us to examine the extra development costs as the number of middleware products in each category changes. The critical assumption in this table is that the cost to developers for each new middleware provider is equivalent to twenty percent of the cost of porting a program to run under a related operating system. This is not to say that each and every middleware product required exactly twenty percent of the effort of a full port to a related operating system. Some middleware is far more complicated than other middleware but that I am taking the average to be twenty percent. I believe that twenty percent is a conservative estimate under the logic that the nine-states envision in putting this proposal forward in the first place.[19]

The cost of altering a program in response to even a single middleware product change has certain fixed components that do not depend on the relative size of the middleware itself. Testing a product under various hardware or operating system environments is no different for small program changes than for large changes. Installation routines and user manuals need to be changed, as do instructions and equipment for technical support specialists. See Appendix 1 for a more detailed discussion of these costs. Additionally, it is almost certain that the logic and structure of the program itself will need to be altered to call new APIs with different functions and properties than those originally supplied by Microsoft.

---

[19] It is important to remember that the number of APIs involved in middleware would have to be quite large if the nine-states' overriding goal of having middleware morph to become an alternative to Windows were to have any chance of success. Since they assume that middleware might become an alternative operating system, and that is the logic behind their binding middleware provision in the first place, they must be willing to acknowledge that middleware must have a

Table 4 presents estimates of the additional development costs as the number of middleware products increase. I assume in this table that the number of alternative non-Microsoft middleware products varies from 2 to 5. I also assume that the average developer relies upon *just three of the twelve middleware categories*. When there are only two alternative middleware products in each category the additional porting costs are estimated to be $32 billion dollars. As the number of alternative middleware products increases, the costs continue to rise, reaching $81 billion with 5 alternative middleware products.

| Table 4: Costs From Supporting Non-Microsoft Middleware | | | | |
|---|---|---|---|---|
| Number of competing non-Microsoft products per middleware category | 2 | 3 | 4 | 5 |
| Number of new middleware components assuming developers call 3 middleware categories | 6 | 9 | 12 | 15 |
| Number of Platform Equivalent Versions if each middleware category imposes 20% of the cost of writing to a related platform | 1.20 | 1.80 | 2.40 | 3.00 |
| Incremental Cost over three years (in billions) | $32.33 | $49 | $65 | $81 |

It is important to remember that these costs are forced upon developers by the extraction of Microsoft's middleware code. If that code were not removed, developers could still add support for non-Microsoft middleware, while still being assured that their program would function properly on any computer with the "default" Microsoft middleware. Under the Settlement, the extra costs to developers writing for non-Microsoft middleware would presumably be less than the expected benefits, or developers would not take on these extra costs. That is why the Settlement can be presumed to create benefits greater than costs, but the nine-state remedy cannot.

---

reasonably large number of APIs and that software writers altering the middleware they write to would find it costly and difficult to do so. In this context, the 15% figure assumed in the text is quite reasonable.

# V.    The Impacts on Consumers and Producers

## A. Higher Costs Will Be Shared by Producers and Consumers

It is an economic truism that increases in variable costs are always split in some fashion by the producers and consumers.[20] Economic theory and decades of analysis make clear that producers will not be able to pass on all variable cost increases to consumers. Nor will the producers absorb the entire cost increase themselves. Variable costs that will be partly borne by consumers and producers are items such as increased technical support expense, and sales and marketing costs. Higher prices make consumers worse off and will decrease their purchases of the product. Higher costs to producers will cause some to exit the industry.

## B. Small Producers Will Exit the Industry in Disproportionate Numbers.

Unlike variable costs, increases in fixed costs are absorbed entirely by producers, reducing the profits of producers and causing some producers to exit the industry.[21] R&D (the design and creation of programs) is largely a fixed cost, and will increase due to the extra costs of rewriting the program. Since there is little reason to expect the cost of rewriting the program to have much to do with the sales of the program, firms with small sales are likely to have a relatively larger increase in R&D costs as a percentage of revenues than do firms with large sales (just as R&D in general is a higher percentage of the costs of firms with small sales in Table A1). It is also the case that the largest component of cost increases is the increase in R&D costs.

Therefore, the billions of dollars of increased development costs from having to alter programs to run with multiple middleware versions will fall disproportionately on small firms, causing them to leave the industry in disproportionate numbers. To the extent that small firms are responsible for new

---

[20] Virtually any microeconomics textbook will have a discussion of how the burden of a new variable cost, such as a sales tax, is split between producers and consumers.

[21] The logic here is that fixed costs can not affect the output level which maximizes a firm's profit since it is a cost that doesn't change when output changes. The fixed cost, however, reduces a firm's profit by an amount equal to the fixed cost, causing some firms to become unprofitable and eventually to leave the industry.

software ideas, this would prove damaging to the progress of the software industry and have negative effects on the consumers of software. [22]

## VI.    Costs That I Have Not Measured

I have focused on a single cost of adding new middleware into the Microsoft platform: the increased cost of producing software running on the Windows platform. This increased cost affects both producers and consumers. There are other costs that may potentially be of even greater magnitude but that I have not attempted to estimate in this study.

Not considered here is the cost to consumers of the decrease in compatibility that will arise for Windows users. No longer can consumers be assured that the programs that they purchase will run successfully under Windows, since some middleware products will likely garner greater market share than the others and applications will not work with computers lacking the proper middleware.

In economic terms, consumers are losing some value from the almost complete compatibility that Microsoft has worked so hard to achieve but which will disappear under a regime of multiple middleware inclusions in Windows.

Other large potential costs arise in related industries. For example, firms providing computer consulting services (which generate larger revenues than the software application market) will find themselves with additional costs and additional business as Windows-based customers try to wade through the now fragmented middleware environment. Although consulting revenues will likely increase as a result, this will be a net drain on the economy since resources will be removed from some productive activity into solving problems that were artificially created by the antitrust authorities.

In addition, independent resellers, or value-added resellers (VARs) are likely to have increased hardware and integration costs for their corporate customers, for many of the same reasons as consulting firms. The process of evaluating, selecting and installing application software packages will be more complex and costly.

---

[22] According to Nathan Associates, in 1997 there were over 230,000 small computer establishments in the US, over half of whom had annual sales under $500,000. These "high-tech entrepreneurs" generated over $300 billion in revenue in 1997 from services, software, and hardware. They employed 2.2 million Americans.  Source: "The New High-Tech Entrepreneurs", by Nathan Associates Inc. 1998.

Another service sector that depends upon operating system stability is the training and educational market. These companies will likely need to add resources while serving the same base of customers.

The consumer costs mentioned earlier were costs that individual users were likely to encounter. In addition to those costs, the internal information technology departments of nearly every organization that uses computers will also find its costs increased. Choosing hardware and software will be a more difficult task.

## VII.  Conclusions

The nine-state remedy, if it worked the way the nine states imagine, will increase costs to both software developers and the public at large, as well as Microsoft. Costs borne by independent software developers alone would rise by between ten and twenty five billion dollars per year.  Alternatively, it is possible that alternative middleware will not be embraced by the market and that a great majority of consumers will continue to prefer Microsoft middleware. In this latter case Microsoft bears the entire cost of massively restructuring Windows, but for no consumer benefit.

These costs are easily avoidable and are therefore wasteful. The Settlement between Microsoft/Department of Justice/nine states also allows middleware choice. The Settlement does not allow large middleware producers to disadvantage their smaller competitors. The Settlement does not impose unnecessary costs on software developers calling middleware APIs. Under the Settlement, Microsoft bears the costs of redesigning Windows, but in a way that is similar to how it already compartmentalizes Windows, and the redesign costs are far lower than under the nine-state remedy.

Given the clear superiority of the Settlement, it seems unreasonable to believe that the nine-state remedy could benefit either consumers or society, although it might benefit certain large Microsoft rivals eager to protect their market positions.

# VIII. Appendix 1: Extra Costs of Producing Software for Multiple Middleware Versions

## A. Research & Development

PC software companies must be selective about which operating systems they support because porting and maintaining their products on a new or altered operating system is costly, and these additional costs have to be justified by the incremental revenues expected from sales of the application in the new middleware environment. The costs discussed are for changed operating system, but they would hold as well for changes in middleware contained in the OS.

One famous analogous instance of increased R&D costs associated with porting applications was Lotus' attempt to upgrade its 1-2-3 spreadsheet from DOS to a graphical version that could run first under DOS and then later be ported to OS/2 and Windows. To enhance portability Lotus converted the code for 1-2-3 into to a high level programming language only to find that the converted program no longer fit within the memory confines of DOS. Lotus finally had to pay an outside firm to help them get the product to work, with an attendant delay of almost a year in the DOS version.[23]

Although many PC software companies have some experience in dealing with the incremental costs and revenues from porting their programs to additional operating systems, these decisions are full of guesswork--questions about the compatibility of the new operating systems to their well-known ancestors, about complexities of learning and handling new features that differentiate the operating systems, and performance characteristics under varying loads to determine "minimum" and "recommended" hardware configurations. There are numerous cases where firms made serious business errors in the decisions to port products. For example, we were told the following story in our interviews with software executives:

> "It took nine man-months to port just the time-entry module the Macintosh platform. That was greater than our effort to code the original module for Windows. We did lots of trial-and-

---

[23] "PC Spreadsheet Software" IDC 1991. From page 9: "The production of Lotus's upgrade to 1-2-3 Release 2.1 proved to be a difficult undertaking for the company…Release 3.0's production schedule stretched out as Lotus attempted to fit the product within reasonable memory restraints…Finally, Lotus went to an outside source, Rational Systems, and purchased DOS extender technology."

error performance tuning. In hindsight, we wouldn't do it again, because the Mac market did not grow like we expected."[24]

Once a software company decides to support a new operating system, they invest in design decisions about how to manage the conversion and maintain the products on multiple platforms. Source code that works on Windows today may have to be altered to find the lowest common denominator among the Windows variations. Much of the rest of the code ends up in "duplicate code trees" which increases the long-term complexity and cost of software maintenance.

After design decisions are made, development begins, and significant incremental costs arise. An incomplete list of some of these costs would include:

- Adding programmers and educating new and old programmers about application programming interfaces to the new operating system.
- Acquiring, equipping, and maintaining extra development computers for the new operating systems.
- Adding testing personnel with knowledge of the new operating system, and equipping them with testing computers running the new operating systems.
- Creating test plans to ensure that software users on multiple operating systems can co-exist and share data on the same network. This "inter-operability" testing increases in complexity with every new version supported.
- Creating, printing, and maintaining new on-line help and user documentation.
- Creating and maintaining system documentation.

## B. Incremental Sales & Marketing

I assume that the total number of software customers will not change as a result of having multiple versions of Windows middleware. Instead, customers who would have purchased application software that ran under the old unified Microsoft Windows operating system would now have to take care in matching the programs they purchase with the middleware that was configured with their version of Windows.

Some selling expenses will increase as the number of middleware products supported increases. When the new versions are first announced, a public relations campaign is usually undertaken. And

---

[24] Rene Adam, Ad Systems, Inc.

when the software is released, a high-profile campaign is typically unveiled, including advertising, public relations, conference booths and sponsorships at events catering to users and developer communities supporting the middleware. New versions of software are often created to take advantage of changes to the OS, so that these costs are likely to multiply with different middleware included with Windows.

For software that is distributed by a salesforce, there will be incremental spending on educating the salesforce about which versions of the OS contain the proper middleware. In addition, the salesforce would need to acquire and maintain incremental equipment to demonstrate their software on each version of middleware supported.

Finally, marketing costs may increase. Software companies will need to create new brochures, web pages, fact sheets, etc. without adding new customers.

## C. Incremental Support Costs

Adding new versions of middleware support will require additional training for current support staff, and increase the training and ramp-up time for new and replacement staff. Call handling times are increased by any complexity that makes it harder for support staff to understand and replicate a caller's problem. Support staff, if they are to do their jobs properly, needs to be able to replicate the same screens, features, and "bugs" as the callers. If callers are using multiple middleware components, support reps will be forced to use the same multiple middleware so problems can be replicated faithfully. This will likely require additional staff and computers, and/or additional training so the support staff can match the calls coming in.

## D. Evidence of these Costs

The cost of porting programs is readily apparent from many diverse pieces of evidence, as well as direct examination. For example, much of the excitement surrounding Java was due to its purported reduction in porting costs, captured in the phrase: "write once, run anywhere". If a program was written in Java, it was supposed to be able to run directly on any machine and operating system, avoiding the R&D costs of porting applications to other operating systems. The downside of this was that Java ran more slowly than programs written directly for a single operating system.

Probably the most famous instance of the importance of porting costs was Intuit's very public hesitation about continuing to support Quicken for the Macintosh a few years back. At the time, the Macintosh version generated approximately 10% of the revenues of the Windows version, indicating that the incremental costs of continuing to upgrade and supporting a Macintosh version must have been in the vicinity of 10% of total revenues.[25]

This 10% value can not be taken as typical, however. There are two factors that might lead us to conclude that it is too low, and one factor that might lead us to conclude that it is too high. On the biased downward side, Quicken had been supporting the Macintosh for many years, so the porting costs should have been considerably less than if Intuit needed to create a brand new port for which it had no experience. Quicken also had a much larger output base upon which to amortize the fixed costs of porting, compared to most other software programs. On the biased upward side, the difference between the Macintosh and Windows will undoubtedly be greater than the differences between the various flavors of a single operating system, which was the comparison in which I was interested.

We can also look to some industry articles for confirmation of the importance of these costs.

?? "Sandra Potter, an analyst at Aberdeen Group Inc. said that although many vendor executives know how much damage proprietary implementations did to the Unix market by killing compatibility among various flavors of Unix, they may still have trouble preventing the same from happening to Linux." [26]

?? "VA Research Inc., which sells computers with a choice of four distributions, may have to drop one because testing four is too burdensome, said CEO Larry Augustin." [27]

?? In an article about the future viability of Linux, the difficulties of producing software for more than a single implementation of the operating system was highlighted: "[Caldera CEO Ransom] Love said he's talked to software developers who are reluctant to jump to Linux because they fear it will fragment... Jeff Carr, the founder of Linux PPC, called the Linux standards base — a

---

25 An article with some of the details, although it prematurely reported the death of the Macintosh version is: "Sales drop killed Quicken for Macintosh: Intuit to discontinue finance product for the Macintosh," Sean Silverthorne, ZDNN, April 20, 1998. The article states that sales for Windows were 1.5 million versus 133,000 for the Macintosh (8.9%).

26 From "Linux-For-All Faces Obstacles" by David Orenstein, ComputerWorld Online News, 3/15/99 http://www.computerworld.com/home/print.nsf/all/990315960E.

27 "Choice, Not Standards, Drives Linux Users", by David Orenstein, ComputerWorld Online News, Mar 8, 1999 full story at http://www.computerworld.com/home/print.nsf/all/9903089576

move to provide common standards for Linux distributions — a crucial and critical movement." [28]

All this evidence is consistent with the conclusion that the economic costs of porting programs from one operating system to another, even for closely related operating systems, can be and often are significant. In many cases these costs are significant enough to deter porting, causing the software developer to forego additional revenues and restricting consumer choice.

Finally, we have some estimates from actual software vendors that were interviewed. [29]

?? Network Associates, a large producer of utility software approaching a billion dollars in yearly sales including the well-known McAfee antivirus software, provided this quote after researching the question of incremental costs: "As a leading developer of network security and management products across heterogeneous operating environments, Network Associates is confident that the costs of two competing Windows operating systems could range from zero after the initial split to up to as much as 200% over time as the operating systems diverged." [30]

?? Dr. Eric B. Allely, President, Tekamah, Inc., a firm involved with producing software for distance learning stated: "Based on the level of work to debug and maintain multiple versions, I would estimate our lab's overhead expenses to increase by 70 to 100%. That would translate into a 30 to 50% increase in product costs and more importantly an extension in the time it takes us to deliver a product." [31]

?? Charles Crystle, Founder and CEO of Chili!Soft states: "Maintaining software for different operating systems is very costly and is a serious drag on our business. To begin with it takes about 6 months for a team of 12 people to port to a new operating system. The additional cost for each new OS is about 80% of the original work. Two additional operating systems will cost us a great deal, both in real costs as well as in opportunity costs--instead of improving our software for our consumers we'll be busy moving it to a new OS." [32]

Of the eleven companies that responded to our requests for estimates of incremental costs, the average estimate was that R&D would increase by 78% for each new Windows platform, with a range of 15% to 100%. The average estimate for increases in support was 47%, although only five

---

[28] "Hamstrung by lack of standards? Linux bigwigs talk up desktop and embedded systems — and flag the need to promote standards for the OS," Lisa M. Bowman, ZDNN, CHICAGO, April 20 1999.

[29] These interviews were conducted by Steve DelBianco, who assisted in this project. Steve now works for ACT, although he did not at the time the interviews were conducted in the spring of 1999.

[30] Richard Greene, Network Associates, via email  4/27/99

[31] Email response on 4/23/99.

[32] Interview 4/27/99. Chili!Soft develops and markets software that enables Web developers to build dynamic Internet applications that work across many different computing platforms, including Windows NT, Sun Solaris and IBM AIX

respondents quantified support costs. Only two companies quantified selling cost increases, with both in the 5-10% range.

# IX. Appendix 2: Calculating Incremental Costs Per Platform

| Table A1: Incremental Cost Percentages for each new Windows version (as percent of revenue) | | | | | | | |
|---|---|---|---|---|---|---|---|
| Adapted from SPA/KPMG study | < $5 mil | $5-50 mil | > $50 mil | approximate weighted average | Incremental Factor | Impact on Total Cost | Explanation |
| R&D: | 26.90% | 20.90% | 16.80% | 17.32% | 25.00% | 4.33% | based on interviews; most implied greater than 20% increase |
| **Sales & Marketing Expenses:** | | | | | | | |
| Outside sales salaries | 3.50% | 4.20% | 5.20% | 5.08% | - | | no new reps needed |
| Inside sales salaries | 1.60% | 3.40% | 2.40% | 2.50% | - | | no new reps needed |
| Outside sales commissions | 1.20% | 1.20% | 1.60% | 1.55% | - | | no new sales |
| Inside sales commissions | 0.30% | 1.40% | 1.00% | 1.03% | - | | no new sales |
| Overhead/depreciation | 1.00% | 1.20% | 2.20% | 2.08% | - | | |
| Other sales expenses | 1.80% | 3.00% | 4.10% | 3.96% | 5% | 0.20% | increased training, demo equip |
| subtotal of sales | 9.40% | 14.40% | 16.50% | 16.21% | | | |
| Marketing salaries | 2.80% | 3.20% | 3.40% | 3.37% | - | | no new staff |
| Advertising | 4.30% | 3.00% | 1.80% | 1.95% | - | | more ads, but smaller circulation |
| Cooperative marketing | 0.60% | 0.70% | 1.00% | 0.97% | 5.00% | 0.05% | w/ new OS vendors |
| Trade shows | 1.80% | 1.50% | 0.80% | 0.88% | 5.00% | 0.04% | for new OS developers & users |
| Direct mail | 1.90% | 0.60% | 0.90% | 0.88% | 5.00% | 0.04% | new production costs |
| Promotions | 0.10% | 0.70% | 1.10% | 1.05% | 5.00% | 0.05% | new production costs |
| Other marketing programs | 1.30% | 1.30% | 1.90% | 1.83% | | | |
| Overhead | 0.40% | 0.50% | 0.60% | 0.59% | 5.00% | 0.03% | |
| subtotal of marketing | 13.20% | 11.50% | 11.50% | 11.52% | | | |
| Technical support salaries | 2.80% | 4.60% | 4.00% | 4.05% | | | more support people and training |
| Tech support overhead/depreciation | 0.70% | 0.50% | 1.60% | 1.48% | | | more support equipment |
| Total Technical Support | 5.50% | 6.40% | 6.90% | 6.83% | 25.00% | 1.71% | from interviews |
| Total Increase in Costs as a Percentage of Revenue | | | | | | 6.46% | |

Data come from: Table 3, page 10, SPA/KPMG. The Incremental Factor is our conservative guess. The approximate weighted average constructed by assuming that the 16 firms in the less than $5 million category average $2.5 million, the 15 firms in the $5-50 million category average 27.5 million, and the 21 firms in the $50 million + category average 50 million except for 10 firms that are over $300 million, which we assume averages to $300 million.

## X.    Addendum: The Cost of the Remedy by State

I was asked to calculate the costs of this remedy for all of the states as well as specifically noting the costs to the nine states (plus the District of Columbia). To do this I needed to apportion the total costs between consumers and producers, since they would be sharing the cost. With the limited data at hand it is not possible to determine exactly how the costs would be split between consumers and producers. Therefore I have assumed an arbitrary 50-50 split between the two groups. I also needed to determine the share of each state relative to the total for the US as a whole.

The first task, then, is to calculate the costs to consumers. This is done in Table 5.

| Table 5: Three Year Costs to US Consumers in All States | | | | |
|---|---|---|---|---|
| Number of competing non-Microsoft products in each of top three middleware categories | 2 | 3 | 4 | 5 |
| Incremental Worldwide Developer Cost over three years (in billions) | 32.3 | 48.5 | 64.7 | 80.8 |
| Cost to Consumers Worldwide if half of costs are passed on to consumers and then marked up by 60%, the average markup according to the KPMG study (in billions) | 25.9 | 38.8 | 51.7 | 64.7 |
| Costs to US Consumers (in billions) | 12.9 | 19.4 | 25.9 | 32.3 |
| Costs by State (in millions) | | | | |
| Alabama | 184.1 | 276.2 | 368.2 | 460.3 |
| Alaska | 34.5 | 51.7 | 69.0 | 86.2 |
| Arizona | 245.9 | 368.9 | 491.8 | 614.8 |
| Arkansas | 94.6 | 141.9 | 189.2 | 236.5 |
| **California** | **1,564.9** | **2,347.3** | **3,129.8** | **3,912.2** |
| Colorado | 250.9 | 376.3 | 501.7 | 627.2 |
| **Connecticut** | **188.4** | **282.6** | **376.8** | **471.1** |
| Delaware | 42.0 | 63.0 | 83.9 | 104.9 |
| **District of Columbia** | **28.8** | **43.2** | **57.6** | **72.0** |
| **Florida** | **763.5** | **1,145.2** | **1,527.0** | **1,908.7** |
| Georgia | 339.2 | 508.8 | 678.4 | 848.0 |
| Hawaii | 51.5 | 77.3 | 103.0 | 128.8 |
| Idaho | 61.5 | 92.2 | 123.0 | 153.7 |
| Illinois | 553.4 | 830.1 | 1,106.8 | 1,383.5 |
| Indiana | 274.5 | 411.8 | 549.0 | 686.3 |
| **Iowa** | **147.5** | **221.2** | **294.9** | **368.7** |

| Table 5: Three Year Costs to US Consumers in All States | | | | |
|---|---|---|---|---|
| Number of competing non-Microsoft products in each of top three middleware categories | 2 | 3 | 4 | 5 |
| **Kansas** | **139.7** | **209.5** | **279.4** | **349.2** |
| Kentucky | 176.6 | 264.8 | 353.1 | 441.4 |
| Louisiana | 166.5 | 249.7 | 332.9 | 416.1 |
| Maine | 67.7 | 101.6 | 135.5 | 169.4 |
| Maryland | 254.7 | 382.0 | 509.4 | 636.7 |
| **Massachusetts** | **311.6** | **467.4** | **623.1** | **778.9** |
| Michigan | 466.6 | 699.8 | 933.1 | 1,166.4 |
| **Minnesota** | **259.0** | **388.5** | **517.9** | **647.4** |
| Mississippi | 94.6 | 141.9 | 189.3 | 236.6 |
| Missouri | 277.3 | 415.9 | 554.5 | 693.2 |
| Montana | 44.5 | 66.7 | 89.0 | 111.2 |
| Nebraska | 78.9 | 118.3 | 157.7 | 197.1 |
| Nevada | 88.0 | 132.1 | 176.1 | 220.1 |
| New Hampshire | 72.5 | 108.7 | 145.0 | 181.2 |
| New Jersey | 399.4 | 599.1 | 798.8 | 998.5 |
| New Mexico | 77.4 | 116.1 | 154.8 | 193.5 |
| New York | 826.0 | 1,239.0 | 1,652.0 | 2,065.0 |
| North Carolina | 343.5 | 515.3 | 687.1 | 858.8 |
| North Dakota | 29.5 | 44.2 | 59.0 | 73.7 |
| Ohio | 528.9 | 793.3 | 1,057.8 | 1,322.2 |
| Oklahoma | 134.4 | 201.6 | 268.8 | 336.0 |
| Oregon | 195.8 | 293.7 | 391.6 | 489.5 |
| Pennsylvania | 554.9 | 832.3 | 1,109.7 | 1,387.1 |
| Rhode Island | 47.1 | 70.6 | 94.2 | 117.7 |
| South Carolina | 160.5 | 240.8 | 321.0 | 401.3 |
| South Dakota | 35.1 | 52.7 | 70.2 | 87.8 |
| Tennessee | 246.1 | 369.2 | 492.3 | 615.3 |
| Texas | 850.7 | 1,276.0 | 1,701.3 | 2,126.7 |
| **Utah** | **111.2** | **166.8** | **222.4** | **278.0** |
| Vermont | 31.1 | 46.6 | 62.2 | 77.7 |
| Virginia | 350.7 | 526.1 | 701.4 | 876.8 |
| Washington | 330.7 | 496.0 | 661.3 | 826.7 |
| **West Virginia** | **75.7** | **113.5** | **151.4** | **189.2** |
| Wisconsin | 254.7 | 382.0 | 509.3 | 636.6 |
| Wyoming | 27.1 | 40.7 | 54.2 | 67.8 |
| *Calculating US Share of Worldwide Software Sales comes from IDC Table 15 "Worldwide Packaged Software Revenue by Primary Market and Region, 2000–2005", which puts North American Share at 53.6%, and then IDC"PC Tracker Data, total PC shipments, excluding servers" puts US share of North American market at 93%, to arrive at my 50% value. Share of States in US total comes from: "Computer ownership data" CPS August Rotation.* | | | | |

In Table 5 I first assume that half the additional developer costs (from Table 4) are passed on to consumers. I then mark these costs up by the average retail markup (taken from the KPMG

study mentioned in the discussion near Table 1 of the report) to arrive at the world-wide consumer cost increase. The share of American consumers is then calculated, based on the US share of the worldwide package software market, and this comes out to almost exactly fifty percent of the world total. Finally, I break down the US total per state, based on computer ownership per state as a share of the national total. The state totals are in millions of dollars, whereas the rest of the table is in billions. The costs to the nine states pushing this remedy are marked in bold text.

The second task is to estimate the costs to developers by state. This is done in Table 6. To determine the developer costs in each of the states I assume that 75% of all Windows developers are in the United States. The lists of the top 50 applications vendors in the World is heavily skewed toward US developers and the 75% assumption seems fairly conservative. Also, the US is a net exporter of software meaning that the US share of total world sales must be larger than its 50% share of worldwide sales. The assumption that developers share the costs 50-50 with consumers is retained. These assumptions allow us to calculate the US share of additional development costs absorbed by the developers. To finish calculating the costs to developers, I use the number of independent software vendor employees per state to calculate each state's share of the development costs.

| Table 6: Three Year Costs to US Software Developers in All States | | | | |
|---|---|---|---|---|
| Number of competing non-Microsoft products in each of top three middleware categories | 2 | 3 | 4 | 5 |
| Incremental Worldwide Developer Cost over three years (in billions) | $32.33 | $48.50 | $64.67 | $80.83 |
| Net Cost to Developers Worldwide if half of additional development costs are absorbed by developers (in billions) | $16.17 | $24.25 | $32.33 | $40.42 |
| Net Costs to US Developers if US Developers make up 75% of World Market (in billions) | $12.13 | $18.19 | $24.25 | $30.31 |
| Costs by State (in millions) | | | | |
| Alabama | $146.46 | $219.70 | $292.93 | $366.16 |
| Alaska | $0.05 | $0.07 | $0.10 | $0.12 |
| Arizona | $34.76 | $52.14 | $69.52 | $86.90 |
| Arkansas | $0.46 | $0.69 | $0.92 | $1.15 |
| **California** | **$2,897.02** | **$4,345.54** | **$5,794.05** | **$7,242.56** |

| Table 6: Three Year Costs to US Software Developers in All States | | | | |
|---|---|---|---|---|
| Number of competing non-Microsoft products in each of top three middleware categories | 2 | 3 | 4 | 5 |
| Colorado | $139.91 | $209.87 | $279.82 | $349.78 |
| **Connecticut** | **$370.62** | **$555.94** | **$741.25** | **$926.56** |
| Delaware | $0.96 | $1.43 | $1.91 | $2.39 |
| **District of Columbia** | **$2.54** | **$3.81** | **$5.07** | **$6.34** |
| **Florida** | **$244.37** | **$366.55** | **$488.73** | **$610.92** |
| Georgia | $197.75 | $296.62 | $395.50 | $494.37 |
| Hawaii | $0.21 | $0.32 | $0.42 | $0.53 |
| Idaho | $2.01 | $3.01 | $4.02 | $5.02 |
| Illinois | $208.23 | $312.34 | $416.46 | $520.57 |
| Indiana | $28.76 | $43.15 | $57.53 | $71.91 |
| **Iowa** | **$10.60** | **$15.90** | **$21.20** | **$26.50** |
| **Kansas** | **$10.95** | **$16.43** | **$21.91** | **$27.38** |
| Kentucky | $5.84 | $8.76 | $11.68 | $14.60 |
| Louisiana | $8.50 | $12.75 | $17.00 | $21.25 |
| Maine | $6.29 | $9.44 | $12.58 | $15.73 |
| Maryland | $235.06 | $352.59 | $470.12 | $587.64 |
| **Massachusetts** | **$976.11** | **$1,464.16** | **$1,952.21** | **$2,440.27** |
| Michigan | $415.87 | $623.80 | $831.73 | $1,039.66 |
| **Minnesota** | **$187.52** | **$281.29** | **$375.05** | **$468.81** |
| Mississippi | $0.46 | $0.70 | $0.93 | $1.16 |
| Missouri | $69.75 | $104.62 | $139.49 | $174.37 |
| Montana | $0.47 | $0.71 | $0.95 | $1.18 |
| Nebraska | $20.79 | $31.19 | $41.59 | $51.98 |
| Nevada | $28.37 | $42.56 | $56.74 | $70.93 |
| New Hampshire | $91.63 | $137.45 | $183.27 | $229.09 |
| New Jersey | $1,734.88 | $2,602.32 | $3,469.76 | $4,337.20 |
| New Mexico | $2.51 | $3.76 | $5.01 | $6.26 |
| New York | $359.41 | $539.12 | $718.83 | $898.53 |
| North Carolina | $129.84 | $194.75 | $259.67 | $324.59 |
| North Dakota | $13.94 | $20.91 | $27.89 | $34.86 |
| Ohio | $82.90 | $124.35 | $165.80 | $207.26 |
| Oklahoma | $8.44 | $12.67 | $16.89 | $21.11 |
| Oregon | $69.37 | $104.06 | $138.74 | $173.43 |
| Pennsylvania | $374.15 | $561.23 | $748.30 | $935.38 |
| Rhode Island | $11.82 | $17.73 | $23.64 | $29.55 |
| South Carolina | $27.29 | $40.94 | $54.59 | $68.23 |
| South Dakota | $0.24 | $0.36 | $0.48 | $0.60 |
| Tennessee | $1,205.17 | $1,807.76 | $2,410.35 | $3,012.93 |
| Texas | $856.24 | $1,284.35 | $1,712.47 | $2,140.59 |
| **Utah** | **$183.82** | **$275.73** | **$367.64** | **$459.55** |
| Vermont | $46.15 | $69.22 | $92.29 | $115.36 |
| Virginia | $194.66 | $291.99 | $389.31 | $486.64 |
| Washington | $397.40 | $596.10 | $794.80 | $993.50 |

| Table 6: Three Year Costs to US Software Developers in All States | | | | |
|---|---|---|---|---|
| Number of competing non-Microsoft products in each of top three middleware categories | 2 | 3 | 4 | 5 |
| **West Virginia** | **$1.32** | **$1.98** | **$2.64** | **$3.30** |
| Wisconsin | $83.11 | $124.67 | $166.22 | $207.78 |
| Wyoming | $0.13 | $0.19 | $0.26 | $0.32 |
| *Share of States in US total comes from: Corporate Technology Information Services (CorpTech), "Corporate Technology Directory", ISV Employees per state.* | | | | |

The results indicate that some states, such as California and Massachusetts, are hit much harder in additional developer costs than was the case for consumer costs, due to their high concentrations of software vendors.

Finally, we can add together the costs to both consumers and developers in the nine states. That calculation is performed in Table 7. The numbers are quite large. These values can let the citizens of the states in question determine what the impact of the remedy might be on the citizens of those states who are supposed to be the beneficiaries of the actions of their attorneys general.

| Table 7: Three Year Costs of 9-state 'Remedy' in All States | | | | |
|---|---|---|---|---|
| Number of competing non-Microsoft products in each of top three middleware categories | 2 | 3 | 4 | 5 |
| Costs by State (in millions) | | | | |
| Alabama | $330.59 | $495.88 | $661.18 | $826.47 |
| Alaska | $34.54 | $51.81 | $69.08 | $86.35 |
| Arizona | $280.67 | $421.00 | $561.34 | $701.67 |
| Arkansas | $95.06 | $142.60 | $190.13 | $237.66 |
| **California** | **$4,461.91** | **$6,692.86** | **$8,923.81** | **$11,154.76** |
| Colorado | $390.78 | $586.16 | $781.55 | $976.94 |
| **Connecticut** | **$559.05** | **$838.57** | **$1,118.10** | **$1,397.62** |
| Delaware | $42.93 | $64.39 | $85.85 | $107.31 |
| **District of Columbia** | **$31.35** | **$47.03** | **$62.71** | **$78.38** |
| **Florida** | **$1,007.85** | **$1,511.77** | **$2,015.69** | **$2,519.62** |
| Georgia | $536.97 | $805.45 | $1,073.94 | $1,342.42 |
| Hawaii | $51.72 | $77.58 | $103.44 | $129.30 |

| Table 7: Three Year Costs of 9-state 'Remedy' in All States | | | | |
|---|---|---|---|---|
| Number of competing non-Microsoft products in each of top three middleware categories | 2 | 3 | 4 | 5 |
| Costs by State (in millions) | | | | |
| Idaho | $63.49 | $95.24 | $126.99 | $158.73 |
| Illinois | $761.62 | $1,142.43 | $1,523.24 | $1,904.05 |
| Indiana | $303.28 | $454.93 | $606.57 | $758.21 |
| **Iowa** | **$158.06** | **$237.10** | **$316.13** | **$395.16** |
| **Kansas** | **$150.63** | **$225.95** | **$301.26** | **$376.58** |
| Kentucky | $182.40 | $273.61 | $364.81 | $456.01 |
| Louisiana | $174.95 | $262.43 | $349.90 | $437.38 |
| Maine | $74.03 | $111.05 | $148.07 | $185.08 |
| Maryland | $489.73 | $734.60 | $979.47 | $1,224.33 |
| **Massachusetts** | **$1,287.68** | **$1,931.51** | **$2,575.35** | **$3,219.19** |
| Michigan | $882.43 | $1,323.65 | $1,764.86 | $2,206.08 |
| **Minnesota** | **$446.49** | **$669.74** | **$892.98** | **$1,116.23** |
| Mississippi | $95.09 | $142.64 | $190.19 | $237.74 |
| Missouri | $347.02 | $520.53 | $694.04 | $867.55 |
| Montana | $44.95 | $67.42 | $89.90 | $112.37 |
| Nebraska | $99.64 | $149.47 | $199.29 | $249.11 |
| Nevada | $116.41 | $174.61 | $232.82 | $291.02 |
| New Hampshire | $164.13 | $246.20 | $328.27 | $410.33 |
| New Jersey | $2,134.27 | $3,201.40 | $4,268.53 | $5,335.66 |
| New Mexico | $79.92 | $119.88 | $159.84 | $199.79 |
| New York | $1,185.41 | $1,778.11 | $2,370.81 | $2,963.52 |
| North Carolina | $473.37 | $710.05 | $946.73 | $1,183.41 |
| North Dakota | $43.43 | $65.14 | $86.85 | $108.57 |
| Ohio | $611.80 | $917.70 | $1,223.60 | $1,529.50 |
| Oklahoma | $142.83 | $214.25 | $285.67 | $357.08 |
| Oregon | $265.15 | $397.73 | $530.31 | $662.88 |
| Pennsylvania | $929.01 | $1,393.51 | $1,858.02 | $2,322.52 |
| Rhode Island | $58.91 | $88.37 | $117.82 | $147.28 |
| South Carolina | $187.80 | $281.71 | $375.61 | $469.51 |
| South Dakota | $35.34 | $53.01 | $70.68 | $88.35 |
| Tennessee | $1,451.31 | $2,176.96 | $2,902.62 | $3,628.27 |
| Texas | $1,706.90 | $2,560.34 | $3,413.79 | $4,267.24 |
| **Utah** | **$295.02** | **$442.53** | **$590.03** | **$737.54** |
| Vermont | $77.22 | $115.84 | $154.45 | $193.06 |
| Virginia | $545.36 | $818.04 | $1,090.72 | $1,363.40 |
| Washington | $728.07 | $1,092.10 | $1,456.13 | $1,820.16 |
| **West Virginia** | **$77.01** | **$115.52** | **$154.03** | **$192.53** |
| Wisconsin | $337.76 | $506.64 | $675.53 | $844.41 |
| Wyoming | $27.25 | $40.88 | $54.51 | $68.13 |
| **Costs to non-settling states** | **$8,475.05** | **$12,712.57** | **$16,950.09** | **$21,187.61** |