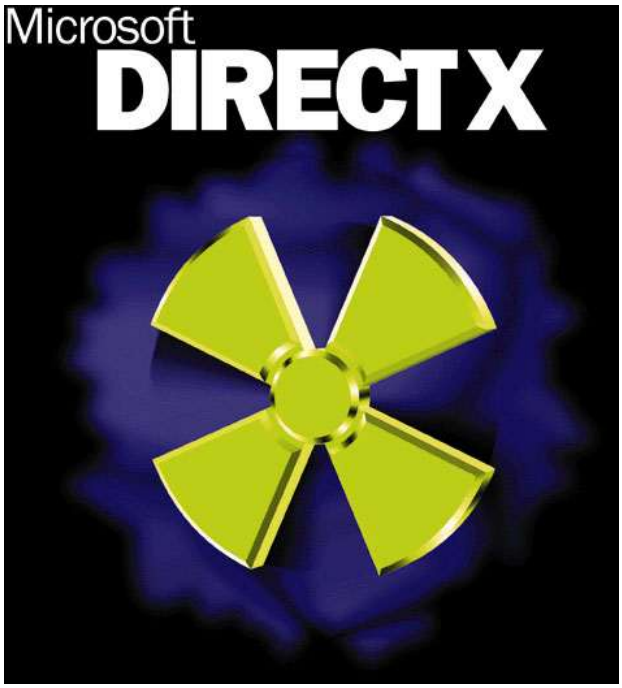
Talisman

GRAPHICS AND MULTIMEDIA SYSTEM

DirectX and Talisman Update



May '97

Jay Torborg	jaytor	Director of Windows Multimedia
Jim Veres	jimveres	Talisman Group Manager
Larry Ockene	lockene	Talisman Software Manager
Bob Gaines	bgaines	Talisman ISV Program Manager
Mark Kenworthy	kenworth	DirectX Program Manager

Executive Summary

THE ENTERTAINMENT PC

Multimedia PC, always on, zero admin, connected PC, sealed case, Internet telephone, digital TV, . . . the tag lines go on. All of these efforts, and more, are working to expand the market for PCs and to place the PC at the center of a digital entertainment system. The Windows Multimedia Group provides the 2D and 3D technology to make the Windows PC the best, most desirable platform for running real-time synthetic 2D and 3D entertainment titles. We accomplish this by providing, as part of Windows, the technically best, most supported API (DirectX). Talisman provides compelling new multimedia technology (HW and SW) that will make future DirectX Windows accelerator cards the best at running these titles.

THE GAME BUSINESS

Game box vendors (Sega, Nintendo) use the time-honored marketing strategy of providing the platform (razor, Barbie doll, Nintendo 64) at or below cost so they can enable a large profitable business in consumables (razor blades, Barbie clothes, game titles). These vendors fully realize that compelling titles are needed at announcement to jump-start a new platform. They put major content programs in-place for each new system. Mario 64 is a recent example.

The PC has a different, harder to jump start model where different companies provide the hardware and the software. All of these companies expect to make money. This causes the evolution of multimedia on PC to be slower that it could be. IHVs focus on keeping their product costs down, on supporting the features that ISVs have traditionally used, and slowly add new capabilities to give them some product differentiation. ISVs code to the widest audience thereby avoiding the latest features until they become pervasive.

While there are exceptions (Myst, Doom, Quake, Warcraft, Duke Nukem,and Flight Simulator being the most obvious), many PC game titles have a shelf life of only a few months. Most game authors target their titles to run on the machines that have the largest installed base during this brief period. This year the more aggressive game authors are targeting a low-end to mid-range Pentium with a 2D Windows accelerator card. 3D hardware vendors try to *"prime the pump"* by paying authors to create optimized versions of their games and then bundling these titles with their cards. The IHV's goal is to make their card pervasive enough for ISVs to take advantage of their HW without having to pay them to do it. Next year we expect game authors to target basic 3D accelerator cards without financial encouragement.

DIRECTX AND TALISMAN STATUS

Widespread Support for DirectX

DirectX achieved universal support this past year in the Windows accelerator card market. All hardware vendors are creating and shipping DirectX drivers. This positions DirectX as the API of choice by game authors that are trying to reach the largest installed base. Virtually all Windows titles being developed this year are being developed with DirectX.

This widespread adoption of DirectX with hardware vendors gives DirectX one year to address its ease of use, stability and performance issues and become the content author's preferred API. DirectX 5, in beta now, is tightly focused at addressing these issues.

Top graphic chip vendors license Talisman

In the eight months since Talisman was first disclosed it has garnered a major awareness in the press, at IHVs, at OEMs, and is beginning to have awareness with game authors. OEMs are asking IHVs about their Talisman plans (this caused S3 to join up). Every technical presentation has been met with positive responses. Talisman has been licensed by, and programs are in place, at all of the top five graphics chip vendors (S3, ATI, Matrox, Trident, Cirrus).[1] Intel likes the technology but doesn't like our licensing terms (Windows only), or playing second fiddle on graphics technology.

DirectX challenges and plan

DirectX is not without its share of criticism and lots of this criticism is public. Beyond the general criticisms of setup/drivers, documentation, and ISV support, these criticisms have primarily been focused on the performance and ease of use of Direct3D. Discussions with game developers and analysis of their code has convinced us that the two problems are related. Content authors had a hard time understanding and using the *"execute buffer"* model of Direct3D 3.0 where a series of triangles are batched and passed into Direct3D in a single API call. Some ISVs used execute buffers in a highly unoptimized fashion by passing a single triangle per execute buffer. This resulted in incurring all of the overhead of execute buffers and none of the benefits. DirectX 5, in beta now, adds a "*draw primitive"* API that is much easier to use, and provides much better performance when authors just want to pass individual triangles or small meshes into Direct3D. Early results show a dramatic improvement on titles that were using execute buffers poorly.

DirectX is positioned to win the PC game API wars due to its universal adoption by the hardware community. The only real competition is in the 3D space, from OpenGL and IHV proprietary APIs like 3Dfx's Glide. Glide is quickly losing momentum as other IHVs catch up to 3Dfx's performance and as we solve the problems discussed above with D3D.

OpenGL is stable and mature, and is well supported in the high-end workstation market. Its stability, however, also means that it does not evolve rapidly to take advantage of hardware innovation. OpenGL does have some technical advantages over D3D (most notably metafiling, printing, and software failover) that we plan to incorporate into D3D in DirectX 6.

Our continued support and enhancement of OpenGL as a Microsoft API has been confusing for our ISV and IHV partners, as well as our internal teams. We now have a proposed plan to integrate the two 3D teams and leverage the best of both APIs as part of our DirectX 6 plans. A separate memo, attached in section 3 following the executive summary discusses these plans. Our intent with this effort is to make DirectX and Direct3D specifically, the primary API for all graphics and multimedia applications including professional and CAD applications.

DirectX has, until now, been driving hard to deliver base functionality. DirectX 5 has largely caught up with IHV hardware functionality. To continue DirectX momentum and continue to have widespread support of IHVs and ISVs we need to push on (in order of importance):

- Universal support by IHVs. While we have this now, we are still in the mode of following the hardware community with feature support. With full Talisman support in DirectX 6, Microsoft can be more proactive in driving new features.
- Ease of use. More example code, substituting one call for many, listening to ISVs and responding. Provide full software fail-over for missing hardware features. Reduce the number of permutations of HW accelerated vs. non-accelerated features by getting IHVs to accelerate the full set of important features through PC 9x standards and branding.
- Highest performance. DirectX will allow us to focus on what the ISV is trying to create, not just pushing triangles. This will allow us to add value to our APIs, not just stay out of the way of the hardware.

[1] S3 has announced that their next generation part will contain Talisman technology. We're in final negotiation on licensing terms with S3 and expect to have a signed agreement before the end of May.

- Product maturity. Enhance the test suite; focus on the stability of each release. Extend beta cycles by moving our release dates back in the calendar year to be more in sync with the ISV development cycle for Christmas titles.
- Advanced Features. Always first with support for advanced features.

Talisman challenges and plan

Our corporate silence on Talisman in the press and the lack of a strong Talisman content program has allowed vendors with proprietary non-Talisman solutions to create a lot of FUD around Talisman.

Widespread Talisman adoption is well underway but not yet assured. In order for Talisman to succeed we need:

- Finish development of the technology. This includes the HW technology; the tools and example content needed for ISVs to create Talisman optimized titles and support for the features in DirectX. This development is well underway and relatively risk free. The HW technology was developed and is being licensed. We've shipped our first DDK and SDK. DirectX 6 will include full support for Talisman (some support is provided in DirectX 5).
- Pervasive installed base of Talisman enabled cards. The path to this is the incorporation of Talisman features by the important IHVs. We're on-track to achieve this with incorporation of many Talisman features by the leading vendors in '98 and full support in '99 if we solve the ISV issue below.
- Titles that take advantage of Talisman features. This is the part of the program that we're under-investing in and could delay or kill the widespread adoption of Talisman technology. As mentioned earlier, game vendors code to the widest audience and, of course, there will be no installed base of Talisman hardware when it *first* ships. Compelling Talisman optimized content is needed when these Talisman accelerator cards first ship to jump-start the industry and get the positive reinforcement cycle going. Talisman needs its Mario 64.

PROPOSED ACCELERATION PLAN

We can improve the marketing and development of both DirectX and Talisman. The following proposed actions should start now and run through the end of the year.

- Increase our marketing efforts behind DirectX.
  - End the confusion in the industry surrounding our positioning of OpenGL vs. DirectX. Make it clear that OpenGL 1.1 is offered and supported for running traditional OpenGL applications. Make it clear that DirectX is Microsoft's mainstream multimedia API. Major R&D efforts are directed at making DirectX the best, most pervasive, highest performance API for all graphics and multimedia applications.
  - Evangelize Microsoft Entertainment Business Unit to use DirectX for all new titles and DirectX with Talisman enhancements for titles targeted for Christmas '98 and beyond.
  - Create a logo program to create brand identity and provide leverage on IHVs to incorporate a consistent set of DirectX acceleration so ISVs can depend on these features being accelerated.
  - Create a co-marketing campaign for DirectX ISVs. Fund Christmas '97 and '98 advertising highlighting the Windows PC and compelling titles as the premier game machine. This is a particularly good year to start this push as there will be no new consumer game machine to capture media and consumer attention.

- Prime the pump for Talisman enabled titles.
  - Have the Microsoft Entertainment Business Unit develop and market 2 to 3 "killer" Talisman optimized titles for the Christmas '98 selling season to accelerate the market acceptance (create consumer pull). These titles would still be usable on lower end systems but only at today's quality and performance levels. Game development is a "hit" market, so it is risky to bet on one title being a winner. Most (if not

all) Talisman hardware vendors will bundle these titles with their products so it should be relatively easy to recoup the investment with profit.

- Currently working with vendor Holy Grail (publisher Playmates) on a Talisman showcase title called Flying Tigers. Fund Holy Grail and two to three more to produce Talisman optimized titles targeted at Chrismas '98. Deliverables here would also include screen shots and demo clips to build media attention at Comdex '97 and WinHEC '98.
- Create showcase Talisman titles out of Microsoft hit titles (Flight Simulator, 3D Baseball, Monster Truck Madness). This will breathe new life into these titles which are more assured of market success; especially after being brought up to Talisman performance and quality levels.

- Support IHVs incorporating DirectX Talisman technology into their products.
  - Co-marketing campaign for DirectX Talisman chip/board vendors. Match funds for chip/board/system companies producing ads that highlight Windows PC as a premier entertainment system showcasing Windows, DirectX, and Talisman acceleration and their hardware.

The above actions are designed to capitalize on the lack of new hardware in 1997 in the game console business to focus attention on the Windows PC as the premier game system. In '97 we expect the PC to pull ahead of the Nintendo 64 on all aspects (except cost). 3D hardware, storage capacity, (both CD-ROM and memory capacity), processing capability, multi-player connectivity (networks and modems), input devices, and the screen will all be superior. We should capitalize on this with an aggressive marketing campaign to position the PC as the top tier for running entertainment titles. 1998 will be an even more impressive year with base 3D cards getting much better and the first Talisman cards coming to market.

Jay Torborg 3D Stragegy Memo

Jay Torborg, Director of Windows Multimedia, authored the following memo in late April describing the current dual 3D API situation and proposing a strategy for unifying Microsoft's two current 3D APIs (OpenGL and Direct3D).

Over the past year or so, we have struggled with having two different 3D APIs. Despite our attempts to position them into different market segments, the obvious overlap in these two products has resulted in a number of problems including:

Confused message to ISVs and IHVs – ISVs don't know which API to use, and IHVs cannot optimize their designs for a single API.

Poor morale on the teams – neither team feels as if they are getting the management support and recognition for their work that they feel they deserve. Both teams are spending way too much of their time defending their product relative to the other.

Inefficient use of resources – the side effect being insufficient resources to invest in significant innovation.

With the reorganization of multimedia (so that the DirectX and OGL teams have a closely coupled management structure), we have had the opportunity to attempt to resolve these problems. We all believe that a single 3D API suitable for all our markets (games through workstation applications) is the optimal solution.

Strategic Assumptions

Our single 3D API needs to be Microsoft controlled so that we can drive innovation on our platform. We obviously need to provide full legacy compatibility with DirectX 5 D3D and OpenGL 1.1, but this can be in the form of separate interfaces if necessary. We need to make some effort to minimize the effort required of current D3D developers to migrate to the new API, but as I will discuss later, this is probably not a terribly significant issue. What is most important is that this API has key Microsoft proprietary innovations that make it compelling for developers to use, and that it have the necessary attributes (documentation, robustness, scalability, good driver coverage, etc.) to be effective for the broad market.

The availability of this new API is also key. The games market is driven by significant Christmas season volume. It's therefore key that we introduce this new API next spring so that we don't go another year with this two API confusion.

Market Dynamics

I want to include a brief discussion about the market dynamics with regard to our two APIs so that you can get a better understanding of why we have made the decision we have.

Direct3D has been in the market for less than a year, and certainly the early feedback was not particularly strong. But the success of DirectDraw in the games market and the availability of a broad range of hardware, some of which has been optimized for Direct3D, have started to improve its acceptance. With DirectX 5, we are addressing D3D's major weaknesses for the games market, but with the resources we have focused on this program, we have still not delivered all the features supported by the latest 3D hardware. We do, however, expect to see 50-75 titles (possibly more) come out this year using D3D.

Much of the momentum behind D3D is in the branding, not the specifics of the API, nor in its technical merits. In fact, we have made a fairly significant change in the DX5 version of D3D to overcome one of the key weaknesses in DX3. The D3D ISVs are able to accommodate fairly significant changes to the API because D3D applications tend to be quick to develop (12-15 months) and are short lived (less than one year). There is limited long-term investment in the code base.

D3D provides very specific guidelines for hardware vendors to develop efficient 3D accelerators for the API. Several key 3D graphics chip vendors (at least five) have implemented hardware interfaces that are optimized for the D3D vertex list structure defined by the D3D driver. Unlike the ISVs, it is somewhat more difficult for IHVs to adopt a new vertex structure, so we need to be sensitive to this issue.

OpenGL is the de facto industry standard for technical markets. Unlike the typical D3D ISV, most of the OpenGL ISVs have made significant long-term investments in the OGL APIs and expect to leverage their code base for many years to come. OpenGL has a reputation as a mature API, and is very stable. The corollary to this is that the API is slow to leverage innovation. It's also an API that Microsoft has limited direct control of, and virtually no market perception of leadership on.

Development Alternatives

It doesn't make sense to start totally from scratch in the creation of this new API, so we really have two development alternatives – start with the D3D code base or the OGL code base.

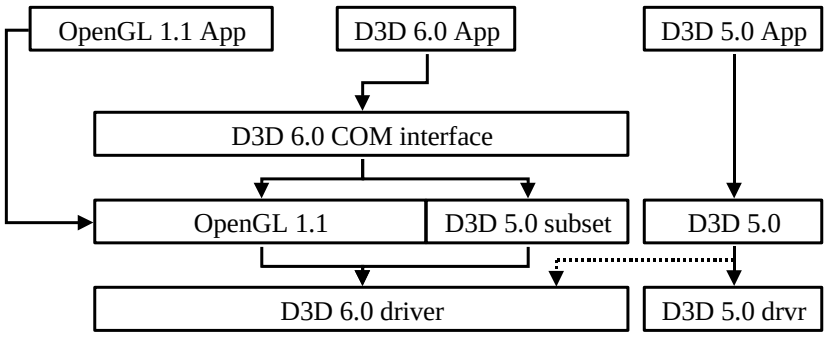
The D3D code base could be extended to support the needs of business and technical 3D applications. The primary features required for this are support for windowing (D3D cannot share the hardware backbuffer or Z-buffer between multiple windows), support for metafiles and printing, and full software failover (emulation of all API features). There are also a number of minor features that would be desired to support easy migration of existing technical applications. Estimates to complete this work vary, but it's unlikely that it could be done in less than two years. This would severely limit the amount of innovation we could introduce during this time.

The second alternative is to start with the OpenGL code base and extend it as necessary to provide simple migration of D3D applications and add our own innovations for advanced rendering concepts (such as many of the features enabled by Talisman). The OpenGL code base already provides the features discussed above that would need to be added to D3D.

The OGL code base already provides a reasonable set of features to address our broad market needs. In this case, the development effort would be focused on integrating D3D features, look and feel, and introducing a number of new features. We don't yet have a detailed development plan, but feel reasonably confident that a solid API could be developed based on this code base which could RTM next Spring.

3D Strategy Proposal

Our current strategy is therefore to build a new COM based D3D API which we will ship as part of DirectX 6 next Spring. We will start with the OGL 1.1 code base as the core engine of this API, and will augment this code base with code/functionality from DX 5 D3D and Talisman, as well as new innovative features. This will be a new API which will not be fully compatible with DX5 D3D or OGL 1.1, but will be a clean solid foundation to build from. We recognize that this API must have significant Microsoft added value, and must not be just a warmed over OGL API.



A new D3D driver model will be created to support the new API. This driver model will leverage the IHV investment in the D3D TLVertex structure as much as is practical and will be designed to allow existing drivers to be ported with as little effort as possible.

Legacy support for OGL 1.1 will be provided by exposing the existing procedural interfaces from the OGL code base. We do not expect to continue to enhance the OGL procedural interfaces. ISVs will need to migrate to the new D3D COM interface to leverage new innovations and integration with DirectX.

DirectX 5 legacy support will be provided by including the DX5 DLLs. While this results in code bloat, we do not see any practical alternatives. The existing DirectX 5 D3D drivers will be used with these DLLs initially. At some point in the future, we will modify the DirectX 5 code base to work with the new DirectX 6 D3D driver model so that a single driver will work with both legacy and new applications. This is represented by the dotted line in the diagram above.

Short Term

In the short term, we plan to ship DirectX 5 D3D in Memphis as planned. We will also ship OGL 1.1 with Memphis but will not provide any proprietary extensions. We will ship support for the MCD (mini-client driver) with Memphis so that we can encourage IHVs that want to provide OGL support to use the MCD model instead of the ICD model.

This plan will allow us to focus virtually all our 3D development resources on the new integrated DirectX 6 D3D API.

Ownership

We believe that this proposal meets our strategic objectives. While we will be starting with the OpenGL code base, we do not intend to develop a warmed-over OpenGL. We intend to add value through the COM wrapping, D3D 5.0 and Talisman enhancements and longer term innovations. Our existing relationships with strategic technical vendors such as Intergraph, DEC, and HP will help us move the technical ISV community to our new API.

Issues

There are a few issues that result from this strategy. The DirectX Media group has a number of dependencies on the current D3D API. While they can continue to use the DirectX 5 interfaces, all new innovation will be on the new API. To the degree that DirectX Media chooses to take advantage of this innovation, this will require some development effort on the part of the DirectX Media team.

As you might imagine, the internal debates as to the direction to take with our 3D strategy have been heated and emotional. While we will do everything we can to engage all members of the two development teams in this new strategy, I expect that some of these individuals will be unhappy with this decision and may choose to leave the team in frustration. Unfortunately, many members of the team are critical for short term deliverables. We need to make sure that these developers stay focused on completing these deliverables.

We also need to ensure that at the executive level we are in agreement with this plan to minimize the amount of energy spent on continuing the debate with other groups at Microsoft.

Mike Abrash 3D Strategy Memo

Mike Abrash, former developer of Quake at Id, current Microsoft researcher, authored the following memo in late April proposing a unification strategy for Microsoft's two current 3D APIs (OpenGL and DirectX).

Summary

The best way for Microsoft to quickly get to a single 3D API that covers the spectrum of 3D apps from games to workstations, and that can serve as a foundation for future innovation, is to work from the OpenGL code base. The real question is how to do this in a way that best meets Microsoft's strategic needs.

Microsoft needs one 3D API that can address all Win32 market segments, from games up to workstation 3D; two years of resource and marketing problems make that clear. More, it needs to get to this point quickly, both to stabilize the 3D industry after years of multiple APIs and constant redesign, and to put the foundation in place so that real 3D innovation on Win32 can commence.

Microsoft currently has two code bases that are possibilities, D3D and OpenGL. OpenGL is capable of addressing the needs of the entire market on all Win32 platforms for Christmas of 1998 at the latest, with minor modifications and evangelism to gain broad driver support. D3D is capable of meeting most of the gaming community's needs for Christmas of 1997, and all of those needs by Christmas of 1998. However, it is not capable of meeting the needs of the workstation market until at least 2000 and more likely later, and even then there will be problems with getting high-end ISVs to port to D3D and workstation OEMs to switch to it.

To put it bluntly, D3D has no compelling technical advantages and many disadvantages relative to OpenGL. If the both APIs were Microsoft-proprietary and a choice had to be made between the two, it would be no contest. The only significant advantage D3D has currently is broader driver coverage, but many IHVs (ATI, 3Dfx, Rendition, Intel) are doing OpenGL drivers even without any Microsoft evangelism, and the OpenGL driver issue is certainly solvable for Christmas of 1998 across Win32—and with a strong effort could be addressed even for Chrismas of 1997. A few other details remain, such as app control over texture management and BeginScene/EndScene for Talisman, but the overall picture is clearly that D3D is almost entirely a subset of OpenGL, and lacks its broad, well-tested feature set, documentation, code maturity, finished implementation (metafiling, printing, full software emulation) and scalability, and has no real technical justification. Consequently, if the goal is a single API, the OpenGL code base has at least a 2-3 year headstart.

The fastest, least-expensive solution, and one that is technically sound, would be just to declare OpenGL the Win32 3D API. However, since it's probably not acceptable strategically for Microsoft to have an open Win32 3D API, there are several other possibilities (all involving calling the result D3D), including:

- simply innovating independently, using the name D3D instead of OpenGL, and not following the evolution of the OpenGL spec;
- putting a COM wrapper API around OpenGL along with making straightforward improvements;
- extensively reworking the code and the API.

The choice among these is tactical, however, mostly having to do with time and resource constraints and with marketing issues. The key points are these: If D3D is the code base that Microsoft takes into the future, OpenGL will still have to be kept around for years, or NT will lose its standing as a 3D workstation. Also, using the D3D code base would amount to Microsoft taking a three-year step backward in terms of functionality, documentation, and maturity in the 3D arena. The effort expended on catching up in those areas would greatly reduce Microsoft's ability to innovate in 3D over the next few years. The only justification put forth for paying this huge cost is the desire for a proprietary API, but this can be addressed at the cost of going sideways for less than a year while the OpenGL code is revamped, legacy D3D support added, and the new marketing/evangelism story rolled out, which seems far preferable to a multi-year step backwards.

To put it another way, we have an API that does what we need but isn't proprietary, and one that doesn't do what we need but is proprietary. We can either spend a modest amount of time and resources making OpenGL proprietary, or a huge amount of time and resources making D3D adequate. Either way, we get a single proprietary API, but at very different costs.

In short, by far the least costly approach, and the one that would take Microsoft farthest into the future and let it innovate soonest, is to work from the OpenGL code base. It is hard to come to any other conclusion; the only real question is how best to innovate on the OpenGL code base in a proprietary fashion, and how to dig out from the current situation that has resulted from two years of dual APIs, and constant change to and heavy evangelism of D3D.

If for any reason that approach isn't feasible, the best alternative would be to maintain OpenGL in ICD form only, restricting its use to high-end workstations, and innovate on D3D, with the intention of evolving D3D into a workstation-class API as quickly as possible. This would serve the needs of workstations and games for the next few years, and would certainly be proprietary. This doesn't eliminate the dual-API problem; moreover, OpenGL ICDs can be done for game-class hardware without Microsoft's involvement, and all Quake-engine-based games will run on OpenGL; in fact, we'd take a great deal of heat for restricting OpenGL support because, due to GLQuake, GL support has become a required graphics adapter feature. Also, it requires a great deal of work just to get back to where OpenGL already is, and it doesn't guarantee that high-end ISVs would port their apps from OpenGL, partly because it turns out that many workstation 3D apps, such as Softimage, aren't very portable.

Notes:

All OpenGL-based approaches would also require full legacy support for current D3D functionality, and for the current OpenGL 1.1 API. This can be provided via the D3D DLL, but a common driver model will be needed within a year to avoid the current requirement for separate D3D and OpenGL drivers.

All OpenGL-based approaches depend on Microsoft having the legal right to take the OpenGL code base in a proprietary direction; jbal has researched this and says it is not a problem, but additional confirmation from Legal is essential.

John Latta Report

John Latta is president of 4th WAVE an analyst organization focusing on the PC 3D graphics market. His executive summary on the current status of Talisman and his recommendations are reproduced below.

Media PC and Talisman

Microsoft is poised to have a significant impact on 3D for Windows through its Talisman research. The next 6 months are critical. The Talisman licensees will have first silicon by Q1 1998. Yet, without games which take advantage of the technology the net effect of the research will not be realized or significantly delayed. Funding should be provided to develop at least two games for Q4 1998. We see Talisman game development as being a logical continuation of Microsoft's research, except now the focus is on using the technology with the best in the game development community so that its adoption moves into mainstream PC market. Besides seeking to support near term efforts to commercialize Talisman there are other equally critical issues surrounding 3D, APIs and the PC 3D platform. Thus, the urgency of getting these titles in place is in context with these issues and the role Microsoft will play.

Forces are at play which could undermine Microsoft's role in 3D and its impact within Windows. SGI is positioning OpenGL as a games API and Intel is seeking to set the platform standards for 3D. In spite of the fact that AGP brings the potential for significant performance improvement Intel's 3D objectives are quite different than Microsoft's. With the 740 2D & 3D graphics accelerator Intel is seeking to increase its dominance of the PC platform. Optimizing the platform performance and the definition/evolution of the API are inextricably linked. At the recent Computer Game Developers Conference the Pentium L1 cache was shown to play a critical role in limiting the polygon rate of 3D accelerators and it was suggested that the API could be modified to overcome this limitation. A prospect which we consider absurd – why should an API compensate for a microprocessor limitation when doing geometry and lighting? 3D is also at the center of the continuing debate – How can innovation in 3D be fostered while having standards which game and application developers can target for delivery across the broadest possible set of systems? Microsoft's role in defining the platform which has the greatest prospects for innovation is being threatened. Intel's recognition of the role of 3D and their domination of the platform will increasingly overlap into API's and the ability of others to innovate. Microsoft cannot afford to relinquish its position in defining how the Media PC and, in particular, 3D is integrated into Windows.

The existing 3D real time industry has evolved over 25+ years. The key industry players today are Lockheed-Martin's Real 3D subsidiary, who is partnered with Intel, and Evans and Sutherland, who is partnered with Mitsubishi. These companies represent the establishment based on 3D technology which renders each scene every frame. However, this technology, also used by SGI, is severely limited in its reliance on large memory bandwidths. In fact, the pace of 3D developments on the PC today is largely driven by the ability of companies to extract the maximum bandwidth from existing memory technologies at the lowest price. Talisman represents a major break from this past 3D tradition. Microsoft's Talisman research not only increases the image quality significantly but reduces the corresponding memory bandwidth requirements over more traditional methods. Talisman represents exactly the type of innovation that Microsoft wants to encourage throughout the industry. Microsoft must carry that research to the market in silicon and content. Its leadership role implies nothing less.

Just as Microsoft has taken a leadership role in PC hardware, especially with PC97 and PC98, the same applies with Talisman. Although some may argue that Microsoft should not be in the silicon business we counter that in 3D this is essential due to the close coupling of 3D with API and the platform. There is a close parallel with Microsoft's ventures in interactive media and MSNBC – media industry participation narrowly focused on the company's objectives is essential. The same criterion applies to 3D. Talisman brings innovation which benefits the whole PC industry while Microsoft defines a new level of quality and performance. This is certainly in synergy with MSNBC.

Microsoft must execute quickly in three areas: complete the transition of Talisman from R&D into shipping products including content, take an aggressive stand on defining 3D APIs and drivers so that broad innovation can be supported and invest in evangelism and marketing to identify Microsoft as the major 3D player on the PC and Windows. Microsoft should invest $2.2m in title development, in conjunction with IMD for at least 2 titles. $1m should be spent on a marketing campaign in conjunction with Talisman licensees to actively promote their products, the technology, and the role of Microsoft in setting the performance envelope for the Media PC. A logical investment by the leader to bring its research investment to mass markets and to drive the PC deeper into the home market.

Talisman Program Update – more detail

OVERVIEW

The Talisman program consists of three efforts:

1. Develop the Talisman technology (hardware and software).
2. Achieve widespread adoption of the technology in the hardware industry.
3. Get compelling content in-place to demonstrate the capability of the technology.

Development of the basic Talisman technology is proceeding well. The first Talisman SDK and DDK shipped just prior to WinHEC with new shipments planned for every quarter over the next year. Support for a subset of the Talisman features is included in DirectX 5, in beta now, with full support committed in DirectX 6.

Adoption of the technology into the IHV community is proceeding well. Licensing status includes:

Have licensed or are in final negotiation on terms	In licensing discussions with	Do not plan to license Talisman
ATI	**NEC/Video Logic**	**3Dfx Interactive**
Matrox	**Hitachi**	**Rendition**
Number 9	**IDT**	
S3	**Chromatics**	
Trident	**nVidia**	
Cirrus Logic	**Intel**	
Fujitsu		
Equator / Hitachi		
Philips		
SGS-Thomson		

Intel is a special case. They have expressed strong interest in incorporating many of the technologies rolled out in Talisman but don't want to live with a "for use with Windows only" restriction and they don't want to call the technology Talisman as that would position them as following Microsoft in graphics technology. We continue to put effort into resolving these issues.

Getting content in-place continues to be critical to the success of Talisman and is not proceeding at the pace that we would like. Our focus to date has been to complete the set of tools needed by an ISV to create Talisman enabled or optimized content and evangelizing ISV to create Talisman optimized content for Christmas '98. ***It is clear to us that it will take more that just evangelism to get three to five Talisman showcase titles ready for WinHEC '98.***

We continue to be very satisfied with the positive response from both the hardware and software community to the technology embodied in Talisman.

ESCALANTE REFERENCE DESIGN

Escalante was a reference design for a PCI add-in card containing a full 2D and 3D graphics subsystem, audio, video, and new serial bus interfaces. Target cost was $ 250 to 270 resulting in a product cost in the $ 450 to 500 range.

A decision was made in April to discontinue development of the Escalante reference card. Cirrus Logic had been struggling to get the Tiler and Sprite chips through chip layout. Their efforts were leading down a path to produce Escalante chips that were more expensive that our design target, over a year late and only achieve 75% of the projected performance. Escalante development had three goals:

1. Take Talisman technology from a research concept to technology that could be deployed. Test the concepts and produce a design kit to enable IHVs to quickly adopt the technology into their parts.
2. Develop a product concept to use for IHV and ISV evangelism.
3. Develop a card that could be used to develop game content during 1997 so the content could be in-place when mass-market cards came to market in the first half of 1998.

We achieved the first two of the program goals. The Escalante development program allowed us to perform virtually all of the development work needed to move the Talisman technology out of research and through development. IHVs and system OEMs have responded well to the technology, but not well to the price of the Escalante card. We missed the last goal as it took longer than we expected to complete the design of the Tiler and Sprite chips, and it took Cirrus much longer than we expected to get the chips synthesized to gates and through layout. With the financial and staffing issues at Cirrus, it became obvious that they were not going to achieve the goal of getting these chips finished and demonstrable by Comdex this year. This put the schedule for Escalante within months of when the first mass-market chips were expected. Implementations from S3, ATI, Matrox, Equator/Hitachi and our second reference effort (Wizard) all were planning to come to market soon after Escalante with dramatically lower costs and superior performance. With these chips available at the same time as Escalante, it makes more sense to perform title development on the mass-market chips instead of Escalante.

There were two primary lessons learned from the Escalante effort.

1. Only work with a single partner on a program of this complexity. Cross-company coordination and information flow was not efficient enough to result in minimal schedules. A major amount of our time and energy was spent communicating with the four other companies participating in Escalante. While communication flow was good between each company and us, it never was excellent between the other companies. Too much of the communication had to go through us.
2. Don't split chip design across two companies. On Escalante, Microsoft did the basic chip architecture and design, Silicon Engineering reduced that design to Verilog code, and Cirrus performed the layout. The Verilog code out of SEI needed a fair amount of changes to meet timing and space constraints imposed by the layout. Having these design tradeoffs go between companies again delayed the program.

Two of the goals for Escalante were achieved. The technology has been developed and is in-use by major chip vendors. Escalante did not achieve its time to market goal and thereby could not be used to get content in-place. That role will fall to the first mass-market chips coming out at the end of this year.

WIZARD PROGRAM

Wizard is a program to produce a mass-market Talisman graphics chip faithful to all of the Talisman concepts. The effort has been underway for about six months now with Fujitsu being the primary implementation partner. We expect them to see first silicon about a year from now. Goals for Wizard are to achieve about three times the 3D rendering performance as Escalante for about one fifth the street price.

- 100 M pixels/second 3D render rate
- 200 M pixels/second 2D affine rate
- AGP connected single chip doing 3D render and 2D affine warp and composite time sequentially
- Support for all Talisman features (antialiasing, anisotropic texture filtering, multi-pass rendering, etc)
- Mass-market pricing (under $ 100 price with 4 MB of graphics memory)

Microsoft support for this effort is focused on helping Fujitsu modify the Escalante C-Model to become Wizard, generating the Windows and DirectX driver and modifying the Escalante test cases to verify the Wizard design.

The Intel architecture labs were instrumental in getting the Wizard program started. They participated in the Wizard program for about 5 months before discontinuing pending resolution of licensing issues. I believe they had little interest in Wizard as a product, and instead, were interested in fully understanding the Talisman technology. Participation in Wizard allowed them to achieve this goal. The Intel systems group continues to be very supportive of the Wizard program to Fujitsu and has indicated strong interest if Talisman optimized titles get in place and they meet their performance and cost goals.

TECHNOLOGY ADOPTION

As shown above, all six of the top six graphic chip IHVs have licensed Talisman and are incorporating features into their '98 products. There are two significant holdouts; 3DFX and Rendition. 3DFX and Rendition feel that they are at the top of the heap in current 3D technology and don't want to change their direction. 3Dfx has also recently filed to go public. As such they need to position themselves as having a valuable market position and proprietary IP.

Talisman Feature	Companies Implementing
Chunking	S3, Matrox, Trident, Fujitsu, Cirrus
Antialiasing	S3, Matrox, Trident, Fujitsu, Cirrus
Anisotropic Texture Filtering	S3[2], Matrox[3], ATI[3], Trident, Fujitsu, Cirrus
32 bpp color	S3, Matrox, Trident, Fujitsu, Cirrus
Texture Compression	S3, Matrox, ATI, Trident, Fujitsu, Cirrus
Compositing of Image Layers	S3, Matrox, ATI, Trident, Fujitsu, Cirrus
Affine Warping of Image Layers	Matrox, ATI, Trident, Fujitsu, Cirrus
Scene update at monitor refresh rate	Matrox, ATI, Fujitsu, Cirrus

[2] S3 uses a different method than we specified to achieve anisotropic texture filtering. They are generating images now to ensure that the quality of their method matches that specified by Talisman.

[3] From one mip level only.

COMPARISON WITH TRADITIONAL ARCHITECTURES

The existing 3Dfx chip set, the new Intel Auburn chip and four Talisman chips are compared. The 3Dfx chip just started shipping in the first quarter. First silicon for the Intel chip is expected this month. The Talisman chips from ATI and Matrox are expected to have first silicon at year-end. The Fujitsu Wizard chip and the Trident chip are both expected to have first silicon about a year from now. All of these parts have about the same cost with 3Dfx being the most expensive.

The 16-bpp pixel rate shown below are a function of the chip's schedule. The chart below shows that the traditional chips stop far short from achieving interesting quality levels. Both the 3Dfx part and the new Intel part only support 16-bit color in their frame buffers. Neither offers antialiasing when doing texturing. The Intel part does not support even trilinear texture filtering. The 3Dfx part does, but at lower performance.

As can be seen from the feature list for the Talisman chips, achieving high performance at high quality is much easier with Talisman technology.

Feature	3Dfx (Non-Talisman)	Intel Auburn (Non-Talisman)	Fujitsu Wizard Full Talisman	Trident Full Talisman	Matrox Partial Talisman	S3 Virge-2 Partial Talisman	ATI Rage NT Partial Talisman
16 bpp Pixel Rate	45 Mps	66 Mps	200 Mps	200 Mps	200 Mps	170 Mps	100 Mps
32 bpp Pixel Rate	Not Avail	Not Avail	100 Mps	200 Mps	133 Mps	120 Mps	100 Mps
Antialiasing	No[4]	No	Yes	Yes	Yes	Yes	No
Trilinear Texture Filtering	Yes 22 Mps	No	Yes 100 Mps	Yes 200 Mps	Yes 133 Mps	Yes 100 Mps	Yes 100 Mps
Anisotropic Texture Filtering	No	No	Yes	Yes	Yes	Yes[2]	Yes
Scene update at the monitor refresh rate	No	No	Yes	Yes	Yes	No	Yes

CONTENT GENERATION

Chicken and Egg problem

The Talisman program will deliver a richer multimedia experience to the user. To accomplish this goal Talisman provides new capabilities in the graphics subsystem that the content author then can use to make their applications more compelling. This creates the age old problem of getting both the new capabilities into the system and the applications that take advantage of these capabilities as soon as possible. While Intel can afford to put new features into their processor chips well ahead of when software takes advantage of these features; (although even this makes them furious), the highly competitive graphics chip market largely cannot afford to put hardware into their chips that will not be used during a single selling season. These companies strive to minimize their die size (hence product cost) to gain a small cost advantage over competing parts and are reluctant to add product cost for features that may not be supported by several significant applications. Similarly with the shelf life of most games less than 2 months game publishers only want to support features that are widespread during that particular 2-

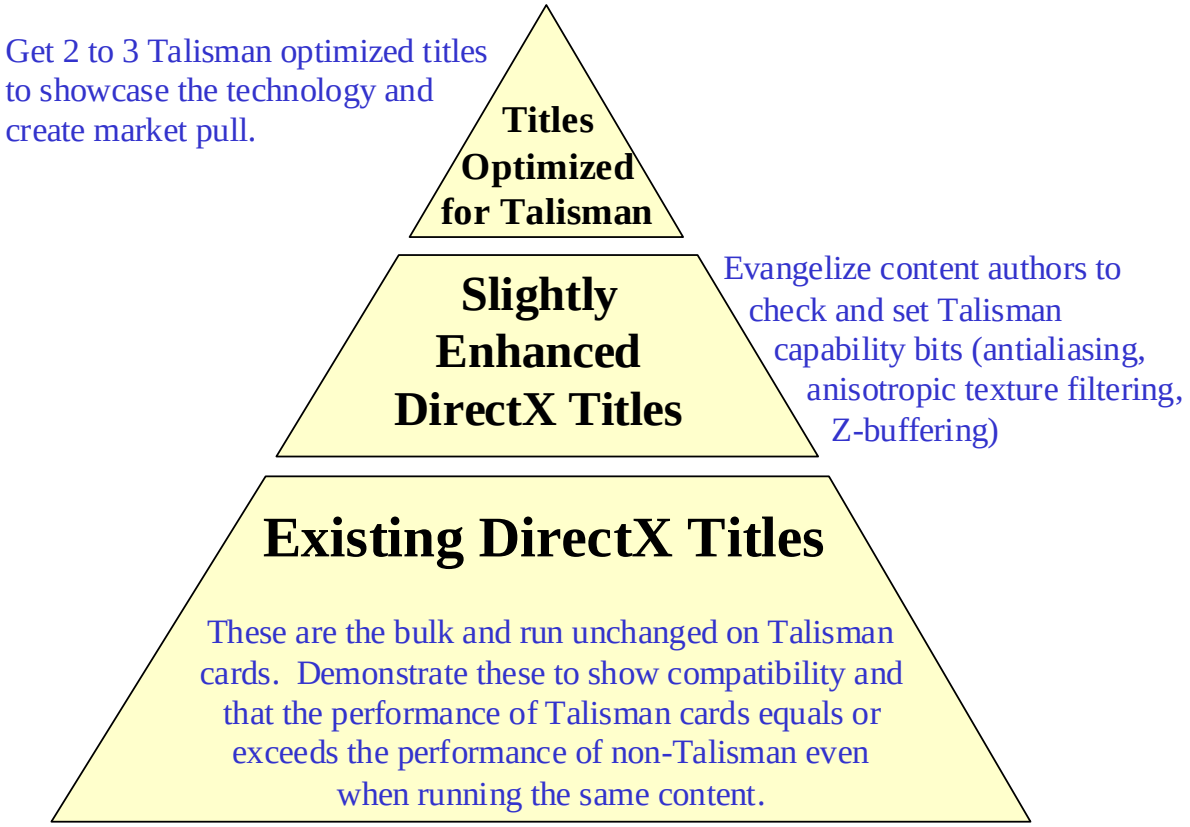
[4] Not with texturing.

month window. Getting both Talisman enabled chips and the content to make use of these chips remains one of the Talisman team's most difficult challenges. So far the team has been very successful in getting hardware companies to start to adopt Talisman features. ***Our content effort has, so far, been an evangelism effort. This, for the reasons just stated, is a tough sell. We need a funded content program to get compelling content in-place for demonstration at WinHEC '98.***

Action Plan

Titles can be grouped into three categories as shown in the diagram below. The bulk of the titles will be DirectX titles with no Talisman optimizations. These will run unchanged on Talisman enabled cards at about the same performance and visual quality as similar non-Talisman enabled cards. We are currently evangelizing the content authors of these titles to do the *"one day"* type of Talisman optimization; check for Talisman capability bits and turn these on when the game starts. These Talisman features include antialiasing and anisotropic texture filtering. Games that can use a hardware Z-buffer can also take advantage of the Talisman Z-buffer without the usual loss of memory (the Talisman Z-buffer is held on-chip and doesn't consume graphics memory). This evangelism effort is a low cost effort with, hopefully, a fair number of DirectX titles moving from the bottom category to the middle category.

A concerted effort is required to generate two to three new compelling showcase titles represented by the small



top pyramid above. These titles showcase what Talisman is all about and create pull for Talisman cards from favorable product reviews and user experiences. Titles in this category need to have full color, high-resolution artwork shot, higher polygon characters, and take advantage of Talisman's ability to render to a DirectDraw surface and reuse objects that either haven't changed, or have changed by only a small amount. Getting these titles created is expected to cost about $ 4M. We expect these titles to be marketed by MS EBU and also be shipped with the Talisman enabled board and bundled system configurations.

A concerted effort should also be made to create showcase titles out of current Microsoft hit titles like Flight Simulator, 3D Baseball and Monster Truck Madness.

SDK

Our software development kit for Talisman is designed to quickly enable a content author to understand and learn how to utilize the benefits of the Talisman extensions. The SDK contains the items listed below.

API Emulation and Reference Renderer

All Talisman DirectX API extensions are emulated in software to facilitate development of Talisman applications. Coupled with this emulation is a high-quality software renderer that supports antialiasing, anisotropic texture filtering, multi-pass rendering, and alpha compositing of affine warped surfaces. This renderer runs at one to three seconds per frame, depending on scene complexity, and hence can demonstrate visual quality and programming model but not performance. Even so, we expect this tool to allow content authors to get started generating content that takes advantage of Talisman technology before hardware becomes available.

Helper Libraries

The following libraries are provided to assist in the rapid development of Talisman-optimized content:

- Affine Library: Performs many of the common sprite reuse functions, such as selecting the initial affine transform for rendering, calculating the best-fit affine transformation for sprite reuse, determining the error associated with reuse, and various utility functions for managing affine matrices.
- Regulation Library: Determines which sprites can be reused and which need to be rerendered based on error tolerances and available system resources.
- Layering Library: Calculates the correct front-to-back layering of surfaces based on geometric bounding information.

Example programs

Example DirectX programs are provided showing how to enable each feature. These include:

- Quality features, such as antialiasing and anisotropic texture filtering
- Object-based rendering, including scene partitioning, rendering to separate surfaces, reusing surfaces through affine warps, alpha compositing, and regulation
- Multi-pass operations (shadows, reflections)
- Motion blur, depth of field, lens flare, and range-based fog

Demonstration program

The Talisman team is putting together a demonstration program that will show all of the Talisman features being used in an interactive setting. This will be used to display Talisman's capabilities at trade shows, conventions and company briefings. The application content will include: 1) 3D geometry that represents a complex visual environment including terrain, structures, animated characters, and effects; 2) High-quality textures; 3) audio elements including synthesized, spatialized sound effects; and 4) video clips to be displayed in the 3D environment. Individual quality features and image reuse parameters can be selectively controlled to demonstrate their impact and performance advantages.

In addition to highlighting the power of the Talisman architecture, this demo has secondary goals of stress testing the hardware, demonstrating how to write scalable content, uncovering authoring tool requirements, validating the DirectX API extensions, and providing richer sample code to ISVs.

Documentation

The Talisman Programmer's Reference provides a description of the Talisman programming model, including partitioning a scene into separable image layers, rendering those layers using Direct3D, and warping and alpha compositing the layers using DirectDraw. It also contains a description of the helper libraries that ship with the SDK.

In addition to our internal documentation, Nigel Thompson is writing a book, "Inside Talisman" that will be published by Microsoft Press early next year. Nigel recently completed another MS Press book titled "3D Graphics Programming for Windows 95."

Tools

These include a Photoshop plug-in to allow images to be edited and saved in TREC compressed form. Future SDKs will include plug-ins for Softimage 3D, Alias Wavefront, and 3D Studio Max.

Jon Peddie Report

Jon Peddie Associates are an industry analyst organization focusing on multimedia.

Specific items of note in the May 2nd report are:

- Talisman-like technology coming from Oak (pages 551-552)
- Cirrus Logic financial results and restructuring (pages 565-567)
- Talisman and accelerating 3D graphics applications (pages 584-593)

The May 2nd report follows in its entirety.

InQuest Report on Talisman

Bert McComas is president of InQuest; a market research organization located in Gilbert Arizona. Bert recently finished and is distributing the attached report containing:

- A description of the Talisman technology
- The programming changes needed to take full advantage of Talisman hardware
- Quality and performance metrics and impact
- Strategies for the content developer
- Strategies for the hardware developer

His report follows in its entirety.