OUR GENEROUS SPONSORS
Thanks to the companies and individuals who made this possible

Jesse Barker  
Lingjun Chen  
English Tiddlywinks Association  
Glenn Fredericks  
Masahito Hirose  
John Kessenrich  
Barthold Lichtenbelt  
Linderdaum  
Tomohiro Matsumoto  
Lawrence McDonough  
Tom Olson  
Kentarō Oku  
“kioku/System K”  
Arzhange Safdarzadeh  
Vicki & Dave Shriener  
Kentarō Suzuki  
Dominic Agoro-Ombaka
# Table of Contents

3 **Table of Contents**

4 **Introduction**

5 **OpenGL Stories**
   - John Carmack
   - Mark Kilgard
   - Mike Bailey
   - Rob Barris
   - Linjun Chen
   - Patrick Cozzi, Christophe Riccio
   - Andrew Garrard
   - Cass Everitt
   - Kevin Goldsmith
   - Masahito Hirose
   - Jonathan Hirshon
   - Eri Hoshino
   - Brent Insko
   - Frank Kane
   - Sergey Kosarevsky
   - Jon Leech
   - Barthold Lichtenbelt
   - Larry McDonough
   - Jon Peddie
   - Ed Plowman
   - Randi Rost
   - SPEC Graphics Performance Group
   - Dave Shreiner
   - Kentaro Suzuki
   - Neil Trevett
   - Shawn Underwood
   - Mitch Williams
   - Tony King-Smith

15 **Photos**
Join us for the festivities as we pay homage to a real game changer - OpenGL! It was in 1992 that the first OpenGL specification was released by Kurt Akeley and Mark Segal. It changed the world.

Thanks to all of you for your dedication to computing development standards and kudos for the passion it takes to succeed.

We are honored to welcome Kurt Akeley as a special guest and to make a few opening remarks.

Kurt has been there since the beginning. Over the years, he has written and collaborated on numerous articles, papers and books. Most notable on this 20th anniversary of OpenGL is the 20th anniversary of the publication of The OpenGL Graphics System: A Specification, Silicon Graphics, Inc., 1992 co-authored with Mark Segal.

He is the recipient of many outstanding contribution awards and is a Fellow of the Association for Computing Machinery. He is also a member of the National Academy of Engineering.

OpenGL standardized access to hardware, pushed the development responsibility of hardware interface programs to hardware manufacturers, and delegated windowing functions to the underlying operating system.

With so many different kinds of graphics hardware, getting them all to speak the same language in this way had a remarkable impact by giving software developers a higher level platform for 3D-software development.

Today, we applaud the founders of OpenGL whose vision and dedication led to the first API standard. Some of them are with us today to join in the celebration and to take us through a retrospective look at the first OpenGL to today's multifunctional versions.

It has been 20 years of growing and learning and adapting to speed of light technology development. It’s a cause worthy of celebration and fun.

Thank you for joining us this evening as we honor OpenGL and its dedicated community!
OpenGL Stories
We asked people what OpenGL means to them

John Carmack
INDUSTRY LEGEND

Way back around 1996, the imminent arrival of 3D acceleration to the consumer space drove me to make a very critical decision. Our existing NEXT-STEP development environment wasn’t going to be getting first class 3D support anytime soon, so I started evaluating all the other platforms available to us. Having previously worked with several rather unsatisfactory proprietary graphics APIs, it was startling how easy it was for me to get things working under OpenGL, and then to make it work on platforms as different as IRIX and WindowsNT.

After writing our next generation game editor (some small bits of which still linger in modern game studios today) with OpenGL, the obvious thing to do was to make a version of Quake that used OpenGL. It didn’t matter that there wasn’t a system under $5,000 at that moment that would be able to properly run it; things were moving rapidly on the hardware front, and it seemed like a good idea to get out in front of it. So glQuake ran on our Intergraph workstations, and also made its way to high end Infinite Reality Systems.

What turned out to be a little surprising to many was how well OpenGL turned out to be suited for the emerging crop of consumer level cards. The rise of the GPU was a technological inevitability, but I like to think that OpenGL and glQuake played a part in making it happen sooner and better.

Mark Kilgard
NVIDIA

My involvement in OpenGL started in 1991 while working at Compaq. My boss discussed with me traveling to California to represent Compaq on a new initiative to define a 3D graphics standard. I went, met Kurt Akeley, Mark Segal, and Phil Karlton. It was one of the first meeting of the OpenGL Architectural Review board. I’ve been involved in OpenGL ever sense. Compaq abandoned its plans for a MIPS-based PC and announced a restructuring of the company. Working for Silicon Graphics looked so much better than Compaq’s restructuring. I took a job at Silicon Graphics working on the X server. I didn’t actually work on OpenGL directly then; I helped with its window system integration. I started writing GLUT and began writing a book on OpenGL. It wasn’t until I switched jobs to work at NVIDIA that I started working on OpenGL itself. I implemented the OpenGL support for NVIDIA’s original GeForce 256. Many GPU generations since, I’m still committed and enthused to OpenGL’s vision of a ubiquitous, powerful, and portable interface for hardware-accelerating graphics. It’s been an amazing 20+ years. Kurt Akeley, Mark Segal, Phil Karlton (R.I.P.), and all the early OpenGL pioneers can look back now with wonder at what they’ve accomplished. I’ve been glad to be a part of it!
Mike Bailey
INDIVIDUAL,
OREGON STATE UNIVERSITY

OpenGL has spent 20 years influencing my life in two ways: my research software development, and in my educational activities.

In research, OpenGL has tracked hardware availability. It has given us all access to the latest speed and feature improvements. Where hardware is able to go, we can rely on OpenGL to bring us along too. And, the fact that it is so cross-platform is a huge plus. Researchers tend to want to bounce around between Windows, Linux, and Macintosh platforms, and now mobile devices as well. No matter where we bounce to, we know that our OpenGL programs will port there. OpenGL’s influence in the education arena has been remarkable. OpenGL is easy to teach (from my point of view) and easy to learn (from the students’ points of views). I was always impressed because students could come into a graphics class with no prior knowledge of 3D programming, and by the end of the first week be smugly showing off their 3D programs to their friends. That kind of quick learning curve is quite motivating, and has brought thousands of new practitioners enthusiastically into the 3D graphics age.

OpenGL, you have had an influence on my life, and many others. Keep up the good work, and we will be back to celebrate on your 30th birthday!!

Rob Barris
NVIDIA

From 2001 through 2011 I worked on the Mac OS X OpenGL graphic engines in games like Warcraft III, World of Warcraft, Portal, Team Fortress 2, Half-Life 2, and Left 4 Dead. Starting in 2011 I joined NVIDIA and currently manage the graphics performance team for Tegra mobile processors.

Linjun Chen
QUALCOMM

Many years ago, I made unsuccessful programming OpenGL on SGI indigo2, and eventually switched to GL. I was hired by Trident Microsystems Inc. in 2001 to develop an OpenGL driver on Windows for a chip called XP4; it is probably the first chip I programmed OpenGL successfully. Happy Anniversary to OpenGL!

Patrick Cozzi,
Christophe Riccio
OPENGL INSIGHTS

OpenGL is more than an API for hardware-accelerated graphics. It is a passionate developer community that brings people together. Patrick Cozzi and Christophe Riccio live in different counties, but work closely together because of OpenGL.

Patrick first started using OpenGL at Analytical Graphics Inc. (AGI) for virtual globe rendering in 2004, started teaching OpenGL at the University of Pennsylvania in 2011, and most recently is using WebGL to develop Cesium, an open-source 3D globe and 2D map engine. Christophe has been an active OpenGL community contributor for over ten years, including developing GLM and the OpenGL Samples Pack. He currently works at AMD doing some OpenGL gardening and is an OpenGL ARB member.

Patrick and Christophe met first on OpenGL.org, and then at SIGGRAPH, and eventually decided to create OpenGL Insights, a contributor-based book that shares the experiences of the OpenGL, OpenGL ES, and WebGL communities. They served as editors, and worked with 52 authors and 17 technical reviewers to make the vision a reality. They are looking forward to a long rest after SIGGRAPH.
Andrew Garrard
THE ENGLISH TIDDLYWINKS ASSOC.

I’m a slight embarrassment to Khronos: I’ve been involved in low-level rendering since before OpenGL existed (my first exposure was an O2), and I’ve been supportively watching the OpenGL spec evolve over that time. However, I’ve been bouncing between a ray tracing hardware company (Advanced Rendering Technology), Direct3D driver writing and embedded 2D combined with GPGPU (3Dlabs/ZiiLABS) while rarely actually coding for a high-level API. I’ve always known how to render a triangle, but was hardly ever actually in a position to use OpenGL to do it. Only in the last few months have I been at Samsung, able to be involved in the Khronos Vision project and using OpenGL | ES and OpenCL. I’m looking forward to finding the free time to produce some free games, especially for the English Tiddlywinks Association (http://www.etwa.org/), and I’ll probably be using GLES to do so - unless I end up ray tracing everything!

Cass Everitt
INDIVIDUAL DONOR

I’ve been using OpenGL since 1993, when I was introduced to it while I was an undergrad at Mississippi State University. I spent a few years working in scientific visualization back when nobody could afford a personal 3D graphics accelerator, but once I was able to buy an NVIDIA Riva TNT, I did my master’s thesis work on it. My thesis work was on bump mapping without DOT3 hardware support. This work was doomed to have a short lifespan as DOT3 capable hardware like the original NVIDIA GeForce flooded the market, but when I posted it on OpenGL.org, it caught the attention of Mark Kilgard. That led to a job at NVIDIA, where I work today. In the last 12 years, I’ve worked on the developer technology, architecture, and software teams at NVIDIA. I also worked stints at id-Software and Epic, which was helpful in learning more about the whole game development process. Pretty much as soon as pocket devices were able to run OpenGL, I began focusing on that new and exciting market. As a hobby, I develop a mobile planetarium and satellite tracker app called Space Junk Pro (http://home.xyzw.us/star3map/) which you can get on Android or iOS. My current personal focus in OpenGL is the open source Regal project (http://github.com/p3/regal), which aims provide the full desktop OpenGL API on all major (including mobile) platforms and the foundation for an ecosystem to facilitate more and better development tools. Space Junk Pro is the first shipping app that uses Regal.

Kevin Goldsmith
ADOBE

Having a world-class, cross-platform open API like OpenGL has played a major part in making Adobe’s applications an indispensable part of many Creative Professional’s tools.

Adobe has made extensive use of OpenGL in our products for many years. We use it for 3D rendering, to drive user interfaces, and for image and video processing algorithm acceleration across products such as: After Effects, Premiere Pro, Photoshop, Revel, Photoshop Touch, Photoshop Elements, Premiere Elements and Flash Player.

Happy Birthday OpenGL!
I started using OpenGL when I was a university student. When I started my first career at SGI Japan as a graphics engineer, I had worked on developing scientific and industrial visualization apps and systems on IRIX and Onyx. In a few years, they gradually moved on to Windows and GPUs. And currently, I’m focusing on developing mobile visualization apps for iOS/AndroidOS and Mobile GPUs. For example, I developed the real time viewer app of the 3D model of the human heart with heart beat animation with OpenGL. It’s for the educational exhibition to visitors of the research institute. I also developed the 3D statistics viewer with OpenGL. It visualizes the statistical data which are published on the internet by various organizations, in graph form. Those graphs are overlaid on the 3D earth model with related geographic location information. I targeted Windows PC when I began to develop these, because they have to handle plenty of data. But, they’re able to run on the mobile devices now with a little change to OpenGL ES! Like this, with innovative efforts of graphics industry, my main computing platform has been dramatically changing, still now. But, my main visualization library doesn’t change from OpenGL. ;-)

Congratulation on your 20th anniversary.

Jonathan Hirshon
Horizon PR

Horizon has been an integral part of OpenGL community marketing and promotion since it was still an SGI solution - and we are honored that we can still contribute to OpenGL’s ongoing momentum and success today! As the Khronos Group PR counsel and one of the only public relations firms specializing in 3D graphics and technology, we are confident in our statement that OpenGL will be celebrating many milestone birthdays to come with a new generation of programmers unlocking its full potential.

The OpenGL ARB has accomplished so much in these last 20 years and we have been privileged to have had the opportunity to interact with some of the greatest technical minds our industry has to offer. Thanks for letting us be a small part of history - and long live the TEAPOT!

Eri Hoshino
DMP

As a leading mobile and embedded GPU IP provider DMP’s offering is mainly based on the little sister of OpenGL, the main versions of OpenGL ES. OpenGL itself is widely used in our company for research and especially several algorithm development. Some of our key technologies like MAESTRO have been developed by experiencing OpenGL to find the most effective way to realize shaders in hardware (HW) IP. In 1st stage we develop the application in shader code using OpenGL. In 2nd stage we replace the 1st stage to the hardware emulated shader using OpenGL before we finally bring it to the full HW based solution.

Brent Insko
INTEL

Intel is delighted to wish OpenGL a Happy 20th Birthday! As a founding member of the OpenGL Architecture Review Board, it is rewarding to be a part of the continued success and growth of OpenGL. The original vision was to create a cross-platform, vendor-independent graphics API. Today, it’s exciting to see how that vision has evolved to devices like smartphones and tablets with the birth and rapid development of OpenGL ES across 3D platforms, unimaginable 20 years ago. Intel looks forward to continue driving graphics innovation in OpenGL through collaboration with fellow Khronos members for many years to come.
Frank Kane  
SUNDOG SOFTWARE

Sundog Software has been supporting OpenGL since its creation in 2006, with the launch of the SilverLining Sky, Cloud, and Weather SDK. The idea was to build an easy-to-use SDK that could take any time, location, and weather conditions, and procedurally generate realistic skies with 3D volumetric clouds at fast framerates. It started as a hobby project, and today is a full time endeavor with customers that include NASA, the US Federal Aviation Administration, TV weather broadcasts, commercial games, and most of the major defense contractors. We’ve also worked with Havok, Unity, Gamebryo, Ogre, OpenSceneGraph, and others to provide tight integrations with popular engines. Originally supporting OpenGL 2.0 and fixed-function pipelines, SilverLining has evolved over the years to support core-profile OpenGL 3.2 and above, real-time GPU ray-casting using GLSL fragment shaders, and an OpenGL ES 2.0 version for mobile and tablet devices is currently in development.

Sundog’s newest product is the Triton Ocean SDK, also supporting OpenGL 2.0, 3.2 core, and above. Triton uses OpenCL and other parallel processing technologies to accelerate a Fast-Fourier Transform based water simulation. Like SilverLining, its API is simple - you tell it the wind or wave conditions you wish to simulate along with the depth of the water, provide any reflection or environment maps you have, and Triton renders realistic water with spray, foam, and ship wakes, at hundreds of frames per second. Although Triton was released under a year ago, it’s already in use by the US Navy and many in-development games and simulation projects. The development community surrounding OpenGL has always been core to our business, and we’re proud to be a part of it. More information on Sundog Software’s products and services is at [http://www.sundog-soft.com/](http://www.sundog-soft.com/)

Sergey Kosarevsky  
LINDERDAUM

We are using GL since early 2002 when we have started experimenting with version 1.1 of the API. At first we had written a couple of demos featuring then-popular adaptive terrain rendering algorithms. Later those sources has become the rendering system of our open source Linderdaum Engine ([http://www.linderdaum.com](http://www.linderdaum.com)). By the end of 2008 we have switched to OpenGL 3.x and at the same time we have implemented hardware-accelerated volume rendering techniques on consumer-class GPUs. In the past few years we are keeping up with the latest mobile technologies. The engine has become available on the Android platform with an OpenGL ES 2.0 rendering backend and we are working towards an iOS port. Linderdaum Puzzle ([http://play.google.com/store/apps/details?id=com.linderdaum.engine.puzzL](http://play.google.com/store/apps/details?id=com.linderdaum.engine.puzzL)) is the first shipping game that uses our Linderdaum Engine.

Jon Leech  
ARB SECRETARY & FORMERLY SGI

I was part of the team at UNC-Chapel Hill developing Pixel-Planes 5, an extremely high performance graphics system whose API was a variant of PHIGS. When our successor machine, Pixel-Flow, was being designed, I initially encouraged use of IrisGL, but shortly into the process, the earliest versions of OpenGL 1.0 became available to universities and we shifted to that. My first exposure to the OpenGL ARB was as a guest at a meeting at DEC in 1994 where I described the programmable shading architecture of Pixel-Flow and the API extensions and shading language we’d developed - a decade before such capabilities were added to OpenGL 2.0.
After joining SGI in 1997 and becoming the ARB Secretary, I learned a lot about the group dynamics of a bunch of strongly opinionated engineers; “Herding cats” sometimes came to mind. At first I thought I was seeing hidden corporate agendas everywhere, but in time, I realized that sometimes the most passionate debates happen between people who share a similar compelling vision - but whose communication styles differ wildly. Before my time with the dedicated ARB members, I had never seen the phenomenon of “violent agreement,” where people are so caught up in their own momentum they don’t realize they are saying the same thing! Despite the occasional healthy fireworks, we’ve not only survived, we’ve prospered over the years. I look forward to sticking around for quite a few more.

Barthold Lichtenbelt
Director of Tegra graphics, NVIDIA
OpenGL ARB Working Group Chair

Got involved in graphics by accident. Hired at HPlabs in Palo Alto to work on 3D image processing. Then transferred to the HP Unix graphics workstation on Fort Collins, CO in 1995. When everyone thought they could sell a graphics card for $10K, without texture mapping functionality. That was another $5K :-)

Joined PC graphics startup Dynamic Pictures. Watched many competitors die and be eaten, including DP. Fun times in the 90s!

Got involved in OGL 1.2 (I believe) - was there from pretty much the beginning of Khronos with Jeff Layne (sp?) trying to set it up. I remember a meeting on the river boat in Sacramento....

Helped move OGL from version 1 to version 2, through a proposal that 3DLabs made for a high level shading language, as the standard interface to programmable units in the gfx hardware. I recall a meeting with NVIDIA (I was at 3Dlabs at the time) and Kurt, who was at NVIDIA at that time, trying to get agreement on the direction to take. NVIDIA was pushing Cg heavily.

I actually worked on the implementation of a DirectX 5 driver for Dynamic Pictures before I started working on OpenGL. Sssssh.....

Took over as chair of the ARB the same day the ARB was subsumed into Khronos. The rest is history.....

Larry McDonough
RESEARCH IN MOTION

I’ve been working with OpenGL since 1993 animating 3D characters in real-time. I got my picture in National Geographic Magazine in October, 1995 showing off the world’s first simultaneous real-time full-body and facial motion capture animation system (called VActor by SimGraphics). This was demo’d in SGI’s booth at Siggraph for a couple of years in a row around that time. Characters were rendered in real-time using OpenGL on Onyx Infinite Reality computers and used in live audience demonstrations - such as in tradeshows, Las Vegas hotels, corporate events, and television.

I joined Silicon Graphics in 1995 and worked with Kurt Akeley directly trying to build a CAD industry consortium centered around OpenGL Optimizer (called the LMVAC for Large Model Visualization API Consortium).
As you take in the wonders of the 39th SIGGRAPH, you see the huge impact OpenGL has had on our industry: *any* visual computing application can exploit high-quality, high-performance OpenGL capabilities. These capabilities enable developers globally and in diverse markets such as broadcasting, CAD/CAM/CAE, entertainment, medical imaging, and virtual reality to produce and display incredibly compelling 2D and 3D graphics. I feel that nothing has been more important, longer lasting, or as broadly significant to the industry than OpenGL - and here why:

‘For the people, by the people’
The OpenGL ARB is independent. Nobody tells them what to do. They listen to what the industry needs and chart their own course. As a result, OpenGL has industry-wide support and has the divine role of being the only truly open, vendor-neutral, multiplatform graphics standard.

‘The strongest Survive’
OpenGL allows new hardware innovations to be accessible through the API via the OpenGL extension mechanism. In this way, innovations appear in the API in a timely fashion, letting application developers and hardware vendors incorporate new features into their normal product release cycles. Extensions are continuously being developed and as they become widely accepted, they are considered for inclusion into the core OpenGL standard. This open process allows OpenGL to evolve in a controlled yet innovative manner that responds to community demands.

In the beginning
Computer graphics was a dog eat dog world, every one (company) for themselves and death to anyone who got in the way.

Chaos, conflict, massive loss of precious engineering hours were the norm as programmers and computer companies fought to find and keep customers.

Here I am to save the day!
And then OpenGL was given to the industry – free, open, friendly, supported, community driven, and life was good.

Peace in the valley - at last
And now all the tools are available, organized, easy to understand and use, and productivity has soared.

OpenGL gives the programmers the tools they need to expose the power of the hardware and take us to levels of creativity, capacity, and consciousness.

OpenGL – where would we be without it?
Ed Plowman
ARM

A long time ago in a North London Business Park, not so far away...“API WARS”!

In the mid 90’s I started my professional graphics career and was plunged directly into the bedlam that was the desktop 3D gold rush. I looked on as my colleagues battled with proprietary API’s, seemingly hundreds of them! The pain and anguish on their faces stayed with me forever... Eventually one by one the vendors fell victim to their own folly and survival depended on an alliance and OpenGL is born. Over the years some have sought to test this alliance raising temperatures, but it has stood the test of time.

When a new war looked to be looming in mobile I recalled the anguished faces of my former comrades and vowed “never again”. When it came time to choose the right path, the directional infinite light source fell squarely on OpenGL. I am hugely thankful to those that created and contributed to OpenGL over the years and thankful for the opportunity to add my own small contribution.

Long may the story continue...

Randi Rost
INTEL CORPORATION

In the late 80’s and early 90’s, I worked at DEC and was responsible for defining the 3D graphics software strategy for DEC’s MIPS-based workstations. I was only a few years out of school, but I led the effort to define a 3D extension to the X Window System called PEX. Over the course of a few years, we started to realize the pros and cons of doing 3D graphics across a network connection that supported the PHIGS and PHIGS+ standards. To allow for direct hardware access and higher performance for immediate mode graphics, we started an effort to define an API more like IrisGL. This new API, designed by Allen Akin, was called “G.” I became the manager of the group tasked with implementing G on DEC’s graphics hardware.

Shortly after we started implementing G, SGI approached us about joining their effort to create an API derived from IrisGL that they hoped would become an industry standard. Their vision was highly aligned with ours. We believed that it would be in the best interests of software developers to have a single standard API rather than coming to market with our IrisGL competitor. We agreed to participate in the effort and we were allowed to have a seat on the Architecture Review Board that SGI had defined. One of the first big management challenges I faced in my career was redirecting my implementation team from G (in which they now strongly believed) to OpenGL. DEC succeeded in becoming one of the first to ship an OpenGL implementation (if I recall correctly, technically, we were the first to ship). Attending the ARB meetings over the first few years and building relationships with some of the industry’s best and brightest people was a lot of fun and a great experience for me. Almost a decade later, I worked at 3Dlabs and we were developing a highly programmable graphics chip called P10. Dave Baldwin, our chief hardware architect, and Neil Trevett, our VP of marketing, believed that we should try and bring programmability to OpenGL, so we worked internally to prepare an initial draft of a shading language to share with the OpenGL ARB.

On Sept. 10, 2001, I traveled to the Bay Area for the OpenGL ARB meeting. The next morning, I met Dale Kirkland for breakfast prior to the ARB meeting to go over last-minute strategy. Instead of talking strategy, we watched the horror of 9/11 play out…it was 7 am PDT, and the World Trade Center was on fire. We watched in disbelief as the first of the towers collapsed.

The ARB meeting that day was at Apple in Cupertino, and I did make the presentation to add a shading language to OpenGL. The mood, as you might expect, was quite somber, and during breaks and lunch we would bring up a web browser and project it on the front screen to see the lat-
est news about the terrorist attacks. By the end of the meeting that day, we received approval to proceed cautiously with the shading language for OpenGL. I was caught in the air traffic shut-down and was stranded in California through the end of that week. So yes, I remember exactly where I was on 9/11…at an OpenGL ARB meeting.

Working with the team that produced the first implementation of the OpenGL Shading Language was one of the highlights of my career. As we implemented more and more of the compiler, we were the first to develop and demonstrate shaders that implemented bump mapping, analytic antialiasing, fractals, procedural textures, and lots more. It was a blast to be part of proving that a shading language for OpenGL was both possible and necessary.

It’s certainly hard to believe that in the 20 years since OpenGL was released, I have worked at seven different companies…and four of those companies no longer exist! What a ride it’s been with OpenGL!

Randi Rost, Intel Corporation
OpenGL ARB representative for DEC, Kubota Pacific Computer (later Kubota Graphics), Hewlett-Packard, Dynamic Pictures, 3Dlabs, Creative Labs

Dave Shreiner
ARM

I started working with OpenGL in early 1992, when I wrote SGI’s introductory OpenGL programming course. During my career, I spent a lot of time with OpenGL developers who were doing amazing things: NASA and the Space Shuttle program making space travel safer; Boeing prototyping their 777; numerous visual simulation companies preparing pilots for all sorts of crafts, to name a few. Now, twenty-one OpenGL-related SIGGRAPH courses, seven editions of three different books, and countless great interactions with developers later, it’s a great pleasure to wish OpenGL a happy 20th birthday.

SPEC Graphics Performance Group
DONOR

Happy Birthday from your friends in the SPEC Graphics Performance Group (SPECgpc). SPECgpc develops SPECviewperf, the leading performance evaluation software for measuring the 3D graphics performance of systems running under the OpenGL application programming interface. The benchmark’s test files, called viewsets, are developed by tracing graphics content from actual applications. Current SPECgpc members include AMD, Dell, Fujitsu, Hewlett-Packard, Intel, Lenovo, NEC and NVIDIA.

Kentaro Suzuki
INDIVIDUAL DONOR

I’m a student in university and studying about computer graphics, and so need to show various images on PC. There are some methods for showing images, but a multiplatform, widely used, and fast graphics API is only OpenGL. Therefore, I use OpenGL whenever I want to show something 2D or 3D graphics. My recent work using OpenGL is a Live Coding Tool for editing GLSL.
(www.youtube.com/watch?v=eRuTRs3hxfc)

Tadamasa Teranishi
INDIVIDUAL DONOR

I am doing application development using OpenGL, and have worked in driver development for 20 years. Since I’ve been using OpenGL from version 1.0, I’m happy to celebrate the 20th anniversary. By an OpenGL mailing list of Japan, what it argued about every day is remembered longingly. My best wishes for OpenGL’s future growth and prosperity.
Neil Trevett  
NVIDIA

I never wanted to do anything but graphics. Making things come to life on screen has always been magical to me. At 3Dlabs (first called benchMark Technologies and then DuPont Pixel back in the day) we started making bit-slice graphics processors in the UK pretty much at the same time that SGI and Pixar were starting to make hardware in California. 3Dlabs always had insight to GL being the right API; before OpenGL we built IRIS GL accelerators for Sun workstations. When OpenGL was born, we built the GLiNT family of 3D processors for OpenGL on PCs, managing to get letters from SGI (GL), Intel (i) AND Microsoft (Windows NT) trademark folks with just a five letter name, pretty good going! When embedded systems started to demand 3D it seemed natural to create a streamlined version of OpenGL. It was too soon for the ARB to develop “Embedded GL” but the ARB community has always encouraged innovation, and so while at 3Dlabs, Randi Rost and I took the proposal for a streamlined GL to the fledgling “Khronos” – where the mobile industry then converged to create OpenGL ES. It’s great to see that OpenGL and OpenGL ES are now the most widely deployed 3D APIs, and enabling a new generation of platforms to take advantage of that ubiquity with WebGL. Every company works to keep the delicate commercial balance between supporting open standards and developing proprietary technologies, but I believe OpenGL has been instrumental in making 3D the heart and soul of billions of devices – and has helped make the world a better place along the way.

Shawn Underwood  
INTEL, FORMERLY SGI

As a product marketing guy I had been aware of OpenGL from roughly its beginnings. But I never fully comprehended it until I went to my first OpenGL BOF at SIGGRAPH having recently joined Silicon Graphics. There were a couple of hundred people packed into a room (most BOFs were a couple of dozen); most of them crazily passionate about how the graphics sausage was being made. That stuck with me, and when we were approached several years later about creating a mobile version of OpenGL, I realized how powerful that community could potentially be; be to drive graphics capability into virtually all consumer electronics. When the ARB later needed a new home, it was a natural move to have it all managed under Khronos Group where that passion for graphics from those dedicated people could flourish.

Mitch Williams  
3D Graphics Dev. & Marketing Manager - 3D-Online

We landed a new client at 3D-Online that wanted to place their mechanical engineering drawings up on the web. Our existing solution could not render the large scale designs so we swapped out our renderer with OpenGL.

The development took only a week. Not only was the client thrilled with the performance, but this opened up several new markets for us.

Tony King-Smith  
VP MARKETING, IMAGINATION

Imagination is a Promoter in Khronos and has a strong belief in open-standards. We have a long history with OpenGL and, having been in the industry since the days when all APIs were proprietary, we are big fans of OpenGL and the transformations it has encouraged in the market. Most recently OpenGL has been a vital contributor to our work in mobile graphics, forming the basis of our PC emulation of OpenGL ES, which is widely used by mobile graphics application developers.
A FEW PHOTOS

Some great memories from over the years

CLOCKWISE FROM TOP:
OpenGL ten years ago
Kurt Akeley’s SGI ID card
A big ball of OpenGL Goo™
Jeremy Sandmel, Rob Mace, Michael Gold (L to R)
The OpenGL Lotus Esprit
OpenGL’s fate sometimes rested in these hands!
HAPPY 20TH BIRTHDAY

www.opengl.org

Barthold Lichtenbelt, OpenGL Work Group Chair, invites you to join us for the exciting journey OpenGL will take over the next 20 years…