
Teach Yourself Demoscene in 14 Days

A guidebook for newcomers to the demoscene, spawned from a certain pouet.net forum thread.

Also available as a seminar video on youtube.

Day 1 - Demoscene?

The demoscene is an underground computer art culture. The term demoscene comes from the word demo, short for demonstration. In the context of the demoscene the word demo means a realtime audiovisual application which is demonstrating the capabilities of the machine it runs on.

Demosceners (“sceners”) are what we call the folks with too much free time that abuse their computer skills to create releases under the demoscene.

Demosceners often use nicknames (“nicks” or “handles”) to identify themselves. They also tend to hang out in so-called demogroups. Some demosceners are active members of multiple demogroups, with or without using the same nickname.

Let’s get one thing clear: the demoscene has no commercial purpose. The only thing you’ll get out of the demoscene, and this only comes after investing a significant amount of your free time into it, is a few useful softskills and a large community of computer nerd friends.

Demoscene releases are meant to show the limits of the machines, the technical skills and artistic sensibility of the makers. There are no rules to what kind of release you can make on the demoscene. Some demos are made as technical benchmarks, others as conceptual art, most are done just for fun. It is entirely up to you to explore what you like doing and share it with other demosceners.

Demoscene releases can be divided into certain categories: * Track, an audio piece, can be in an executable format, in a tracker module format or in a pre-rendered wav/mp3 format * Graphics entry, drawn or rendered images with fixed resolutions and/or a restricted color palette * Demo, an audiovisual real-time executable demonstration for a certain platform * Intro, typically a demo with file size limitation all packed into a single executable file that includes all the assets (popular size formats are 256bytes, 512bytes, 1kb, 4kb, 8kb, 64kb) * Animation, rendered graphics videos * Demopack, a collection of demos in a single disk * Musicdisk, a collection of demoscene tracks with an executable player interface * Diskmag, a collection of texts about the demoscene with an executable graphics interface * Wild entry, everything else (including live performances, videos of demos on uncommon platforms, videos about demomaking, etc)

Releases typically occur at demoparties, gathering events for demosceners.

Demoparties come in many flavors but they are usually 2 to 3 day events focused on bringing together demosceners and challenge them to participate in a series of competitions (also called compos) on

different categories. The entries are shown on the big screen, people vote for what they like and the winners take home a symbolic prize.

Given all this knowledge, your mission, should you choose to accept it, is to, in 14 days, make your first demoscene entry, meet some fellow demosceners, participate in a compo at a demoparty and eventually “make a megademo-shock to Japanese brain” as some sceners say.

On this first day of the rest of your life you should probably still google a bit more about the demoscene and watch some demos! If you have trouble running most demos on your platform of choice, Annikras spent a lot of time capturing demos into high quality videos and uploading them to his youtube channel, which might be useful, although demos are always best experienced running live on their target platform.

There is this great documentary about the demoscene called Moleman 2 and if you’re a scholar you can find a list of scene related scientific writings at Demoscene Research site.

Signing Up and Signing In

Other than your nickname, you don’t have to give anyone any information to join the scene or release a production. However, sometimes it’s nice to be able to talk to folks who have been doing this for a while :) . The easiest way is to register for an account (a “SceneID”) at scene.org. That will give you access to a number of scene websites, including:

- Pouët.net: The largest site for scene-wide discussion of demos and other topics. The “BBS” is the discussion forum. A good idea is to lurk for a while and learn the etiquette before you post!
- Demozoo: An archive of information about scene demos, graphics, and music.
- Wanted!: A site for asking for people to work with you on your releases — and for offering to work with others on theirs!

These sites have a big “SceneID” button — click that to login. It usually looks like:



Bear in mind that not all demoscene content is necessarily safe for work or suitable for kids. That said, there is a tremendous amount of information in these sites, and a lot of very experienced people hang out in the forums.

Day 2 - Find Your Vocation

On the second day of the rest of your life you should figure out what you want to do on the demoscene.

A demo is usually a group effort. Sceners are typically divided into 3 roles:

1. Graphician - responsible for the artwork, 3d modeling and animations
2. Musician - composer and producer of the audio soundtrack
3. Coder - programmer of “effects” and tools that will enable you to get your demo done before the deadline

You can always just do everything yourself, but it obviously requires a high time investment to be proficient in each of the areas. So it's common practice that sceners specialize in one of the roles while “outsourcing” the others to their group mates.

So, tell me, what do you want to do when you grow up? Draw imaginary worlds? Annoy your dog with high pitch sounds? Spend your whole night debugging spaghetti code? Now is the right time to think about it. It's never a final binding decision. You can always go back and learn something else, but if you want to finish a demo before the deadline it really helps to know what you will be focusing on first.

There is an additional role in the demoscene, which is the role of the organizer, often not directly involved in the practical production of demos, people in these roles are often critical to a successful sprint and keeping the spirit alive within the group. Their tasks can range from finding complementary group members for the project, managing moods, reminding of deadlines, organizing events, promoting releases, etc.

Remember that a demoscener can easily wear multiple hats of these different roles.

Don't be scared to try out different roles, even if you don't stick with it you will undoubtedly learn invaluable information on how it works. This knowledge will help you in the future.

This does not imply it will be easy to master any of these roles. All of them are life-long commitments that take years of practice and exploration to achieve something you could consider great, but we all have to start somewhere, so let's get started!

By the end of day 2 you should have an idea of what you want to more seriously explore. If you have doubts what each role actually entails, i strongly recommend you to google for some beginners tutorials on the different areas.

Day 3 - Basics of Graphics

The role of doing graphics for a demo is typically supported by a so called graphician. To be a successful graphician in the demoscene you should hopefully enjoy drawing and being randomly creative. There are many tools you can use to draw graphics on the computer.

Pixel art graphics typically focus on a limited palette of colors and image size. The restrictions come from the oldschool machines that had such limitations on hardware. Small images are typically called sprites. 2D animations are created by cycling between the different frames of a sprite animation. This is the basis of 2d animation. You can google for pixel art graphics tools and find a whole bunch of modern editors for pixel art. Real sceners will tell you about Graft2 or the more recent Multipaint.

If pixeling makes your stomach all queasy, don't worry, that doesn't mean your career as a graphician will be over before it even started. You can still take the role of 3D modeler (using 3DS Max, Maya, Lightwave, ZBrush, Blender, etc) and 3D animator.

The graphician is often also responsible for the overall design of the demo. Including concept, scene composition, transitions, etc.

Most graphicians are proficient with Adobe Photoshop and Illustrator, or their (possibly free) alternatives.

By the end of day 3 you should know what tools to use if you want to do graphics for a demo or for a graphics compo entry. So just try a few of them out, check some tutorials, learn the application shortcuts and try to master the most interesting tool you could find.

Day 4 - Basics of Audio

There are many tools to produce audio freely available for download. Ranging from sample editors to complete DAW (Digital Audio Workstation) solutions.

Modern PC demos typically play a pre-recorded final mix of the audio (an mp3 of the track), but the roots of the demoscene involve sample generation and the so called tracker formats. Most demos for oldschool platforms still feature live mixing playback of a tracked track.

The tracker format can be divided into 3 main concepts: samples & instruments, patterns and sequencing order. You can typically load or configure the parameters of samples that can be played. Certain trackers allow you to configure samples into instruments by defining behaviors on how the sample will be played when triggered under different pitches. The samples or instruments are then sequenced in patterns that will be played back under the stipulated tempo. And you can set the order of how the patterns will be played under the song sequence. These simple rules are the basis for most trackers that exist out there, each of them with their unique, quirky and often highly counter-intuitive menus that you'll grow to love.

Certain platforms have sample and instrument restrictions, such as only playing sinus, saw or square wave forms. Or only allowing certain types of filters. The limitations of each soundchip platform gives it a signature sound, the exploration of these signature 8bit and 16 bit sounds is what popularized the chiptune music genre.

To create sound for limited size intros demomakers have developed their own synthesizers, forcing the musicians to generate instruments procedurally. Storing the required function steps to recreate the instruments procedurally takes much less space than storing the entire sample. Popular demoscene production synths include 4klang and clinkster.

Of course if you are doing sound for demos instead of intros you are not limited by size, so you can do your soundtrack in whatever application you prefer: Fruity Loops Studio, Ableton Live, Cubase, Reason, Logic Pro, etc.

By the end of day 4 you should know what tools to use if you want to do sound for a demo or a music compo entry. Try a few of them out, check some tutorials, learn the shortcuts and master the tool you find most interesting.

Day 5 - Basics of Code

Learning how to program is relatively easy, there are tons of books and free courses on the subject. Khan Academy or Codecademy are as good places as any to learn how to program in general. You'll need to learn about variables, functions, classes, data structures, loops and algorithms.

There are many programming languages, platforms and styles of programming with their respective zealots preaching "the way". Some languages favor certain styles over others but overall you should focus on the language you're most interested in learning or most comfortable with.

To avoid the initial frustration of not being able to get something showing on screen while you build your rendering engine you should probably start learning graphics programming using environments such as Processing, Unity3D or cables.gl, these environments are great ways to get something going fast and you can always move away from them and do your own engine later on when you're more experienced. Some folks have prepared a demoscene starter kit.

There are many ways to program a demo. There is no such thing as the single best way. In other words, you should do demos as it feels most natural to you. As long as the entry gets done in time to get shown on the big screen, you're doing it right.

Some programmers get a bit caught up in the right way to code, reading books, following best practices, debating them, learning new trendy methodologies or endlessly refactoring their code and never actually releasing a demo. If that's you I strongly suggest that you take a look at <http://programming-motherfucker.com/> and just release your first demo.

Another point worth addressing is that you don't have to prepare the entire development chaintool / engine / framework yourself to do a demo. Some folks prefer to work on tools / engines / frameworks and then having other demosceners use them to make demos. While others hate developing tools and engines and just enjoy putting a demo together. Of course it's useful to dabble on both sides of the spectrum, but at the end of the day you should focus on what you are most interested or comfortable with.

There are some great online resources to learn programming graphics. If you're interested in learning OpenGL you can check the good old NeHe Tutorials or the more recent Learn OpenGL tutorial website. A more recent bible of modern graphics programming is also the Physically Based Rendering book. Also worth checking is the complete list of Awesome Creative Coding.

Most demo effects in the recent years (2010-2018) are heavily based on pixel and fragment shaders, either coded in HLSL or GLSL. To learn shader coding you have this great online book called The Book of Shaders. There is also a well known online sandbox for open source GLSL experiments known as Shadertoy. If you seen shader live coding during demoscene events it's likely that it was done using Bonzomatic.

If interested in doing tiny bytesize intros (128bytes, 256bytes, 512bytes) you can find some great resources on this subject at sizecoding.org. If you're more interested in 1k / 4k / 8k development, you can find useful resources at [in4k](http://in4k.com).

Try to program a new effect each day and you'll eventually start to get the hang of it and figure out which areas you should read up on to improve your skillset.

By the end of day 5 you should know what it takes to take on the programming role of demomaking.

Day 6 - Tools and Resources

There are many ways to do demos. Some folks enjoy coding everything from scratch, others prefer to use tools to facilitate their work. There are many types of tools, some are commercial (like Maya, 3DS Max, Photoshop), others are open source or freeware. And then we have the so called demotools. Demotools are tools made by demosceners especially to help create better demos. Demotools are especially important if you're working on size limited entries. Some groups develop their internal demotools, others prefer to go open source in order to have more users and raise the bar. The end goal is always to make more good demos.

I would strongly advise you to take a quick look at the different demotools available and see which you can use, this is important to avoid reinventing the wheel and achieve good results before you grow frustrated with the learning curve. You can find a list of the most famous demotools on [pouet](http://pouet.net), they range from texture generators, music trackers, synths or graphics tools or format converters to full featured demomaking engines, executable packers and linkers.

Here is a list put together by cce/peisik of all demotools (with screenshots) put together in one place for easy reference.

The best way to know what tool might interest you is to either check them all out one at a time, or visit a demoparty on location and just randomly ask another fellow scener. Your tools largely depend on your demoscene role of choice and target platform so i hope you have a hint of a plan by now.

As you progress you will notice several limitations and quirks with the tools you use, and will most likely feel the need to either develop your own set of tools or contribute to active existing projects.

Similarly important to knowing the tools of the trade available to you are what resources are out there, including free tracks, samples, sprites, or animated models; usable tools, editors and engines; tutorials; and ways to get motivation or support, including Discord, Reddit or irc channels. It's usually a matter of doing a few google searches or asking fellow sceners about their demomaking habits.

Demosceners typically have very strong and very individual opinions on what is cool and what is lame. The demoscene has no written rules. Different people have different concepts of what "their" demoscene is about. So it's quite easy to get negative commentary. You should realize that an opinion is just an opinion, you don't have to please everyone. The demoscene is a hobby, so you should above all, aim to please yourself first. If you put effort and dedication into your entry I'm sure people will acknowledge your effort regardless of if they liked it or disagree with some of the "shortcuts" you took in their eyes to get the demo done.

My recommendation is to always be honest with your sources, entries typically have a readme.txt file in the package, this is a text document where you can describe your work process, what tools you used and why. If you include proper documented information the people who read it will no longer feel that cheated if your demo used a ripped soundtrack, repurposed some shaders or used a commercial engine. They will evaluate it accordingly.

That being said, the more "original" material and effort you put into the entry the more impact it will have on other demosceners. Generally speaking: reusing with proper credit is generally considered fair game, claiming ownership of someone else's work is not.

By the end of day 6 you should have a good idea of what tools and resources are out there to assist you on your plan for world domination.

Day 7 - Experiment

Whether you're a musician, graphician or coder there is one golden rule in the demoscene: Don't be afraid to experiment. I promise you on first hand experience that as long as you keep experimenting with things you will eventually release something you are truly proud of. So don't be afraid to experiment!

Experiment with tools, experiment with ideas.

Repurpose, remix, discuss things with others.

Don't be afraid to fail, it's normal to fail, that's how you find interesting things, that's how you learn. So go forth and fail, fail hard, fail miserably, fail often, eventually you will succeed!

Don't expect your first track to be great. Don't expect your first pixel graphics to look good. Don't expect your first demo to be any good. The first demos from 90% of sceners are a total load of crap!

You should also be aware that it's quite common to get creative blocks or lose your motivation to do "demoscene things" from time to time. Just remember that demoscene is a hobby, you're free to do whatever you want at whatever pace you please. Some sceners take decades to release something, working on their masterpieces on and off!

At the end of day 7 you should now have a good grasp on the importance of experimenting with things.

Day 8 - Achievable Goals

As previously mentioned there are plenty of ways to make a demo. The only "right way" is your own way, the one that is the most interesting and fun for you.

That being said, an effective rule to ensure a release will be released is to aim for achievable goals.

If you are not reasonable with your goals you'll soon find yourself demotivated and failing to meet the deadline. For most projects it's better to start small and build on top of what you already have working.

If you have no idea where to start here are a few pointers: * Some people try to replicate some effects after seeing something interesting on other demos, movies, adverts or nature. Building up a batch of different effects which can later be adapted to match a track. * Others prefer to nail down a concept first and only then get to work on it's different components. * Some sceners only feel the drive to create something when they are challenged with a very specific issue. * Some coders only do engines, frameworks and tools. Sharing them with other sceners to see what they can do with them. * Some graphicicians fool around with engines and tools until something seems interesting. * Some musicians prepare timelines and storyboards for their sounds.

All these approaches are valid. As mentioned several times already: demoscene is a hobby. So don't treat it as work, do whatever you like whenever you feel like doing it.

Some pro tips to keep in mind:

1. There is a finite number of productive hours per day, there is a finite number of days until the demoparty deadline, aim for achievable goals. If you achieve them fast, build on top of them.

-
2. If you have no ideas, try random things. Eventually you will come up with a concept. Don't let a lack of initial concept paralyse you.
 3. Work on the project at least 5 minutes each day, even if it's just tweaking some colors. This helps to keep the project in the back of your mind and eventually you will find the motivation to spend more than just 5 minutes and do a bigger push.
 4. Communicate your progress with your team mates. They might give you important feedback, and it can also motivate them to contribute.

Some additional protips by Gargaj: Getting stuff done - practical solutions for pulling off large projects

If you keep all these things in mind by the end of day 8 you should be able to lay down a work plan aimed at reaching achievable goals.

Day 9 - Proof of Concept

Now it's time to stop procrastinating and get your hands dirty! It's been scientifically proven that anyone spending over 10,000 hours doing any task will eventually become an expert in it. It's very hard for the human mind *not* to learn from mistakes. So don't be afraid to make them, as previously mentioned: fail often, fail hard, eventually you will succeed.

It's normal to be afraid of tackling a new challenge. Some people simply lack confidence. Others prefer to brainstorm and try to figure out the best way to attack it before they start. There is nothing wrong with taking your time planning a strategy, but it's all quite useless if you don't actually sit down and try to get it done. So now is the time to try it!

If you're feeling clueless, try using the whiteboard or the good old pen and paper, drawing helps the brain puzzle things out. It doesn't need to be anything fancy, just a sketch, notes, a list of ideas, a list of tasks, or a mapping of concepts, even some simple workflow charts can be extremely helpful to get your thoughts in check.

Force yourself to work on the issue atleast 5 to 15 minutes every day. Even if it's just analysing what you have, or doing small changes, open your tool and try something. More often then not you'll end up spending more time working on it then you initially planned. This daily discipline is useful to keep the task fresh in your mind.

If an empty canvas is too daunting, throw something random at it. Either play some random notes or draw some random shapes. Random generators are an easy catalyst for new ideas.

Trying to do a proof of concept for your idea is the best way to get things rolling but don't expect the final results to match your original idea. Don't get disapointed if it's not what you had envisioned. Atleast it's something. Even if you conclude it's just crap, by going through the process of creating it

you learned things which will help your next attempts. You gain invaluable experience points from everything you do.

By the end of day 9 you should be aware of the importance of building a proof of concept of your idea and some insight on how to break out of the demosceners procrastination curse.

Day 10 - Deadlines Are Important

Nothing like a good old deadline to force you to shape your ideas into a finalized entry.

Demoscene has a natural deadline called the next demoparty. Every demoscener has an “oh shit, now i have to make an entry” moment. It usually comes right after buying the flight tickets to a demoparty.

Sure you could just go to the demoparty and socialize. Nothing wrong with that. Except that everyone at the party will most likely ask if you’re working on an entry for the compos, and there is only such a limited amount of plausible excuses you can offer that justify not working on an entry while there is time left.

Even half an hour is plenty of time to do *something* for a compo. It might not be the mona lisa of the demoscene, but it’s an entry. For those who didn’t have time to do anything before the party there are also fastcompos. These competitions are prepared specifically to force you to do something within small time constraints.

If you don’t have any releases for the party, you’ll spend the event watching other people’s entries on the big screen and wondering if you couldn’t have done something that would place higher than that piece of crap you just saw or heard.

So don’t be afraid of the deadline. Deadline is your friend. It forces you to stop the daydreaming bullshit and focus on what’s really important and actually achievable. If the deadline wasn’t there you would most probably never actually deliver anything. “Art is never finished, only abandoned.” Look at the deadline as the point in time where you will be liberated from your current artistic burden. And if you don’t buy into that philosophy you can always release a final version of your entry later on.

By the end of the 10th day you should realize the importance of deadlines, how to not fear them, but respect them for the panic and closure they can bring to the demo making creative process.

Day 11 - Get Help

You can do great demos alone. Lots of demosceners make demos alone. That doesn’t mean you have to.

Most demosceners prefer to work in groups. There are advantages and disadvantages to both approaches. Which one you follow entirely depends on your workflow and your ability to find people who share your vision.

Regardless if you work alone or in a demogroup, you may sometimes find yourself in a bit of a pickle halfway through your demo making adventure. If you ever do, you should be aware that you're not alone and that other demosceners tend to be helpful in all sorts of ways:

1. They can lend you their insight into your problem.
2. You can randomly ask them if they are interested in contributing assets to your demo.
3. They tend to share how they have their workflow set up.
4. You can send them previews of your work-in-progress and ask for feedback on how to improve it.
5. They can lend you computers to finish or submit your entry at the partyplace.
6. They can bring you food, beer or coffee while you're busy working on an entry.
7. They might even teach you how to use tools and tricks that make your life easier.

My point is, there are plenty of others that know what you're trying to do and will gladly help you where they can. Most of them have already tackled that specific problem you're trying to solve. So why not ask them for some insight? It might save you many frustrating hours.

If you're shy to approach people, there are also recorded conference talks of demosceners covering different topics of interest to the demoscene which you can find on the internet and learn from.

By the end of the 11th day you should know the importance of getting help when you are in dire need and that demosceners are typically easy to approach on such topics.

Day 12 - Crunch Time

The good thing about a deadline is that you see it coming months ahead, so technically speaking you can actually plan for something decent and start on time.

But ofcourse that the thing about deadlines is that - unless you're a total control freak - they tend to sneak up on you. You start off thinking you have lots of time, so you plan this huge awesome project, and then as time grows shorter you start realizing you aren't actually keeping up with your workplan and that your idea needs to be replaced by something simpler.

As mentioned before, my suggestion to avoid this point of failure is to always plan realistically. Build the core of your entry early on and then spend the extra time before the deadline polishing things

up, testing it on different machines and maybe getting feedback from people whose opinion you respect?

Some folks enjoy procrastinating a lot, discussing what the project could be, over and over again, instead of actually trying things or following the plan to accomplish what was already defined. These people tend to re-open things that were already closed and done with, etc. Planning things properly is very important, but at some point you need to stop planning and just close things. This is what crunch time is for, when the deadline is looming close and you desperately want to finish your release, you enter crunch time and close things down.

Crunch time means you and your team are sitting down together to work on all the things that need to get done, and not leaving until they are all done. No interruptions except for food, bathroom and sleep.

Productivity under crunch time is often higher than average, but it will soon start to deteriorate as the crunch period extends itself. This is normal, you and your team are only human, you will get tired and start to lose focus and patience. Long crunch times can be counter productive. They can also leave you quite burned out. So learn to plan ahead and avoid unnecessary crunch time periods.

Typically crunch time should only be enforced to wrap things up: Clean the last bugs, implement the last effect, integrate the final artwork, tweak some colors and syncs.

To push the project forward some people enjoy doing their projects in semi-regular sprints. A sprint is a short period of time where you enter crunch mode totally focused on that project alone. They are great ways to kickstart a project, tackle a specific project hurdle or just add some more content to a project that's been idling.

When you're planning your project and defining some sprints or crunch time, be aware that your end result is never quite what you intended to achieve, always reserve some buffer time to redo everything, clear bugs or scrap everything and start anew. Buffer time should typically be an extra 20% or 50% of what you realistically estimated for the task. Some people are more horrible than others at predicting or controlling how much time they spend on each specific task and require bigger buffer times or a simple reality check to help them close the task and shift their focus to the next one on the plan.

Creative projects always take longer than initially expected. This is completely normal and the only thing you should do about it is to expect it. If it's not obvious enough let me lay it out explicitly, it's due to two very simple reasons:

1. any creative process involves dealing with unknowns
2. you can't estimate unknowns

A project that falls exactly onto its predicted timeframe is a project where you were surely not being creative enough. Fucking boring if you ask me, but go for whatever rocks your boat.

A good rule of thumb to deliver on time is to always keep your codebase fully playable, if you are missing assets (sound or graphics) use stand-ins, don't spend too much time on them (they will be replaced anyways), do them as horrible and quickly as possible, just make sure they are in the expected format. Then send that preview to the person responsible for the assets and ask them to replace them when ready. And focus your time tweaking something else until you hear back from them for a new version.

By the end of day 12 you should know the dangers and importance of crunch times to deliver a demoscene entry on time. Use them wisely!

Day 13 - The Real Party Is Outside

Demoparties are where demosceners gather to meet each other, finish their productions and share them on the big screen.

At some point some folks realized there was more to life then sitting inside a smelly party place finishing entries. They realized there was a whole world out there to be explored. I'm ofcourse talking about the outside of the partyplace, where the demosceners gather to meet each other, talk about how late they are running to finish their productions on time or just idle around, waiting for the compos to start on the big screen.

As it happens, inside of demoparty halls regulations tend to be more strict. You usually can't eat, drink or smoke. Sometimes the sound is too loud. In other occasions they won't let you make any sound of your own. And so, after a few hours of working on your entry, you tend to enjoy walking outside to get some air. This is the perfect opportunity to socialize with fellow demosceners who are doing the same, just outside the partyplace. And this is why demosceners are known to mention that the real party is outside.

During particularly cold winters this task can be challenging. For these occasions demosceners have repurposed the ancient sacred technology of gathering wood and setting it on fire, these are known as campfires and they become social gathering points where sceners crowd to bond and warm each other you up. They are free to join.

Demoparties typically have certain areas reserved for sleeping. These are called sleeping areas. These are not the most indicated place to eat, drink, smoke or socialize. Demosceners (like any other creature) tend to dislike being awakened when they are tired and will often complain that you the real party is outside and that you should go there. You should strongly follow their advice.

Socializing is important. Sceners are usually quite social. Don't be afraid of talking to random demosceners. Don't be afraid of introducing yourself to people whose work you admire. By experience i can tell you they will most likely appreciate your interest and share with you their thoughts on your

work and your methods. They might even give you some insight and motivation on how to improve the entry you are currently working on.

Last but not least remember not to be an ass. No one likes annoying people, be them demosceners or not. If you can't communicate without being rude keep your opinions to yourself. If you can't hold your alcohol, don't drink yourself silly. This should be common knowledge but it's always useful to keep in mind.

By the end of the 13th day you should know where the real party is located and not be afraid to go there.

Day 14 - Pouet != Scene && Scene != Pouet

After the demoparty comes the official release of your entry.

Most entries nowadays are automatically released by the party voting system. This means they will be uploaded to the internet, where dozens of people will be able to download, watch, listen and inevitably leave their public comments on your entry.

You should be aware by now that most people say things online they would never say in person. Keep in mind you cannot see them, you often don't even know them, it's quite hard to understand what they mean exactly, and quite easy to take negative feedback the wrong way. Don't fall for that trap.

Pouet.net is a website where most demoscene entries are listed and commented on by the international demoscene community. Like any online forum it's quite easy to get misunderstood and start a flamewar over the silliest of things. So this is the most important thing you should know about pouet: Pouet is not the demoscene. And also, the demoscene is not pouet.

Pouet is a place to write your opinions on each others entries, it can also be a great source of community feedback. Pouet is unexplained memes, pigs, buckets, pictures of ponies and trumpets. Don't make the mistake of reading the comments on pouet and taking them too seriously.

Demoscene on the other hand is what you can make of it, it's a supportive community, a culture, some people even call it a family or a way of life.

By the end of the 14th day you should know all about the demoscene, how to make your first entry, the importance of attending demoparties, getting releases finished in time and getting to know the community.

Above all the demoscene is about having fun doing digital art. So make some demos and have fun!

Acknowledgements

Written by ps with support of cxw, Danny and hardy. Original idea by Saga Musix.