

This page lists the generic rules for speedrun hosting at SDA. Every run submitted to SDA will be independently [verified](#) against these rules and to ensure that the submitted runs meet the site's quality requirements in terms of audio/video and gameplay.

## Contents

- [1 Common questions](#)
- [2 Fundamental rules](#)
- [3 Categories](#)
- [4 Saving and timing](#)
- [5 Hosting](#)

## Common questions

### What if I disagree with your rules?

[Discuss it](#) on the forum and explain why. We're always open to suggestions and welcome intelligent discussion. If enough people hate something, maybe we'll compromise. [It's happened before.](#)

### What if I have a rule question that is not covered here?

First, use the search function in the forum and see if someone has already asked the same question. Next, if there is a game thread in one of the sub-forums, ask there and see if you get any response. If all else fail, explain the question the best you can in the [SDA discussion](#) section of the forum. Remember that there isn't necessarily an expert among the SDA staff or the forum users at your game. In order for them to be able to help you with an answer, please consider being as clear as you can possibly be when you ask (without being too wordy...).

## Fundamental rules

- **Recording:** Speedruns hosted by SDA should be captured at the native framerate and dimensions (depending on region and platform). The video needs to contain only the gameplay footage and the first audio track needs to contain only the game audio (additional audio tracks can be added with audio commentary though). You can find more information about recording and processing game footage right here in the SDA knowledge base. You can also ask for help in the tech sub-forum if you didn't find the answer in the knowledge base.
- **Emulators and Virtualization:** We will **not** accept speedruns recorded on emulators (ZSNES, VBA, etc.). Emulators commonly allow for recording games frame-by-frame and then playing back the input at normal speed. Emulators also give the player access by default to features not present on the original consoles, such as RAM-watch or the possibility to use, for example, keyboard as input device. Finally, many emulators have minor inaccuracies in timing and slowdown that inhibit accurate comparisons between runs. Note that an exception is made for officially sanctioned emulators such as the Game Boy Player, the Virtual Console, and the GameTap. Due to problems with many official emulators, we will place them in separate categories if the known differences make comparison with runs made on other platforms difficult. For the purpose of this rule:
  - DOSBox can be used for SDA-submissions. It has seen very widespread use for re-releases of DOS-era PC games in the past years and is frequently the only reasonable choice for playing and recording those games.
  - Virtualization software (VMWare, VirtualPC, etc.) is permitted for games that wouldn't

otherwise run natively on a modern operating system.

- Programs such as WINE that attempt to replicate the Windows system calls and structure on Unix systems are not allowed.
- Flashcarts (such as Everdrive) are a grey area. In many cases there is no possibility to distinguish between the use of a flashcart and an original cart, in which case there is no reason not to accept them. However, if any differences would be discovered in the future, applicable runs can be expected to be removed from the site.
- **System modification:** You are not allowed to modify your system or use extra hardware such as GameSharks and Game Genies. These devices let you alter game parameters and can give you an unfair advantage. Modchips and boot disks used for playing imports and officially released add-ons are allowed. For example, the PS2 HDD is allowed (provided the game itself has an option to install data to the HDD), while the HD Loader is not. In addition, modifications that enhance the quality of the audio-visual output of a console are allowed.
- **Interacting with the game** other than through the gameplay itself is not allowed. This rule covers both passive methods (such as RAM-watch) and active methods (such as removing or altering a game disc/cartridge/file while the game is running).
- **Third-party controllers:** You must only use features that are available on any controllers that were officially bundled with the system. Thus, turbo-fire is not allowed except for systems such as the TurboGrafx-16 that come with official turbo-fire controllers. Of course, if a game itself provides a turbo-fire option, then it may be used. If you use an unusual or non-standard input device you should be open about it (for example mentioning it in the run comments), even if it doesn't introduce any additional features. There is probably a reason why you decided not to use a standard controller in the first place and other people trying to compare themselves with the time you have achieved have the right to know if anything was used that could be seen as a competitive edge.
- **Codes:** Using a beneficial cheat code is not allowed: something that gives more lives, reduces damage, etc. If a code is only cosmetic, like suit-less Samus in NES Metroid, that is acceptable. A code that increases the game's difficulty, such as Donkey Kong Country 3 105%, may qualify as a separate category.
- **Impossible inputs:** Some games, such as The Legend of Zelda: A Link to the Past, show unusual behavior when one feeds them with usually impossible input, such as up+down pressed simultaneously. As such actions require worn out controllers, non-standard controllers or excessive force, they are treated as hardware modification and are thus banned as well.
- Many PC games allow you to use the in-game console to write **scripts** or macros to automate certain actions. There are also external input scripting programs such as AutoHotkey. These are not permitted to be used for any form of automation in PC games. You are however free to **bind** keys in whatever way the game allows you to do through the game's menu system or config file and it's expected that you use this to your advantage (e.g. binding of a free-spinning mouse-wheel to the jump button in Half-Life games to allow a simple way of bunny-hopping).
- As a general rule, speedruns submitted to SDA are supposed to be using the game **patch** that allows the fastest possible completion of the game. SDA doesn't consider different patches to be a reason to create new categories, so make sure that you choose the correct one for your game.
- Many speedruns utilize **glitches** (unexpected behavior due to the game's programming), to the runner's advantage. Glitches are allowed and runs submitted to SDA are expected to fully utilize beneficial glitches and bugs in their runs to save time within the confines of the game's behavior. In extreme cases, glitches allow you to skip huge chunks of the game and

- that can warrant a separate category. More about that further down.
- SDA will accept hosting speedruns for many **games**, but not all. Here are some examples of games that won't be hosted:
    - Alpha/beta versions or early access of a game
    - Shareware and demos (i.e. partial full games)
    - Romhacks
    - Game mods (defined as requiring the original game in order to play)
    - Games that can't be sped up. Examples are games that run on a fixed timer or rail shooters. A rail shooter can be an acceptable game choice if there are enough gameplay elements that allow the game to be sped up compared to a normal playthrough. Examples could be lag reduction techniques, particularly long boss fights or boost abilities.
    - Games without clearly defined start and end points.
    - Games where you play against other human players.
    - Games with "morally questionable" content (extreme racism or adult content etc).

## Categories

Not all categories apply to all games and some games can even have game specific categories not mentioned here. In general, all runs must complete the game to an ending. We have grouped the categories into three groups, "segmentation", "completion percentage" and "additional category tags".

### Segmentation

SDA accepts speedruns completed either in a single sitting, or in multiple segments recorded over a longer time period.

- Games that allow you to save your progress and continue later can be done using **segments**. You can retry segments as much as you want, in order to optimize them. Keep in mind that the purpose of segmentation is not to make life easier for you or to reduce the amount of time it takes you to produce a run. A segmented run implies a higher level of risk-taking and a lower tolerance for mistakes. Use as many segments as is optimal to achieve the fastest final time. In particular, do not feel like you must use roughly the same number of segments as a run you are attempting to obsolete. There obviously needs to be continuity between segments in terms of inventory, experience points or whatever is applicable for the individual game. Note however that it's allowed for an SDA-submission to have discrepancies between two segments as long as none of it is to your advantage. This gives you the possibility to later redo segments that are in the middle of the run even in games where it's virtually impossible to end a segment with the exact same stats as before.
- Runs done in a single sitting are referred to as **single segment (SS)** runs.
- **Single segment with resets:** You are allowed to save and resume in a single segment run as long as it is part of the same game session. This will add the **with resets** tag to your run.
- For some games, it makes sense to track times for **individual levels (IL)**. Usually, this is the case if the game is separated into sections that can be individually selected and are independent of each other (for instance, no status effects or equipment carry over between levels). If there is a minor connection between levels but an obvious base state exists, ILs may be done starting from this base state (for example, the Yoshi's Island ILs all start with 0 eggs and don't permit using any of the bonus game items from the pause screen). If the game does not currently have any IL runs then all of the levels must be completed as part of the initial submission. For future improvements, each IL may be obsoleted individually.
- Some games have timed **mini-games** that can be competed in isolation from the rest of the

game. These may qualify as additional categories for games that already have a full run.

### **Completion percentage**

Beyond this basic consideration of the type of segmentation, a run usually also falls in one of the following categories:

- **any%:** This is the most basic goal of a speedrun: beating the game as quickly as possible. The aim is to reach an ending (or for an IL, the end of the level) as fast as you can manage to within the boundaries of the game. This may include skipping entire sections of the game, skipping key items, leaving the boundaries of the game world and abusing programming oversights left in the game. Most runs fall into this category.
- **100%:** A lot of games, especially ones following an open-world design, have collectibles and core items that a player may or may not pick up over the course of the run. The goal of a 100% run is not only to beat the game quickly, but also to bring it to a 'full' completion in doing so. This usually involves collecting all of the collectibles and items that grant permanent status changes as well as finishing all of the quests and/or levels in the game. This category is usually easy to define for games that track the completion rate, but other games may also have 100% runs. [This forum topic](#) contains rulings on 100% definitions for a number of games. This thread is mainly for posting the rulings, not for the discussion itself. If you wish to discuss the 100% definition for a game that's not mentioned in that thread, first post in the relevant game thread (or create one if there is none for your game) and see if you can get feedback from other people who know the game.
- **low%:** While the 100% category aims to complete a game as fully as possible, the goal of a low% run is the opposite of that: Beat the game using as few resources as possible. This is probably the least well defined of these categories and strongly depends on the specifics of each game. For games that track a completion percentage, the goal is to minimize that completion rate while still beating the game. For other games, the goal is usually to replicate that behavior, which means foregoing as many collectibles and permanent power-ups as possible. See above under "100%" for how to discuss the definition of the category for specific games. For this category, an accepted run that manages to beat the game on a lower completion rate than a currently published run will obsolete that run even if the completion time is slower.

### **Additional category tags**

In addition, there are frequently more game-specific categories and tags that are added to a run beyond those. Not every possibility will be listed here, but some of the most frequently used are:

- **Character used:** For games with multiple playable characters that don't just constitute a cosmetic difference, each playable character may constitute a separate category. However, for games with a large roster of characters (such as racing games), you are usually expected to just pick the fastest character instead of having individual categories for all character choices.
- **Difficulty used:** Many games have difficulty settings you can adjust. There are generally three difficulty levels that are different categories on SDA: Whatever is fastest, the default and the hardest difficulty level. On rare occasions, other difficulty levels may be tracked. It can for example be if an intermediate difficulty allows for specific shortcuts or glitches that significantly change the route. For games with no speedrun history, SDA is more relaxed with the choice of difficulty level. However, if the game starts to see more activity and the game page fills up, runs on "in-between" difficulties can be expected to be obsoleted in favor of more competitive difficulties. For games that allow you to change the difficulty settings during the game (such as the Left 4 Dead games), you are allowed to mix and match the difficulty in order to achieve the fastest time possible. In that case the run receives a mixed difficulty tag.

- In some games you might be able to commit suicide in order to save time by teleporting back to a location you've been already and must return to or to refill your character's resources, such as weapons or health. Any run with deaths will be placed in the **with deaths** category. This also includes games that allow you to continue after a game over.
- Some games have tricks or glitches that allow you to skip a major part of the game. In this case there will be a "**with large-skips**" category for the game (for instance, skipping entire dungeons in Zelda games).
- **with uber large-skip glitches:** Some games contain glitches that go beyond skipping portions of the game and allow you to literally skip the vast majority of the game. Credit warp runs typically go into this category.
- **Multiplayer** runs are considered as a separate category to single player runs. The more players you have, the more things can be done at the same time, but can also mean additional lag and the worn-out term "no chain is stronger than its weakest link" might also apply. There is in general no distinction on SDA between the number of players in a multiplayer run. Use as many players as you deem necessary at any given point during the run to achieve the fastest completion time.
- **New game+/DLC** means that you use stats, equipment, characters or other game features that are not available when you start a game from scratch. DLC (Down-Loadable Content) should be self-explanatory, while new game+ refers to starting the speedrun from a completed game file.
- Runs with **warps** are essentially treated as runs with large-skips, described above.

If you feel that a certain category would make for an interesting run, feel free to ask about it on the forum or contact an SDA admin about it. Please just keep in mind that the more arbitrary the rules are, the less likely it will be seen as a good fit for the site.

## Timing

Always remember that speed is the first and foremost priority; side issues such as entertainment are secondary. Someone can beat your run later simply by omitting your time waster. If you waste enough time, the verifiers will reject your run outright. Similarly, if a game lets you skip cut-scenes or advance through text quickly, then you must do so (exception being if the game has an in-game timer that doesn't run during cut-scenes – it's still preferable to skip what's skippable though).

Below follows the **general** outline for how timing is done on SDA, including some recurring special cases. It's not possible to cover every imaginable situation though, so there will still be many games that must be treated as special cases.

- The game's internal timer will generally be used unless it is inconsistent or fails to display the time after completion.
  - For games without timers, a simple real-time measure is used. There are many different types of software that can do this. Avidemux for example (be sure to use a version that allows frame count, at the time of writing, that means v2.5 or earlier).
- **Starting point:** when the player first gains control of the game's character, timing begins. This is defined as the frame before the character starts to move (in case that's hard to define, a reference point, such as fade-in can be used instead). If the game starts by going through menus that have an impact on the gameplay (adjusting stats or equipment) or an overworld map (or level selection screen etc), then those are defined as "gain of control". However, if character creation is done as an independent activity before the game starts, it's not considered as part of the run time, even when it involves defining stats, equipment or

similar. Having camera control (for example Half-Life), but no means of impacting the character's movement, is not considered gain of control for the purpose of SDA-timing.

- **Ending point:** at the end when control is lost, even if that's long after the final battle, the timing stops. This is generally defined as the frame the character freezes on. If loss of control is hard to define for a game, a reference point, such as fade-out to credits can also be used. For RPGs, we use the frame the hit points of the final hit on the final boss shows up (unless there are gameplay elements coming after that point). Possible movement that can occur during or after the ending credits does not count (unless it involves actual gameplay elements that affect the outcome).

- **Loading screens** are counted for console games, but not for PC.

For segmented runs, timing for a segment stops at the first system-dependent activity. It can be the save screen (automatic saving) or when the save menu screen appears (manual saving). The starting point for a segment depends on if you actually load a game state or if you're just back at the starting menu with your progress saved. If there is a loading of the save, we will use the frame before the loading screen fade-out as starting point. This means timing might start with a cut-scene, even though there is no character control at that point. In the second case (you start a new segment from the start menu), timing will begin as soon as you activate the menu option that starts the new segment. This is generally the option that leads to a new menu with gameplay-related, and not only cosmetic, options.

- For some speedruns, the route and strategies depend on whether the in-game timer or real-time is used. In that case, both types of run can be hosted on the site side-by-side.

## Hosting

- Speedruns accepted on SDA will be made available both on SDA's server and on archive.org.
- For each game there can be multiple categories of runs that are posted on SDA. These generally exist independently of each other and are also improved and obsoleted independently, except for when a new run in a more restrictive category beats the time of a currently posted run or when a new run in a comparable category improves on an existing run beyond the scope of the category difference. For instance, a run on an easier difficulty that significantly improves on an existing run beyond the time saved from playing on an easier difficulty level or a run on a faster version that clearly improves on an existing run more than the version difference accounts for. Cross-category obsoletions are done on a case by case basis.
- Once a speedrun has been published on SDA, it will be hosted until it's obsoleted by another run or if there are doubts about the validity of the speedrun (for example cheating, inconsistencies in segmentation or not sticking to the category rules). The obsoleted run will still be available on archive.org, though.