Energy Efficient Games: Case Studies

2025, Sustainable Games Alliance (SGA)

Resources

Microsoft Case Studies

White paper: Reducing Fortnite's power consumption

A Comparative Analysis of Energy Consumption Between the Widespread Unity and Unreal Video Game Engines

Dynamic Power States - Microsoft Game Dev

Hauke Tießen: Developing Games for Devices with Batteries

SGA Meetup Recordings

Game Industry Experts Discuss Energy Efficiency and Player Experience

Optimizing Game Performance with Torbjörn Söderman, Technical Director at GOALS

Reducing the Games Industry Climate Impact While Improving Player Experience - SGA @ Unity Montréal

Improving energy efficiency without compromising the player experience with Ubisoft and Microsoft

Pragmatic ways to optimize energy efficiency with Hauke Tießen

Tools

Intel VTune

AMD uProf

Windows RAPL driver

Watt Wiser

Xbox Sustainability Toolkit

Windows Performance Analyzer

Apple Xcode power profiler

Android power profiler

Nvidia Nsight

What can we do about player emissions?

Case studies of successful interventions

- Eco Modes: Epic & *Fortnite* – saved a wind farm worth of power; Ubisoft *For Honour; Call of Duty*
- Pause rendering + 2.5D menus: *Halo Infinite*
- Screen Dimming: Bethesda's *TES: Online*
  - *Power saving on mobile + detecting "running on battery"*
- Controller disconnect: *Minecraft*
- Hardware limitations > stability – *Battlefield 6*
- Optimising for low fan noise/heat: *Goals* Esport
- Playstation 5's new "Power Saver" mode
- Xbox Dynamic Power States (DPS) system
- Graphics options in constrained FPS modes
- Game engine differences in energy consumption

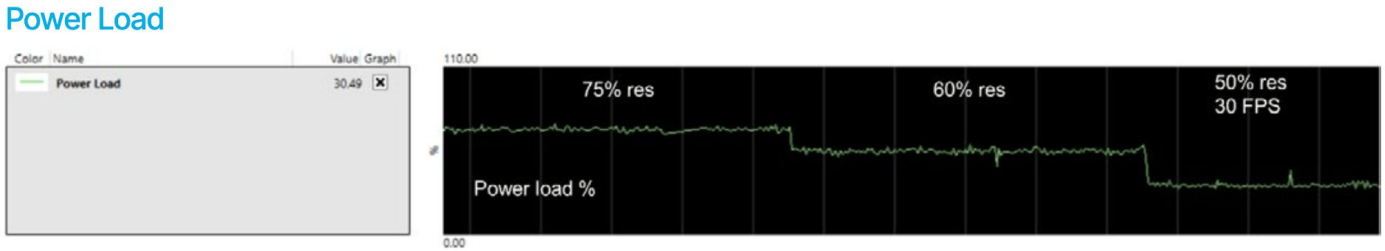
All of these examples save power *without detracting from player experience – often by improving them!*

What can we do about gamer emissions?

Eco Modes: Fortnite

Experiments

Microsoft provides a “power percentage” metric in their PIX profiling tool. This estimates the power draw of key hardware components on an Xbox Series X dev kit. We tried reducing resolution from 75% to 50% and halving the frame rate (to 30 fps) in the *Fortnite* Lobby and measured the impact in PIX. The adjusted settings reduced the power metric by around half.



The PIX power percentage metric showing different energy saving configurations on an Xbox Series X

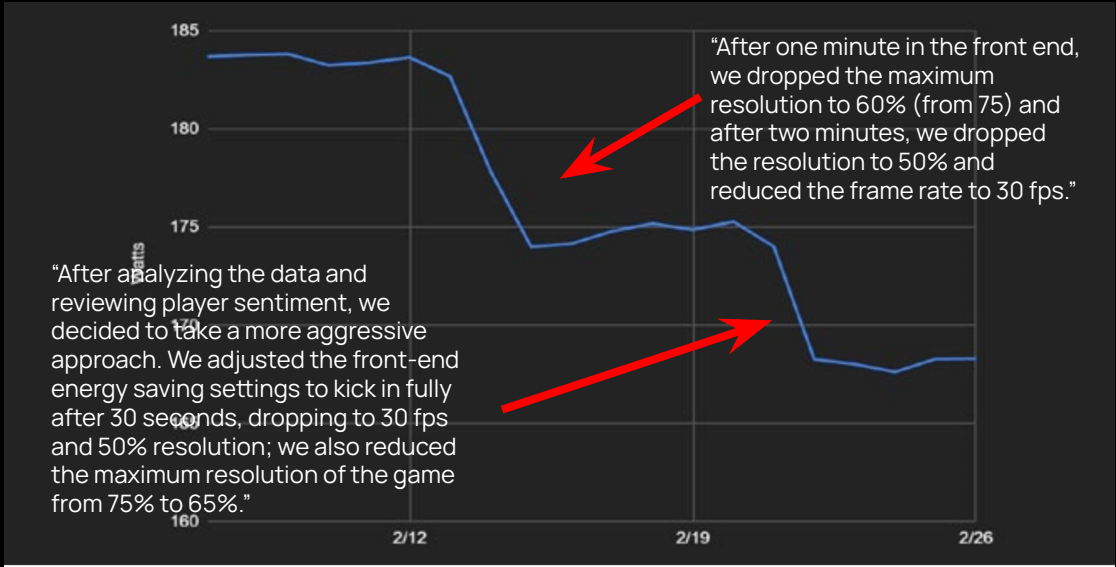
Fortnite experimented with resolution scaling & idle states (AFK on menus)

FPS and resolution are major determinants of power use

What can we do about gamer emissions?

Eco Modes: Fortnite

- Live game implementation
- “As a result of these changes, we estimate around 200 MWh per day of savings across our total player base, or 73 GWh per year (equivalent to 14 wind turbines running for a year).”
- Just from changing behaviour in *idle states!*



Average power consumption by date on Xbox Series X, showing initial and improved energy-saving configurations

What can we do about gamer emissions?

Eco Modes: Fortnite

- Epic games whitepaper has code examples of eco-mode implementation in Unreal

Implementation example

We provide some example code for handling inactivity-based energy saving in a UE5 game in front-end screens. This is intended as a starting point for a game-specific implementation. Developers may wish to expand this and incorporate support for their own game-specific logic.

The inactivity time threshold and the frame rate and resolution can be configured via console variables. This enables you to override per platform via DeviceProfile ini files:

C/C++

```
float GEnergySavingInactivityTime = 30;
static FAutoConsoleVariableRef CVarEnergySavingInactivityTime(
        TEXT("EnergySaving.InactivityTime"),
        GEnergySavingInactivityTime,
        TEXT("Idle time threshold at which energy saving kicks in (seconds). Set to 0 to disable"),
        ECVF_Default);

int32 GEnergySavingMaxFps = 0;
static FAutoConsoleVariableRef CVarEnergySavingMaxFps(
        TEXT("EnergySaving.MaxFps"),
        GEnergySavingMaxFps,
        TEXT("Max FPS for the energy saving mode. Set to 0 to disable"),
        ECVF_Default);

int32 GEnergySavingMaxScreenPercentage = 0;
static FAutoConsoleVariableRef CVarEnergySavingScreenPercentage(
        TEXT("EnergySaving.ScreenPercentage"),
        GEnergySavingScreenPercentage,
        TEXT("Max resolution percentage for the energy saving mode. Set to 0 to disable"),
        ECVF_Default);
```

What can we do about gamer emissions?

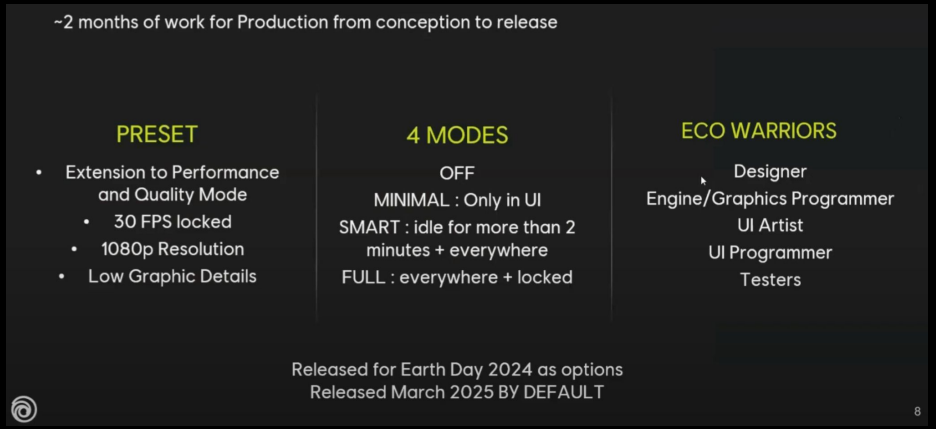
Eco Modes: validating measurements with simple power meter

- You don't need super fancy equipment – wall plug power meters are good enough!
- Any meaningful intervention that shifts power consumption will be measurable on a $20 wall plug power meter (kill-a-watt, etc)
- For already low-power devices (mobile/switch/etc) more accurate tools will be needed (and these can still be meaningful in aggregate)

What can we do about gamer emissions?

Eco Modes: For Honor

- Started with two graphics modes
- Similar process as Fortnite, testing with power meters
- Expanded to add eco modes
- If you leave it on “default” most keep it



EXISTING GRAPHIC MODES

_PERFORMANCE MODE (PS5, XBOX SERIES S/X)

- Unlocked Framerate (up to 120 or more)
- Resolution 1080p
- Low Graphic details

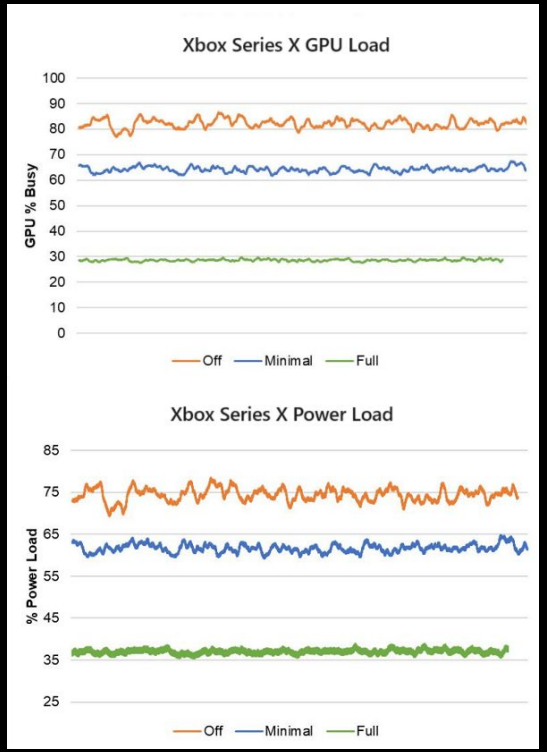
_QUALITY MODE (PS5, XBOX SERIES X)

- 60 FPS
- Resolution 4K
- High Graphic Details

What can we do about gamer emissions?

Eco Modes: Call of Duty

- Off / Minimal / Full eco modes



What can we do about gamer emissions?

Pause Menu Rendering – Halo Infinite

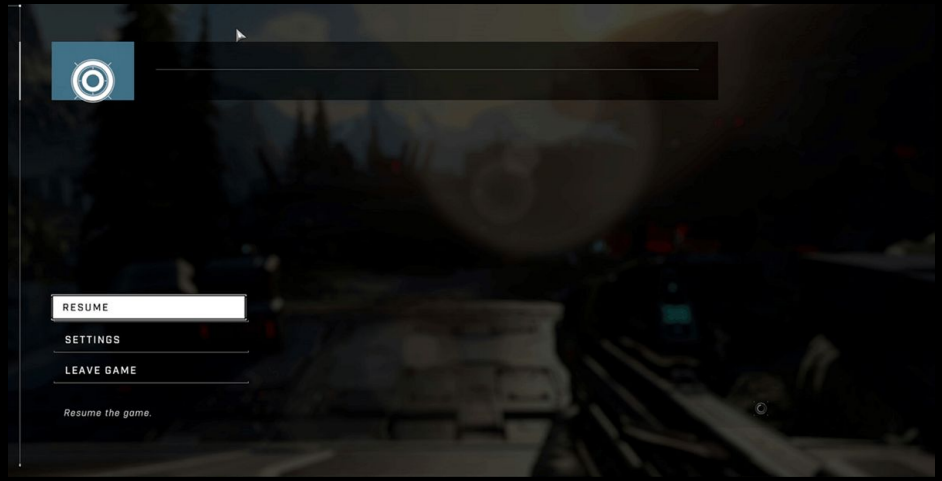
- “The image… shows our Campaign gameplay running on Xbox Series X at 4K / 60FPS. We've measured our Power Consumption with multiple devices and confirmed our total power draw was 64% of our GPU + CPU capacity, or 185 watts total AC power.”

- When you pause? It’s *still rendering the whole scene @ 4K & 60FPS*

What can we do about gamer emissions?

Pause Menu Rendering – Halo Infinite

- “The game is paused in the same location, but the resolution is automatically lowered to 1080p. We were able to measure a 55% power draw on the GPU + CPU after our change, or 165 watts total AC power.”
- 20 watts power saving for FREE
- Stopping the world render while paused, replacing it with a static image, lowering FPS/res is very easy and saves real power
- *We can go further…* with 2.5D!

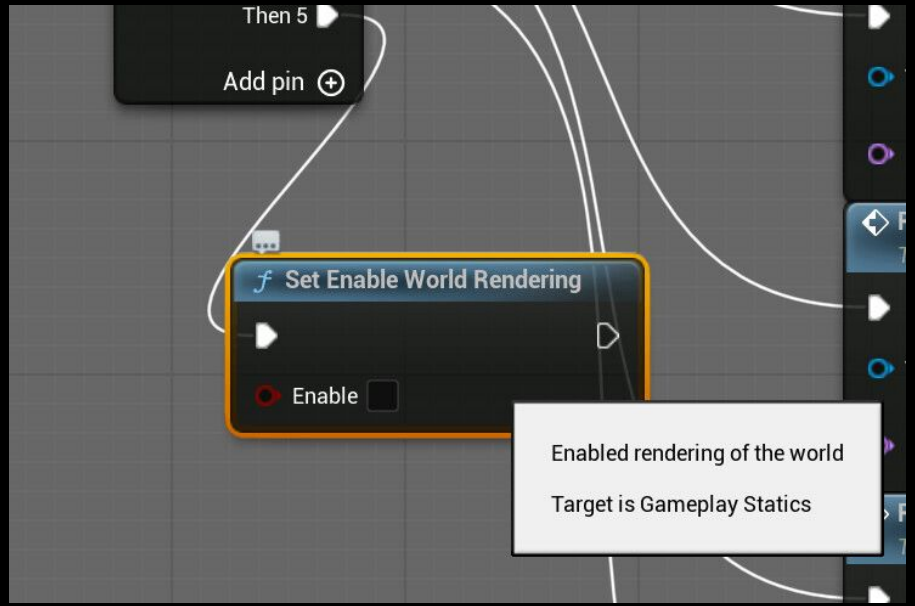


What can we do about gamer emissions?

Pause Menu Rendering – Unreal implementation

“Opening the same menu with world rendering disabled results in a 35% decrease in power consumption, from 93 to 59 watts. You could save even more energy by pausing the gameplay logic while in the menu.

If you really want to keep the transparent menus, you can take a screenshot when you open the menu and display that while the menu is open. If you don’t want to pause the game or change the background, you can reduce the rendering resolution instead. Since the game is blurred anyway, even ridiculously low resolutions are fine”



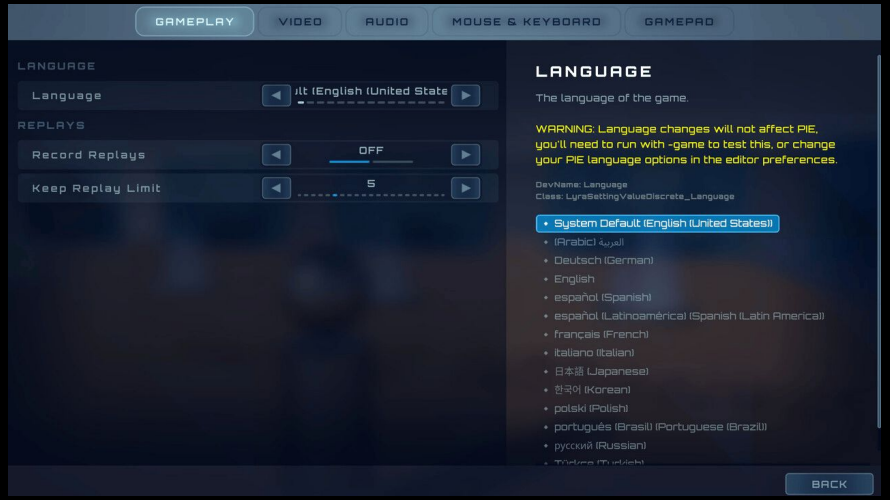
https://haukethiessen.artstation.com/blog/e1npn/developing-games-for-devices-with-batteries

What can we do about gamer emissions?

Pause Menu Rendering – Unreal implementation

“In this screenshot, the screen percentage is reduced to just 10%. In case you are not familiar with how screen percentage works in Unreal: It scales the internal resolution used to render the image. For example, if the output resolution is 2560x1440 and the screen percentage is 10%, then the internal resolution is 256x144 pixels. This means the number of pixels rendered drops by 99%!

…It’s not as efficient as disabling the rendering completely, but in my tests, the power consumption was just slightly ›60 watts.”



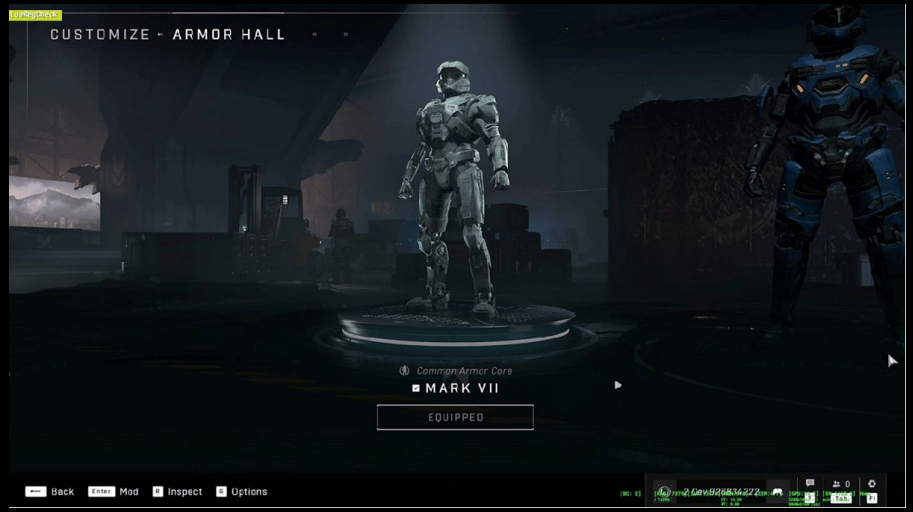
https://haukethiessen.artstation.com/blog/e1npn/developing-games-for-devices-with-batteries

What can we do about gamer emissions?

2.5D Menus – Halo Infinite

- Originally shipped with 3D menus @ 60FPS and 4K
  - Average GPU: 96.2%
  - Average Power: 67.5%

- Started by dropping FPS to 30…

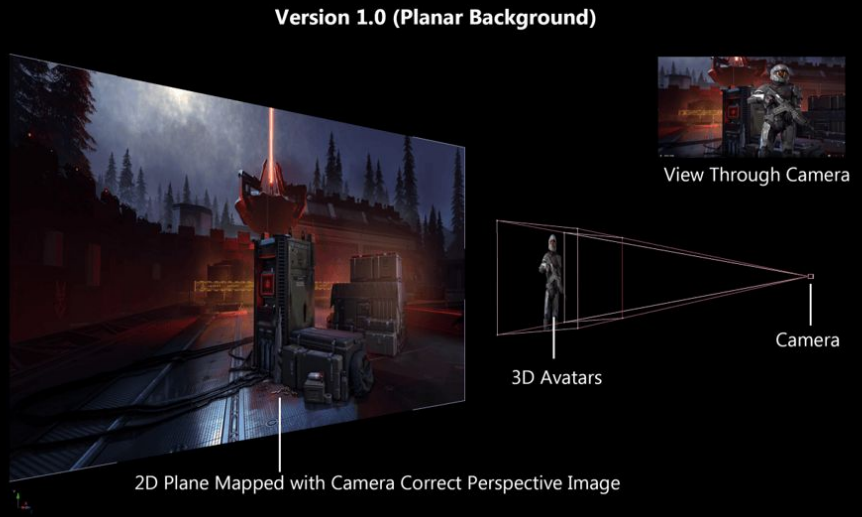


What can we do about gamer emissions?

2.5D Menus – Halo Infinite

Upsides:

- Minimal components
- Easy to upgrade
- 2D background image can be created in any tool, generated in-house or outsourced
- Load times should significantly improve
- Power consumption and energy bills for gamers' devices should be lower

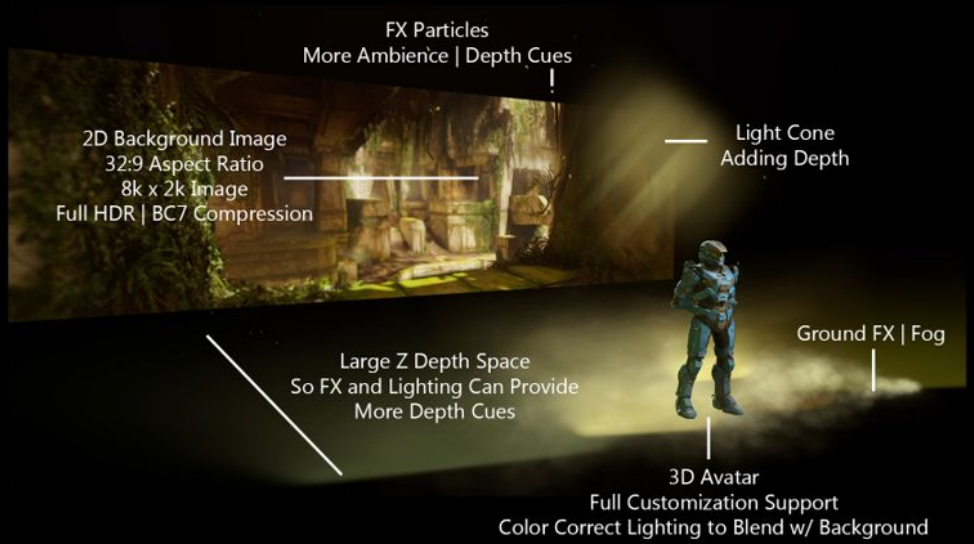


What can we do about gamer emissions?

2.5D Menus – Halo Infinite

Constraints:

- 3D avatars need to be lit to match the new 2D backplate
- Background cannot support motion
- Cannot support DOF on avatars but you can bake DOF into background image



What can we do about gamer emissions?

2.5D Menus – Halo Infinite

3D MP Menu Season 4

3D Scene 64.8%

2.5D MP Menu Season 5

2.5D Scene 47.5%

30% Power Savings | With Resolution Locked

What can we do about gamer emissions?

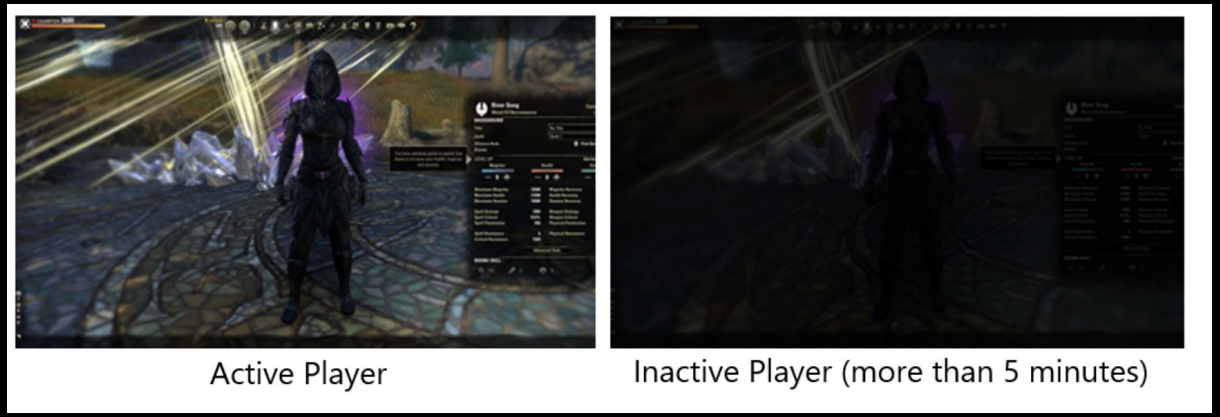
Screen Dimming: TES Online

- Screens use power:
  - OLED screens in particular!
- Dimming the screen when inactive just makes sense
- OLED “dark modes” can save ~15 watts on OLED laptops
- Dark-mode option for interface scenes?

What can we do about gamer emissions?

Screen Dimming: TES Online

- “All major consoles have some OS level setting to do this but there is no PC counterpart, so we implemented it ourselves”
- Currently no way to directly measure but the savings are real



Active Player

Inactive Player (more than 5 minutes)

What can we do about gamer emissions?

Power saving on mobile

- Reduces battery consumption (less fear of running out)
- Reduces daily charges – less wear on the battery
- Efficient games = more devices that can run it

Lesson from Roblox, Subway Surfers etc:

If you can make your game run on a potato, you can reach an incredible number of players

What can we do about gamer emissions?

Detect whether you're running on battery – Unreal

In unreal you can detect whether the device is running on battery and optimise for power life:

```
bool FWindowsPlatformMisc::IsRunningOnBattery()
{
    SYSTEM_POWER_STATUS status;
    GetSystemPowerStatus(&status);
    switch(status.BatteryFlag)
    {
    case 4://   "Critical-the battery capacity is at less than five percent"
    case 2://   "Low-the battery capacity is at less than 33 percent"
    case 1://   "High-the battery capacity is at more than 66 percent"
    case 8://   "Charging"
        return true;
    case 128:// "No system battery" - desktop, NB: UPS don't count as batteries under Windows
    case 255:// "Unknown status-unable to read the battery flag information"
    default:
        return false;
    }
}
```

https://haukethiessen.artstation.com/blog/e1npn/developing-games-for-devices-with-batteries

What can we do about gamer emissions?

Detect whether you're running on battery – Unity

Unity has a same capability.

SystemInfo.batteryStatus

Leave feedback

public static BatteryStatus **batteryStatus**;

Description

Returns the current status of the device's battery (Read Only).

See the BatteryStatus enumeration for possible values.

The battery status includes information about whether the device is plugged in to a power source and whether the battery is charging. If battery status is not available on your target platform, this property returns BatteryStatus.Unknown.

Additional resources: batteryLevel.

What can we do about gamer emissions?

Disconnected Controllers: Minecraft

- Controller inactivity = AFK
- “we lowered the frame rate when the controller disconnection notice is on screen for greater than 60 seconds.”

Device & Area	Screen Time %	Median FPS
Windows	7.1%	60 FPS
Xbox One: controller disconnected screen	14%	60 FPS
Xbox One: pause screen	1%	60 FPS
Xbox Series: controller disconnected screen	10.6%	60 FPS
Xbox Series: pause screen	0.5%	60 FPS

What can we do about gamer emissions?

Disconnected Controllers: Minecraft

Y-axis is “estimated average power consumption”

	Avg % In Active Gameplay	Avg % During Stage 1	Avg % During State 2	Avg Wattage In Active Gameplay	Avg Wattage During Stage 1	Avg Wattage During Stage 2
Xbox Series X	24%	21%	17%	102W	89W	81W
Xbox Series S	31%	26%	19%	62W	52W	48W

What can we do about gamer emissions?

Productive hardware limitations: Battlefield 6

- Optimisation for Xbox Series S hardware constraint “made the whole game more stable”
- Incredible co-benefit of targeting legacy/lower spec hardware
- Unknown effect on power – but if GPU/CPU load = power, this would *probably* also save power…
  - …but only if the CPU & GPU aren’t being blasted at 100% capacity…
  - Need an (eco-mode style) FPS/res cap
  - There *are* places you can make efficiency improvements without it translating into saving energy

VIDEO GAME NEWS

15TH SEP 2025 / 3:15 PM

Battlefield 6’s tech lead says optimizing for Xbox Series S ‘made the whole game more stable’

DEVELOPER DISCUSSES CHALLENGE OF MAKING BATTLEFIELD 6 RUN AT 60FPS ON XBOX SERIES S

Posted by
Andy Robinson

Post a comment

Battlefield 6’s technical lead says that optimizing the shooter for Microsoft’s entry-level Xbox Series S console ‘made the whole game more stable’.

What can we do about gamer emissions?

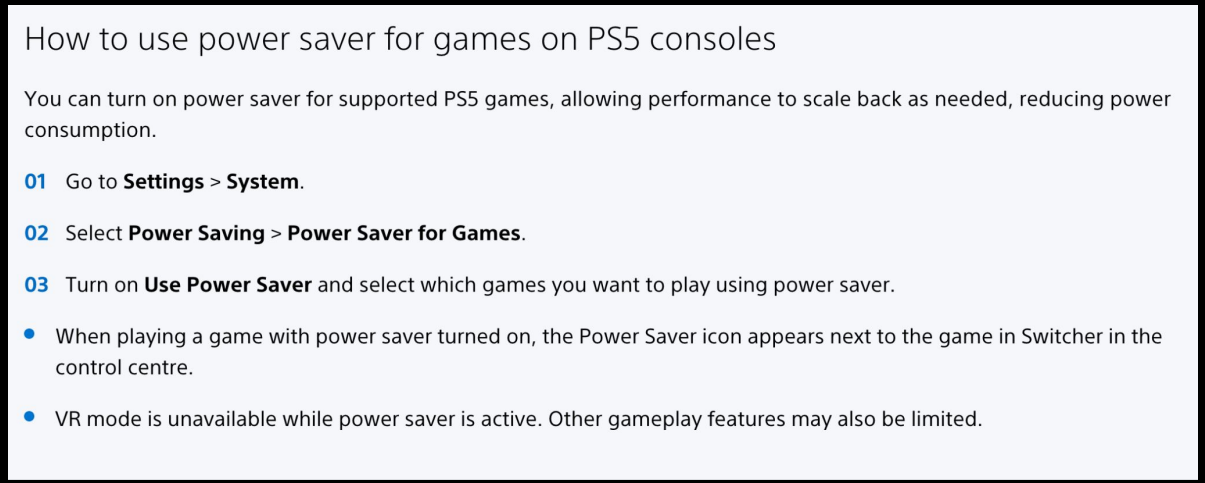
GPU Heat/Noise: GOALS

- Goals is a highly optimised game – can easily run at 500+ FPS w high end PCs
- No practical use for 500+ FPS – taxes hardware, generates heat + noise
- Technical Director Torbjörn Söderman wrote some code to monitor the heat/load on player GPUs and target a good user experience (quiet, not hot PCs!)
- Also saves power – and not just in *idle state*
- *Access recording of this method*

What can we do about gamer emissions?

Playstation 5 energy saver mode

- Just launched in Sept 2025 PS update
- Not a lot of info – *possibly* throttles or disables PS hardware (because VR mode unavailable)
- Too early to know!



How to use power saver for games on PS5 consoles

You can turn on power saver for supported PS5 games, allowing performance to scale back as needed, reducing power consumption.

01 Go to **Settings** > **System**.

02 Select **Power Saving** > **Power Saver for Games**.

03 Turn on **Use Power Saver** and select which games you want to play using power saver.

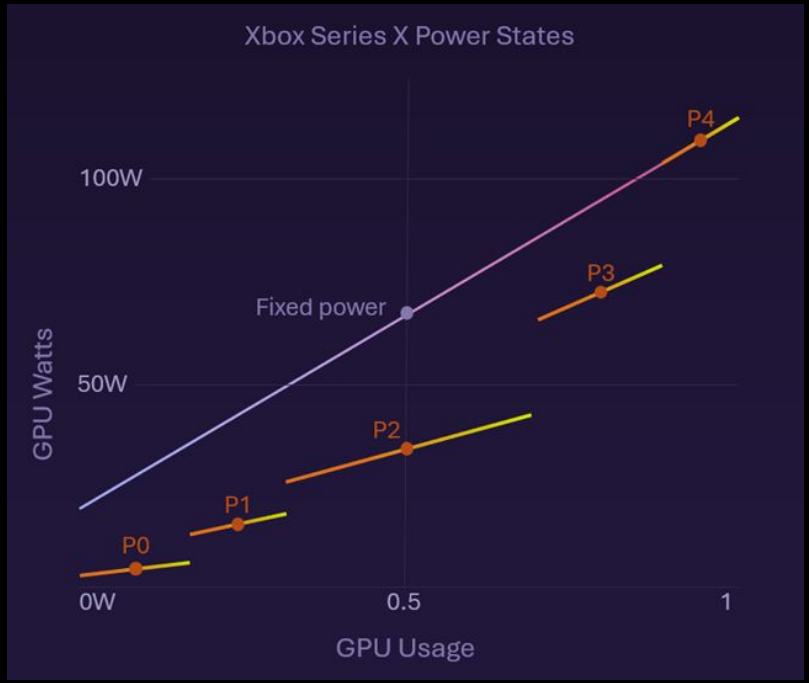
- When playing a game with power saver turned on, the Power Saver icon appears next to the game in Switcher in the control centre.
- VR mode is unavailable while power saver is active. Other gameplay features may also be limited.

What can we do about gamer emissions?

Xbox Dynamic Power States (DPS)

- “The intent of DPS is to unlock voltage savings for games without impacting performance. How do we achieve this?
  - Enable GPU dynamic power states
  - GPU calculates normalized frame statistics and stores it as historical data
  - GPU automatically changes power states based on the current game demands and the historical data, saving energy”

[Dynamic Power States - Microsoft Game Dev]



What can we do about gamer emissions?

Xbox Dynamic Power States (DPS)

Call of Duty© Black Ops 6:

"We spent a few days experimenting with DPS and the results were promising. Specifically, we could see an additional 10-15% power savings in areas beyond those we introduced using the Xbox Sustainability Toolkit. We were already throttling parts of the game engine down (for example, while in menus or front-end screen), with no change to the players' framerates, which saved us a lot of power, but DPS allowed us to get additional savings with no effort."

"At the time of shipping Black Ops 6, our current implementation plan settled on:

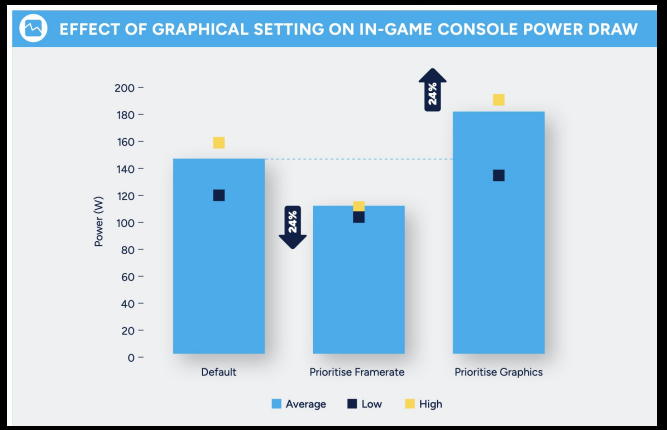
- *Enable DPS in all areas of the game where we're already applying power saving measures*
- *Expose a flag that enables DPS at all times, which is off by default. Since we are very sensitive to even the smallest drop in performance, we will be using this flag to perform some multi-day performance testing and go from there. It would be nice to turn on all the time, but only once we have sufficient data to back that decision."*

*– **Rulon Raymond, Senior Director of Technology***

What can we do about gamer emissions?

Graphics options in constrained FPS modes

In contexts where FPS is capped (e.g. at 60FPS) by hardware capability (e.g. certain televisions), graphic mode options that prioritise “FPS” can reduce energy consumption vs modes that prioritise “graphics”:



Untangling the carbon complexities of the games industry report

What can we do about gamer emissions?

Game engine energy consumption differences

Unity vs Unreal have different energy use characteristics on tests of performance on:

- Physics
- Static mesh rendering
- Animated dynamic mesh rendering

Unlikely to be actionable for most developers – but could become future points of differentiation

From preprint paper: A Comparative Analysis of Energy Consumption Between the Widespread Unity and Unreal Video Game Engines

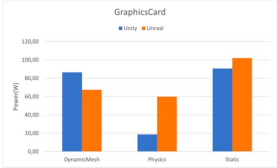


Figure 5: Graphic card power required for the three scenarios

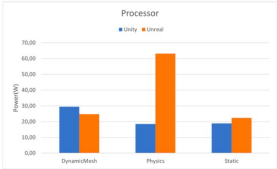


Figure 6: Processor power required for the three scenarios

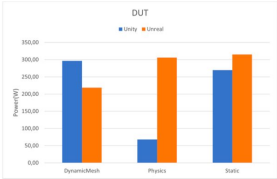


Figure 7: DUT power required for the three scenarios

Test case	Engine	HDD	GraphicsCard	Processor	DUT
DynamicMesh (W)	Unity	1,87	86,45	29,53	296,59
DynamicMesh (W)	UE	1,87	67,46	24,80	218,74
Power diff (Unity-UE)		0,01%	21,97%	16,02%	26,25%

Table 7: Power required (W) for Unity and UE (Unreal Engine). Dynamic Mesh

Test case	Engine	HDD	GraphicsCard	Processor	DUT
Physics (W)	Unity	1,34	18,82	18,50	67,92
Physics (W)	UE	1,88	59,93	63,19	306,17
Power diff (Unity-UE)		-40,92%	-218,45%	-241,49%	-350,78%

Table 8: Power required (W) for Unity and UE (Unreal Engine). Physics

5.2.1. Answering RQ1 - Physics

When asked whether there is a relationship between the energy consumption of the physics-related scenario between the versions coded in the two engines (RQ1), it is observed that there is a significant difference in consumption, with the DUT consumption of the Unreal Engine version being almost 4,5 times higher than that of the Unity version (351 %). This difference in the overall consumption of the device also appears in the consumption of the different components, (218 % for the graphics card and 241 % for the processor). For the hard disk consumption, the difference between the two engines is smaller (about 41%), but significant.

5.2.2. Answering RQ2 - Static Mesh rendering

When asked if there is a relationship between the energy consumption of the DUT in a static mesh rendering task and the game engine used (RQ2), the answer is affirmative. There is a difference of 17% in favour of the Unity version in the overall consumption of the device in this type of task. Once again, the differences in hard disk consumption are very small,(less than 2%), while the differences in this case for both the consumption of the graphics card and the processor show differences ranging from 12% to 18%.

5.2.3. Answering RQ3 - Animated Dynamic Mesh rendering

Finally, for the third research question about whether there are differences in consumption between the versions coded in Unity and Unreal engine in the rendering of dynamic meshes, the results indicate that, in this case, it is the Unreal Engine that has 26% lower energy consumption than the Unity engine. This difference in consumption is evident in the consumption of the graphics card, (22% lower in the case of Unreal Engine and 16% lower in the processor), while the consumption of the hard disk is virtually identical between the two versions.

5.3. Putting the results in context

We can state that, in light of the results obtained, there are significant differences between the versions using one or the other video game engine in the case of the scenarios analyzed. Our research has confirmed significant differences in the energy consumption of video game engines: 351% in Physics in favor of Unity, 17% in Static Mesh in favor of Unity, and 26% in Dynamic Mesh in favor of Unreal Engine.

By combining both engines and theoretically choosing the most efficient engine technology for each scenario (assuming the simplification that the three components have a similar weight in the context of a video game), it would be possible to outperform any of the two engines, as presented in Table 9. When comparing this data with a weighted average accounting

Get started on energy efficiency and decarbonization

Join us

Read the SGA statute for details
https://sustainablegamesalliance.org/join/statute/

Apply for a membership by sending a membership request to
join@sustainablegamesalliance.org


SGA
Sustainable
Games Alliance