

A man wearing a cap is focused on his work, holding a handheld electronic device with a screen and a walkie-talkie. The scene is dimly lit with a strong red light source, creating a dramatic, high-tech atmosphere. The background is blurred, showing what appears to be a workshop or laboratory setting.

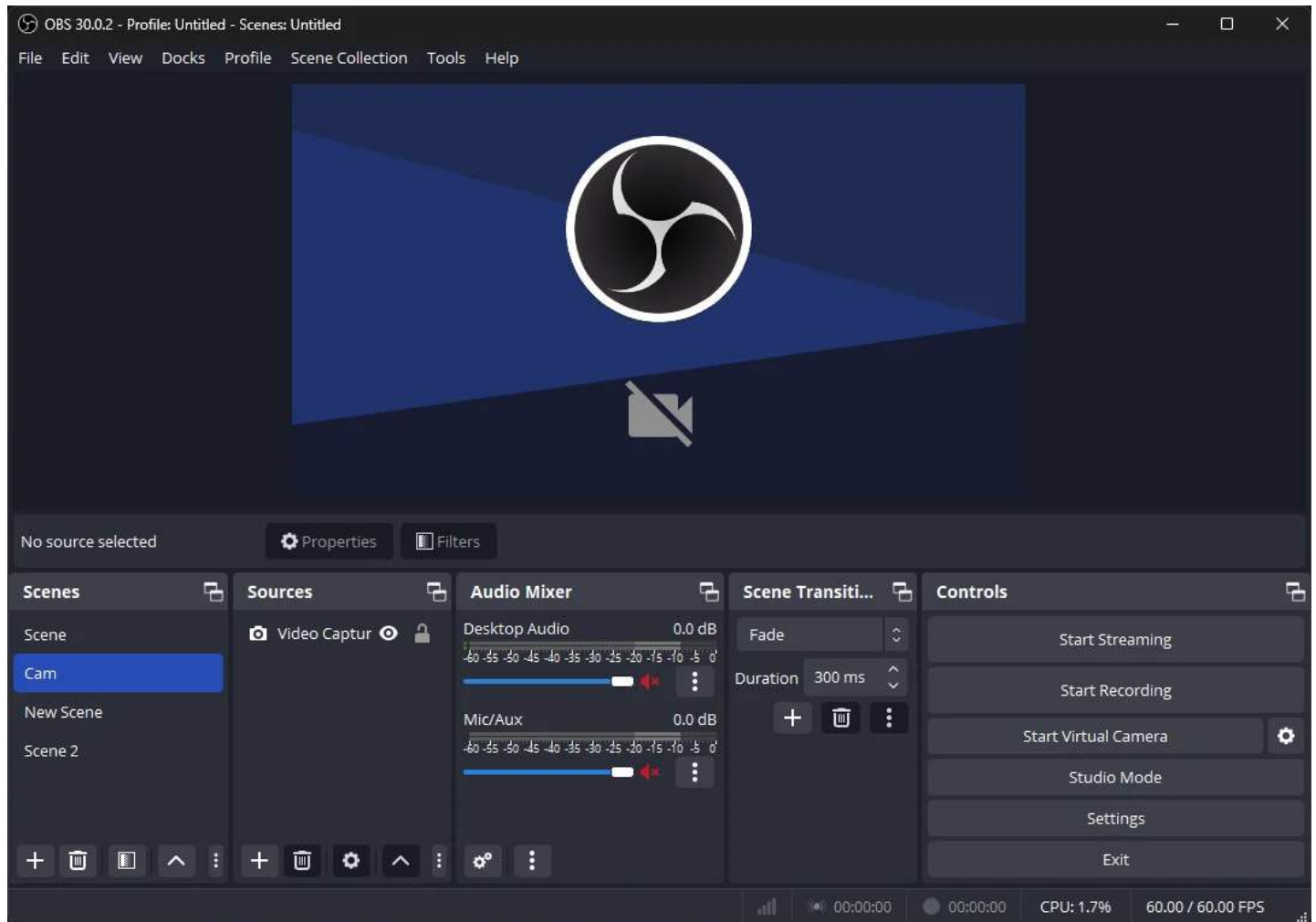
Instruct

Author: Yuri Tricys
Date: July 14, 2025

PIDP:3240

Media Enhanced Learning

TEACHING WITH OPEN BROADCAST SOFTWARE [OBS]



Open Broadcaster Software [Studio]

Introduction

Traditional text-based approaches to the creation of educational materials are limiting in terms of student learning outcomes. In modern society, educational environments are integrated with many multimedia elements, such as images, audio, video, and even interactive software, which are believed to significantly enhance educational outcomes by providing a more engaging and effective learning experience.

This is rooted in Mayer's multimedia principle, which suggests that learners absorb information better when it is presented through a combination of visual and auditory channels (Digital Learning Institute, n.d.). By leveraging multimedia, educators can create dynamic and interactive learning experiences that improve student understanding and retention.

This guide explores some of the benefits of Open Broadcaster Software Studio [OBS], a highly utilized tool in multimedia creation, as well as some of its shortcomings. OBS has gained widespread adoption due to its flexibility, customizability, and open-source nature, making it an ideal solution for educators seeking to create high-quality multimedia content.

Open BroadCaster Software Studio [OBS]

Originally developed for gamers and streamers, OBS has emerged as a leading tool for educators aiming to create high-quality multimedia content (Bunnie, 2021). OBS is available for Windows, macOS, and Linux, making it accessible to a broad range of users.

It's robust and versatile application for professional video production, live streaming, and recording. Its appeal lies in the unique blend of customizability, interactivity, and ease of access, combined with its open-source and free nature. A thriving community continuously develops plugins, enhancing its capabilities and ensuring it remains a top choice for content creators. With OBS, users can effortlessly incorporate various features to enhance their content. Notable among these are live streaming and podcasting, dynamic scene transitions and camera adjustments, and customizable keyboard shortcuts for streamlined workflows. Furthermore, OBS supports transparent video overlays, audio enhancements via filters and plugins, and automatic scene switching, making it a versatile tool for various applications, from online meetings and virtual events to vlogging and recorded content.

OBS Use Cases

OBS Studio (Open Broadcaster Software) is a powerful, free, and open-source tool with a vast range of applications for content creators, educators, and professionals. Its flexibility and plugin ecosystem make it suitable for both simple and highly customized workflows.

Key Features and Educational Benefits of OBS

- Professional Video Production
 - Live Streaming and Recording
 - Custom Audio Triggers
 - Active Plugin Ecosystem
 - Customizability
- Scene Switching
 - Automated Scene Switching
 - Audio/Video Filters
 - Accessibility
 - Interactivity



Live Streaming

OBS is widely recognized for its robust live streaming capabilities. It allows users to broadcast to platforms like YouTube, Twitch, and Facebook with high-quality video and audio. Its support for multiple video sources, real-time mixing, and encoding ensures professional-grade streams for gaming, webinars, performances, or live events (Whitehead, 2025).

Podcasting

OBS is increasingly popular for video podcasting. Creators can record or stream their podcasts with dynamic scene changes, overlays, and live audio enhancements. This approach reduces post-production time, as elements like intros, outros, and music can be triggered live, resulting in a polished, ready-to-publish episode immediately after recording (Forest, 2021).

Scene Switching and Camera Angles

One of OBS's standout features is its ability to switch between scenes—each scene can represent a different camera angle, layout, or content source. This is essential for live productions, interviews, or tutorials, where seamless transitions between speakers, screens, or media enhance viewer engagement. Hotkeys and plugins further automate and streamline scene switching, making it easy to manage complex productions solo (Shkodrov, 2022).

Customizing Keyboard Shortcuts

OBS offers extensive hotkey customization, allowing users to assign keyboard shortcuts to nearly any function—scene changes, starting/stopping streams, muting audio, and more. This speeds up workflows and enables creators to focus on their content rather than on technical controls (No Rest for the Wicked, 2024).

Transparent Video Feed Overlays

With OBS, users can add transparent video overlays—such as VTuber avatars, dynamic graphics, or lower-thirds—directly onto their streams or recordings. This is achieved through plugins and advanced settings that support alpha channels, providing a professional look and enabling creative branding or interactive elements (Valerimko, 2021).

Enhancing Audio with Filters and Plugins

Audio quality is crucial for any production. OBS supports a variety of built-in audio filters (noise suppression, compressors, EQ) and third-party VST plugins. These tools allow users to clean up background noise, balance levels, and add effects in real time, ensuring that both live and recorded content sound polished without the need for extensive post-processing (Open Broadcaster Software, 2022).

Custom Keys for Triggering Sounds and Effects

OBS can be configured to trigger custom sounds, music, or effects via hotkeys or integrated soundboard plugins. This feature is especially useful for streamers and podcasters who want to add sound effects, jingles, or stingers live, saving time in post-production and enhancing audience engagement (Open Broadcaster Software, 2022).

OBS Studio's versatility, combined with its active plugin ecosystem, makes it an indispensable tool for anyone looking to elevate their video, audio, or streaming projects (Wikipedia, n.d.)

Let's take a look at some case studies!

Case Study 1

Live-Coding a BrowserSource Overlay Server for OBS

David Wilson: The System Crafter

To enhance viewer interaction and stream quality, David Wilson, the creator of the System Crafters YouTube channel, developed innovative and engaging strategy involving a custom-coded BrowserSource overlay.



Coding his Live Server

Innovative Use of BrowserSource Overlays

Wilson's approach to OBS utilizes OBS BrowserSource functionality to display an overlaid custom-tailored HTML and CSS page, alongside live video stream of desktop software and his camera feed. The top image on the left shows David coding the live server, while the bottom image shows visible text from chat participants on the right side of the screen. Again the custom-server data stream integrates real-time chat allow viewers of the live stream to view their own comments while the stream and recording takes place.



Live Server Overlay Top Right

Custom Server for Enhanced Functionality

- TypeScript/Node.js pages
- Serves data from YouTube Live Stream API
- Integrates with OBS as BrowserSource

Real-Time Chat Integration and Benefits

- Live dynamic chat displays
- Subscriber and participant alerts
- Detailed notifications
- Real-time engagement
- More communal education

Conclusion

The System Crafter case study illustrates the transformative potential of Open Broadcast Software (OBS) in educational streaming. By architecting a custom BrowserSource overlay -- a dynamic HTML/CSS interface powered by a live server -- Wilson seamlessly integrates his desktop workflow, webcam feed, and real-time viewer chat into a cohesive broadcast. This innovative approach not only elevates stream production quality but also bridges the gap between technical instruction and audience participation, allowing learners to visually track coding processes while seeing their contributions reflected instantly on-screen.

This case study also demonstrates the general versatility of OBS. Through customization and use of OBS plugins -- provided and maintained through a large open-source community, educators can accommodate many unique use-cases, ultimately enhancing learner engagement and fostering more dynamic and appealing learning environments.

Case Study 2

Creating a Video Overlay Using OBS, Greenscreen, and Chroma Key



Kevin Stratvert

https://youtu.be/ZW_9FwBArDw

Let's look at one of the many use cases for OBS in further detail, drawing from the expertise of well-known tech YouTuber Kevin Stratvert (Stratvert, 2020).

Step 1: Acquire a Greenscreen

Purchase an inexpensive greenscreen (or a reversible green/blue screen) from retailers like Amazon for \$15–\$45. Ensure the fabric is large enough to fill the camera frame when placed behind you.

Step 2: Set Up Lighting

Proper lighting is critical for effective chroma keying:

- Position lights to evenly illuminate the greenscreen, avoiding shadows or bright spots.
- Use soft, diffused lighting to minimize wrinkles and uneven color.
- Test the setup to ensure no visible gradients or folds in the background.

Step 3: Install and Launch OBS

Download and install *Open Broadcaster Software (OBS)* from <https://obsproject.com/>. OBS is free, cross-platform (Windows/macOS/Linux), and ideal for live streaming or recording.

Step 4: Create a New Scene in OBS

- Click the **+** icon in the "Scenes" section at the bottom-left.
- Name the scene (e.g., *Green Screen*) and confirm.

Step 5: Add Video Capture Source

- In the "Sources" section under your new scene, click the **+** icon and select **Video Capture Device**.
- Choose your camera (e.g., Cam Link 4K) from the dropdown menu.
- Adjust resolution and frame rate settings as needed, then confirm.

Step 6: Apply Chroma Key Effect

- Right-click the video source in "Sources" and select **Filters**.
- Under "Effect Filters," click **+** → **Chroma Key** and name it (e.g., *Green Key*).
- Set the **Key Color Type** to green (or blue, depending on your screen).
- Adjust parameters:
 - **Key Color Spill Reduction**: Increase to reduce green halos (e.g., 120 in the example).

- Use sliders like *Similarity*, *Smoothness*, and *Opacity* to refine the key.

Step 7: Add Background Content

Insert media to appear behind the keyed video:

- Click the **+** icon in “Sources” again. Options include:
 - **Image Source** (for a static background).
 - **Browser Source** (for live or animated visuals).
 - **Window Capture** (for desktop screen-sharing).
- Browse to your content (e.g., a weather forecast image) and confirm.

Step 8: Organize Layers

- In the “Sources” list, drag the **video capture device** (e.g., your webcam feed) **above** the background media to ensure the subject appears in front.
- Resize/position the background source to cover the entire frame using the transform handles.

Step 9: Test and Refine

- Preview the scene and adjust chroma key settings or lighting to eliminate artifacts like residual green in hair or clothing.
- Recalibrate if colors spill or if edges appear jagged.

Step 10: Stream or Record

- Use **Start Streaming** (for live platforms like YouTube/Twitch) or **Start Recording** (to save an MP4 file).

This method enhances educational videos by overlaying instructors on dynamic content, ideal for web development tutorials. Proper lighting and chroma key tuning ensure professional results. Reversible green/blue screens offer flexibility, while OBS’s free tools simplify the workflow.

I Short-Comings of OBS Software

OBS Software is incredibly useful and popular; however, it has several notable shortcomings. Many users find it complex and unintuitive, especially for beginners. The software interface can be overwhelming, with numerous settings for scenes, sources, and encoding, which can lead to confusion and mistakes during setup or live production. On some machines, technical issues such as high CPU/GPU usage, rendering lag, and compatibility problems with plugins – especially on non-Windows platforms – are also common. Support is largely community-based, which can be unreliable or slow in urgent situations (Steve Sandie, 2020).

Moreover, from an instructional design perspective, OBS can encourage the use of “talking head” videos, where the instructor’s face is visible throughout the lesson. This approach can contradict Mayer’s Image Principle, which suggests that showing the instructor is not necessary for effective learning and may even distract from the instructional content. Research indicates that students often learn better from clear visuals, graphics, and animations rather than constant footage of the teacher (Ontario Virtual School, 2022).

Finally, as an educational tool, while OBS can excite, engage, and educate students, quite possibly to every level of Bloom’s taxonomy, it’s not a great tool for evaluation, as it doesn’t shine in a direct user to user way, like Zoom.

Key Shortcomings

- Steep learning curve
- Unintuitive interface
- High system resource usage (CPU/GPU)
- Limited official support
- Plugin compatibility issues (Mac/Linux)
- Tendency to promote “talking head” videos
- Too many configuration options

Conclusion

In conclusion, OBS offers transformative potential for web development education by enabling high-quality multimedia content creation. Its open-source, cross-platform flexibility supports dynamic tutorials through features like scene transitions, live streaming, and audio enhancements. Case studies demonstrate its effectiveness in integrating interactive elements and improving video production with tools like greenscreen effects.

While its complexity and resource demands pose challenges, OBS remains a robust solution for educators aiming to create engaging, visually rich materials that surpass traditional text-based methods. Strategic use of OBS can significantly enhance learning experiences, making it a valuable asset in modern digital education.

References

Wilson, D. (2020, August 28). *Live coding a BrowserSource Overlay Server for OBS* [Video]. System Crafters.

<https://www.youtube.com/watch?v=rqwMpgTCOuM>

Digital Learning Institute. (n.d.). Mayer’s 12 Principles of Multimedia Learning. Retrieved from

<https://www.digitallearninginstitute.com/blog/mayers-principles-multimedia-learning>

Bunnie, B. (2021, April 5). Using Open Broadcaster Software to Deliver Better Lessons. Root & Stem Magazine.

<https://rootandstem.ca/using-open-broadcaster-software-deliver-better-lessons/>

Stratvert, K. (2020, October 27). *How to use OBS green screen: Chroma key tutorial* [Video]. YouTube.

https://youtu.be/ZW_9FwBArDw

OBS Studio. (2024, February 9). *How to stream on YouTube with OBS*. [Image] <https://www.obsstudio.net/stream-on-youtube-with-obs/>

Loftus, K. (n.d.). [Cover photo]. Unsplash. <https://unsplash.com/photos/man-in-black-and-red-long-sleeve-shirt-holding-black-video-camera-6bQtPRQT6oI>

Brame C. J. (2016). Effective Educational Videos: Principles and Guidelines for Maximizing Student Learning from Video Content. *CBE life sciences education*, 15(4), es6. <https://doi.org/10.1187/cbe.16-03-0125> [Retrieved from:

<https://pmc.ncbi.nlm.nih.gov/articles/PMC5132380/>]

Whitehead, J. (2025, May 12). How to use OBS studio for professional video streaming in 2025. *Da Cast*. [Retrieved from:

<https://www.dacast.com/blog/how-to-use-obs-professional-video-streaming/>]

Forest, M. (2021, December 27). Recording a live video podcast with OBS. *Squares TV*. [Retrieved from: <https://squares.tv/posts/recording-a-live-podcast-with-obs>]

Shkodrov, D. (2022, May 17). What are OBS scenes and how to use them + examples. *Streaming Video Provider*. [Retrieved from: <https://www.streamingvideoprovider.com/blog/obs-scenes.html>]

No Rest for the Wicked. (2024, November 13). How to create custom keyboard shortcuts in OBS studio | Easy guide | 2025 [Video]. *YouTube*. [Retrieved from: <https://www.youtube.com/watch?v=XrzNn6jRiXA>]

Valerimko. (2021, October 24). Does OBS Support Transparent Video. *Reddit*. [Retrieved From: https://www.reddit.com/r/obs/comments/qtn6tb/does_obs_support_transparent_video/]

Open Broadcaster Software. (2022, January 12). VST 2.x plugin filter. *obsproject.com*. [Retrieved from: <https://obsproject.com/kb/vst-2-x-plugin-filter>]

Open Broadcaster Software. (2022, February 17). Soundboard dock 2.0.0. *obsproject.com* [Retrieved from: <https://obsproject.com/forum/threads/soundboard-dock.153576/>]

Sande, S. (2020, July 13). OBS Chapter 6: Open Broadcaster Software Limitations and Alternatives. Rocket Yard. <https://eshop.macsales.com/blog/63712-obs-chapter-6-open-broadcaster-software-limitations-and-alternatives/>

Ontario Virtual School. (2022, April 22). How OVS Applies Mayer's 12 Principles of Multimedia Learning. Ontario Virtual School. <https://www.ontariovirtualschool.ca/how-ovs-applies-mayers-12-principles-of-multimedia-learning/>

AI Models Used In This Report: The ideas, structure, writing, and editing in this paper were performed by the author. Various AI models were used in collecting data, verifying data, and formatting various arguments. Models used include: Perplexity Sonar, Llama 3.1 8B, Meta Llama 4 Maverick, Meta Llama 4 Scout, Meta Llama 3.1 405b Instruct, Cognitive Computations Dolphin, Mistral 24b Venice Edition, Mistral AI Devstral Small 2505, Qwen 2.5 235b a22b, Google Gemma 3 27b, Qwen 14B, Llama 3.1 8B, Llama 3.2 Vision, Claude 3.5 Haiku, Deepseek R1 Distill Llama 70b, Qwen 14B.