

Best practices for developers

URL: <https://docs.grassfish.com/docs/best-practices-spot-development>

Archiviert am: 2025-07-17 18:42:32

In this article, you'll gain an understanding of the key requirements and best practices when working with HTML spots.

By keeping these recommendations in mind during the development process, you can ensure that your HTML Basic and HTML Wizard spots are properly optimized for performance, display, and compatibility with target hardware and browsers.

Check resolution and color

When creating HTML Basic and HTML Wizard spots, it's essential to design them according to the resolution in which they will be displayed. This ensures optimal performance and user experience across different devices and platforms.

1:1 playback for media

Ensure that media is designed for 1:1 playback. This is especially important for HTML Basic spots, as they are not automatically scaled by the player. Any content that exceeds the designated resolution will be truncated, leading to display issues. To avoid this, make sure that media elements are optimized for the desired resolution.

Background color consistency

The background color of the website should match the background color of the playlist in the player to avoid flickering. A mismatch in background colors can lead to visual issues during transitions or content loading.

Test your HTML spots

Testing is an important part of spot development to make sure that your spots look and work as desired.

Target device tests

We recommend to test each HTML spot directly on the target device to ensure it displays correctly. This helps identify and resolve any device-specific issues before deployment. Testing on the actual hardware is critical to ensure the correct resolution, media quality, and performance.

Consider player performance and stability

It's important to consider the performance and stability of your player(s) when developing HTML spots.

Performance considerations

- Your HTML spots should be optimized for performance, particularly in relation to memory management. Note that poorly programmed HTML spots can negatively impact the stability of the player. The performance of your player depends on the device's hardware and performance issues can vary across devices.

- To maintain a stable and smooth user experience, be mindful of the impact that HTML spots have on memory usage, CPU load, and other system resources.

Choose media and animations

Media and animations are an integral part of most HTML spots. However, consider the following points when choosing media and animations.

Media formats

- Videos: we recommend using MP4 format with H.264 encoding for video content. This format ensures broad compatibility and efficient playback across various devices and browsers.
- Images: we support PNG and JPG. Both formats are widely supported and offer good quality with optimized file sizes for web use.

Hardware and browser considerations

Consider the hardware and browser that the player uses when choosing media and animations. More complex media or animations may cause performance issues, so it's essential to test your HTML spots on the target hardware and browser. Early testing also helps identify and mitigate performance issues before they affect the user experience.

Animation optimization

While animations can enhance the user experience, we recommend to implement them carefully to avoid excessive resource consumption. Use lightweight animation techniques and consider the performance limitations of the player hardware and browser.

Choose data formats

Note that data files are not uploaded as spots in IXM One and must be stored on the server as external data files. Ensure that the data files are properly linked to the HTML spots for correct functionality.

JSON and XML

We recommend using JSON or XML formats. These formats are well-suited for structured data and widely supported across platforms.

Consider the browser

The browser on both Windows and Linux players is Chromium Embedded Framework (CEF). It's crucial to optimize your website for compatibility with CEF and test it there.