# Article information:

Streamline your container build and publish with .NET 8 - .NET Blog
<https://devblogs.microsoft.com/dotnet/streamline-container-build-dotnet-8/>

# Article summary:

1. .NET 8 introduces improvements for building and using containers, including non-root images, smaller image sizes, and built-in image publishing.

2. The recommended approach for container publishing is using dotnet publish, which leverages MSBuild to make decisions on the best base image to use.

3. The article provides demos on how to publish container images using dotnet publish, produce images for specific distros with the SDK, and build globalization-friendly chiseled images.

# Article rating:

May be slightly imbalanced: The article presents the information in a generally reliable way, but there are minor points of consideration that could be explored further or claims that are not fully backed by appropriate evidence. Some perspectives may also be omitted, and you are encouraged to use the research topics section to explore the topic further.

# Article analysis:

The article "Streamline your container build and publish with .NET 8" on the .NET Blog provides a detailed overview of the improvements in container building and publishing with .NET 8. While the article offers valuable information on how to use dotnet publish to create container images, produce images for specific distros, and build globalization-friendly chiseled images, there are some potential biases and areas that could be improved.

One potential bias in the article is its heavy promotion of using dotnet publish as the recommended approach for container publishing. While dotnet publish may indeed simplify the process of creating container images, it is important to note that Dockerfiles remain popular among developers. The article briefly mentions Dockerfiles but does not provide an in-depth comparison between using dotnet publish and Dockerfiles. This lack of balanced information could lead readers to believe that dotnet publish is the only or best option for container publishing.

Additionally, while the article highlights the benefits of using chiseled images for globalization-friendly applications, it fails to mention any potential drawbacks or limitations of this approach. It would have been helpful to include information on any performance implications or compatibility issues that may arise when using chiseled images.

Furthermore, the article focuses primarily on showcasing the capabilities of .NET 8 for container building and publishing without delving into potential challenges or complexities that developers may encounter. It would have been beneficial to include a section on common pitfalls or troubleshooting tips for working with containers in .NET 8.

Overall, while the article provides useful insights into streamlining container build and publish processes with .NET 8, it could benefit from a more balanced presentation of different approaches, considerations of potential risks or limitations, and a more comprehensive exploration of all aspects related to container development with .NET 8.

# Topics for further research:

* Differences between using dotnet publish and Dockerfiles for container image creation
* Performance implications of using chiseled images in containerization
* Compatibility issues with chiseled images in container development
* Common pitfalls in container building and publishing with .NET 8
* Troubleshooting tips for working with containers in .NET 8
* Best practices for container development with .NET 8

# Report location:

<https://www.fullpicture.app/item/2e0b26636c7819f98ca2450e44bc5623>