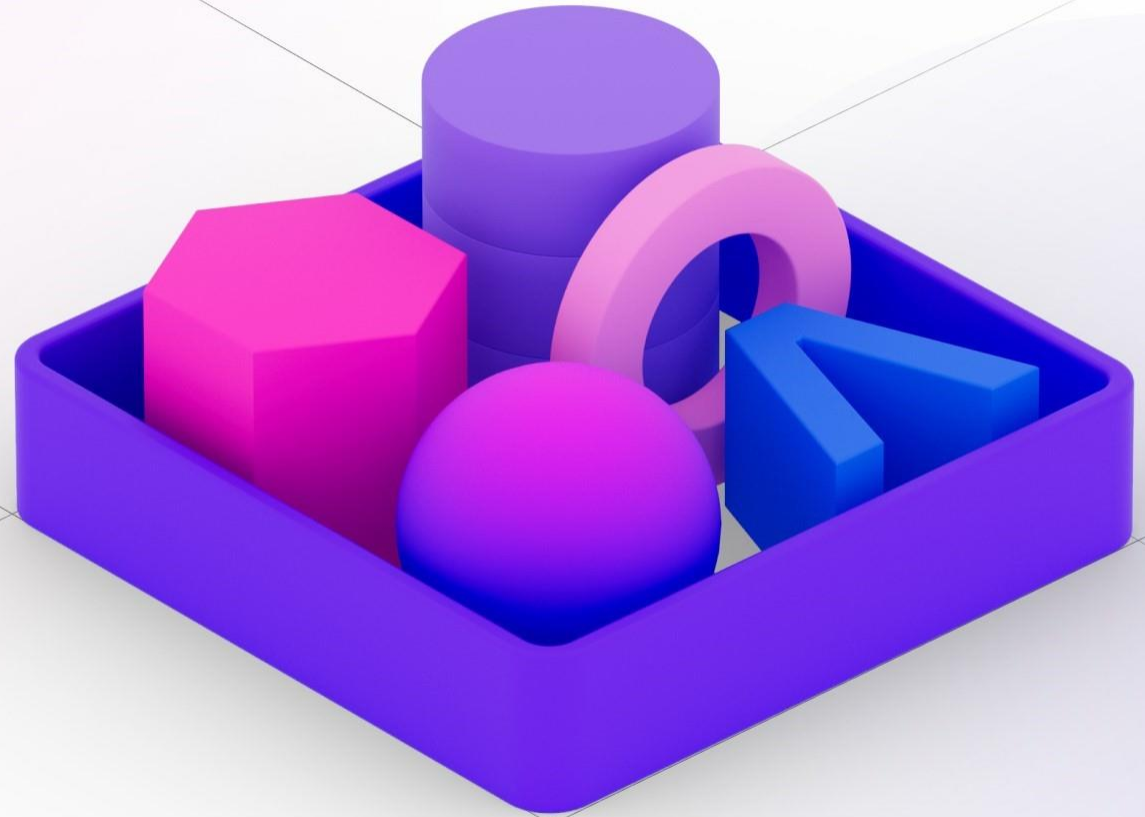


.NET Conf TAIWAN



有在關注 AOT 嗎？ You Should

Kevin Yang

Agenda

- What's AOT?
- What should we care about AOT?
- How do we use AOT?
- What bout Container?

.NET Application Compilation & Execute



What is native AOT compilation?

A publishing process that compiles .NET apps “ahead of time” (AOT) to native code

.NET Application AOT Compilation & Execute



Non-AOT vs. AOT

Non-AOT



AOT



How does native AOT work?

- C# is compiled to IL on build
- IL is compiled to platform code on publish
- Published app:
 - Has no JIT
 - Still contains a runtime & GC (is still “managed”)
 - Is single-file
 - Is trimmed to reduce app size
 - Is OS & architecture specific, e.g. linux-x64

AOT Benefits

- Minimized disk footprint
 - Reduced startup time
 - Reduced memory demand
- = Better performance

<https://aka.ms/aspnet/nativeaot/benchmarks>



source: <https://learn.microsoft.com/zh-tw/aspnet/core/fundamentals/native-aot?view=aspnetcore-8.0>

ASP.NET Core and native AOT compatibility

Not Supported

- MVC
- Blazor Server
- SignalR
- Authentication

Partially Supported

- Minimal APIs

功能	完全支援	部分支援	不支援
gRPC	✓		
最小 API		✓	
MVC			✗
Blazor Server			✗
SignalR			✗
驗證			✗ (很快就會支援 JWT)
CORS	✓		
HealthChecks	✓		
HttpLogging	✓		
當地語系化	✓		
OutputCaching	✓		
RateLimiting	✓		
RequestDecompression	✓		
ResponseCaching	✓		
ResponseCompression	✓		
Rewrite	✓		
工作階段			✗
Spa			✗
StaticFiles	✓		
WebSocket	✓		

AOT Limitations

- No dynamic loading, for example, `Assembly.LoadFile`.
- No run-time code generation, for example, `System.Reflection.Emit`.
- No C++/CLI.
- Windows: No built-in COM.
- Requires trimming, which has limitations.
- Implies compilation into a single file, which has known incompatibilities.
- Apps include required runtime libraries (just like self-contained apps, increasing their size as compared to framework-dependent apps).
- `System.Linq.Expressions` always use their interpreted form, which is slower than run-time generated compiled code.
- Not all the runtime libraries are fully annotated to be Native AOT compatible. That is, some warnings in the runtime libraries aren't actionable by end developers.

Impact of no JIT compilation

- No runtime code generation
 - No platform optimizations
 - No Dynamic PGO
- No Assembly.LoadFile
- No Expression compilation
- No Reflection.Emit

Impact of trimming

- Unreferenced code (no called) is removed.
- No assembly or type scanning
- Code might be kept that isn't called at runtime due to API design

Other considerations

- Requires extra build-time pre-requisites
 - Visual Studio C++ tools on Windows
 - Clang on Linux
 - Xcode on macOS
- Cannot publish cross-platform

How do we use AOT?

Demo Time



Publishing options

- Framework-dependent (FD) – Default
- Self-contained (SCD)
- Trimmed
- ReadyToRun (R2R)
- Single file
- Native AOT

```
<PublishSelfContained>true</PublishSelfContained>  
<PublishTrimmed>true</PublishTrimmed>  
<PublishSingleFile>true</PublishSingleFile>  
<PublishReadyToRun>true</PublishReadyToRun>  
<PublishAot>true</PublishAot>
```

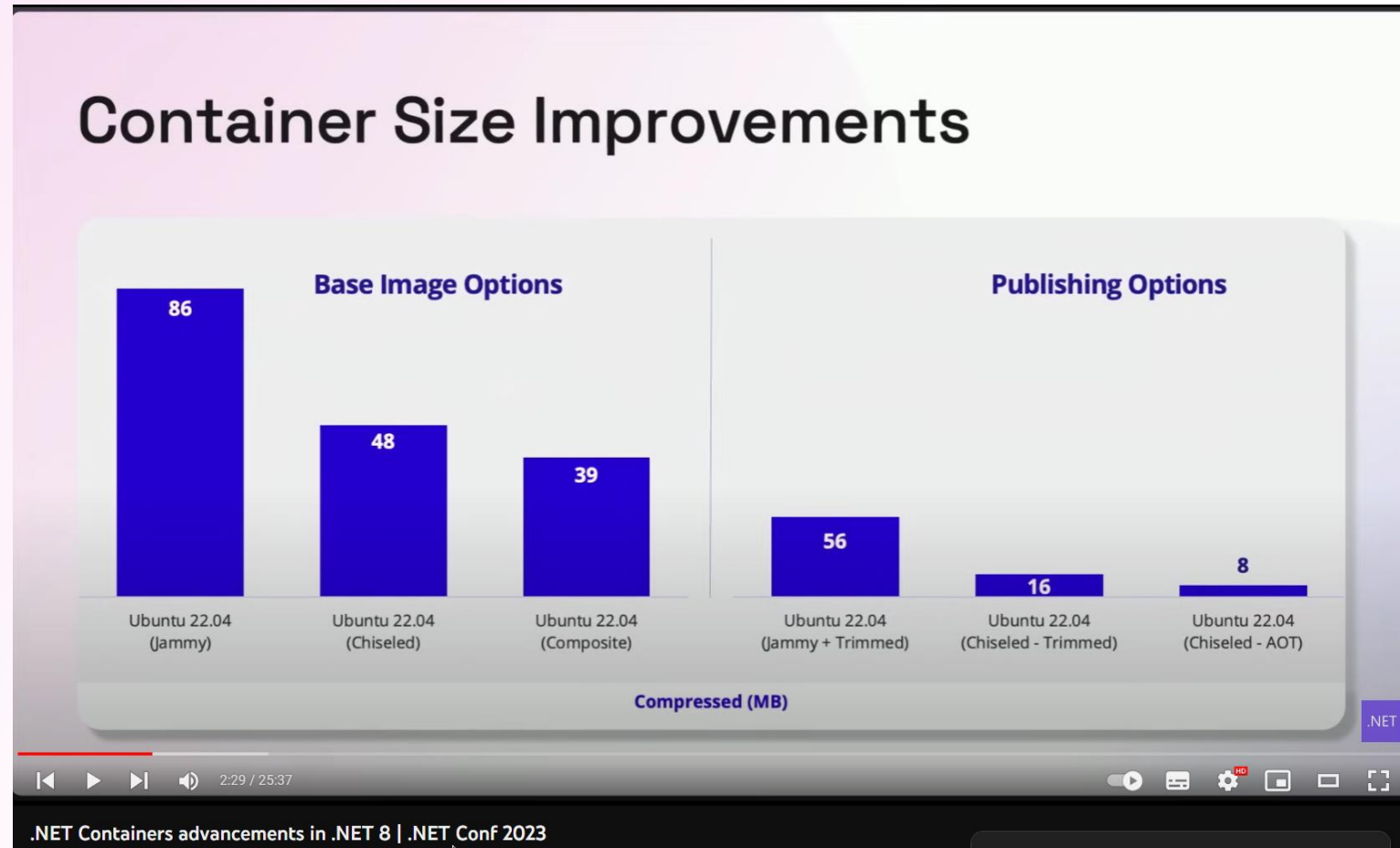

Containers



When dealing with Images...

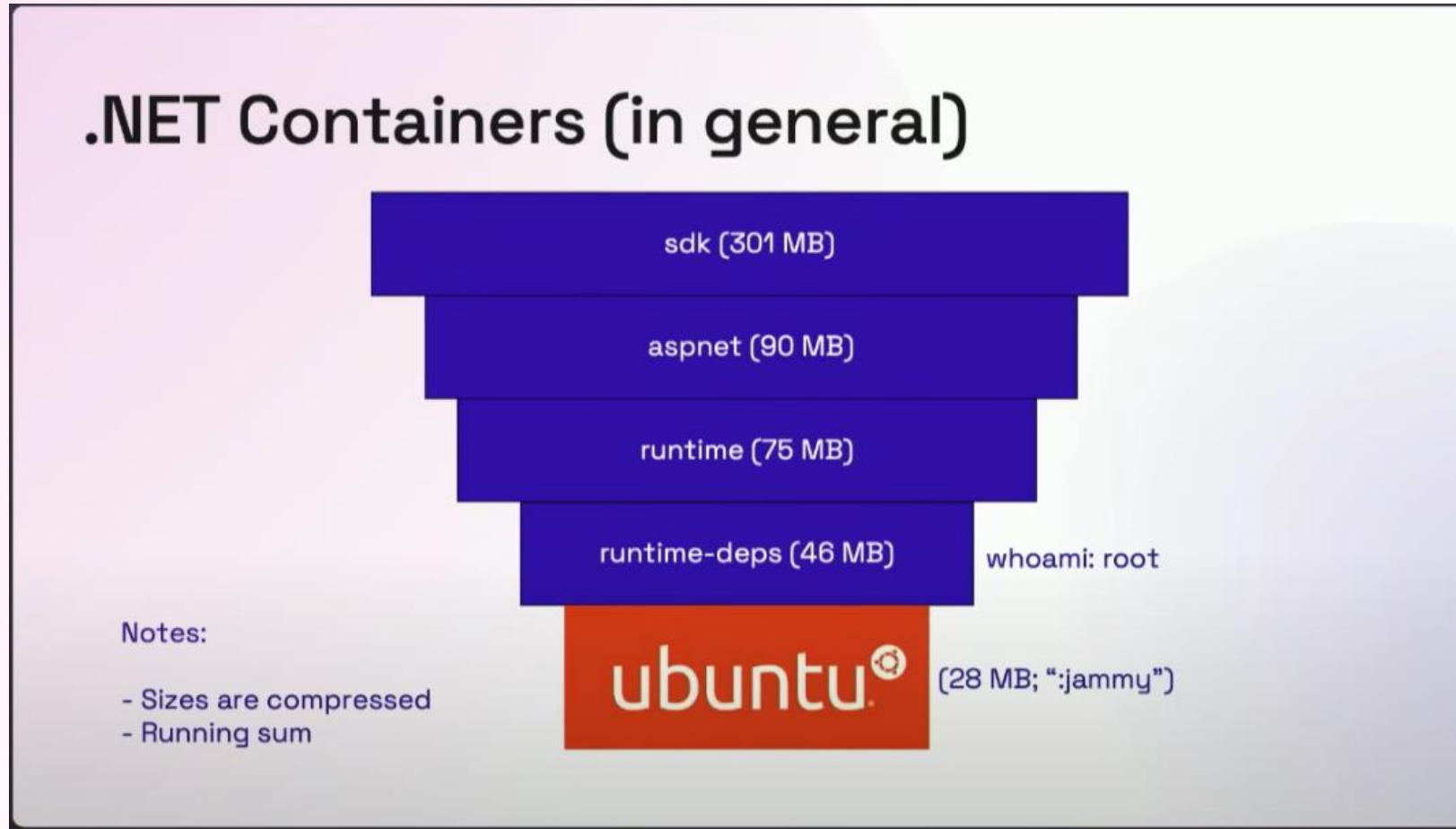
- Image Size
- Security Issues
 - CVE
 - non-root user
- Lib dependencies
- OS
-

Container Size Improvements



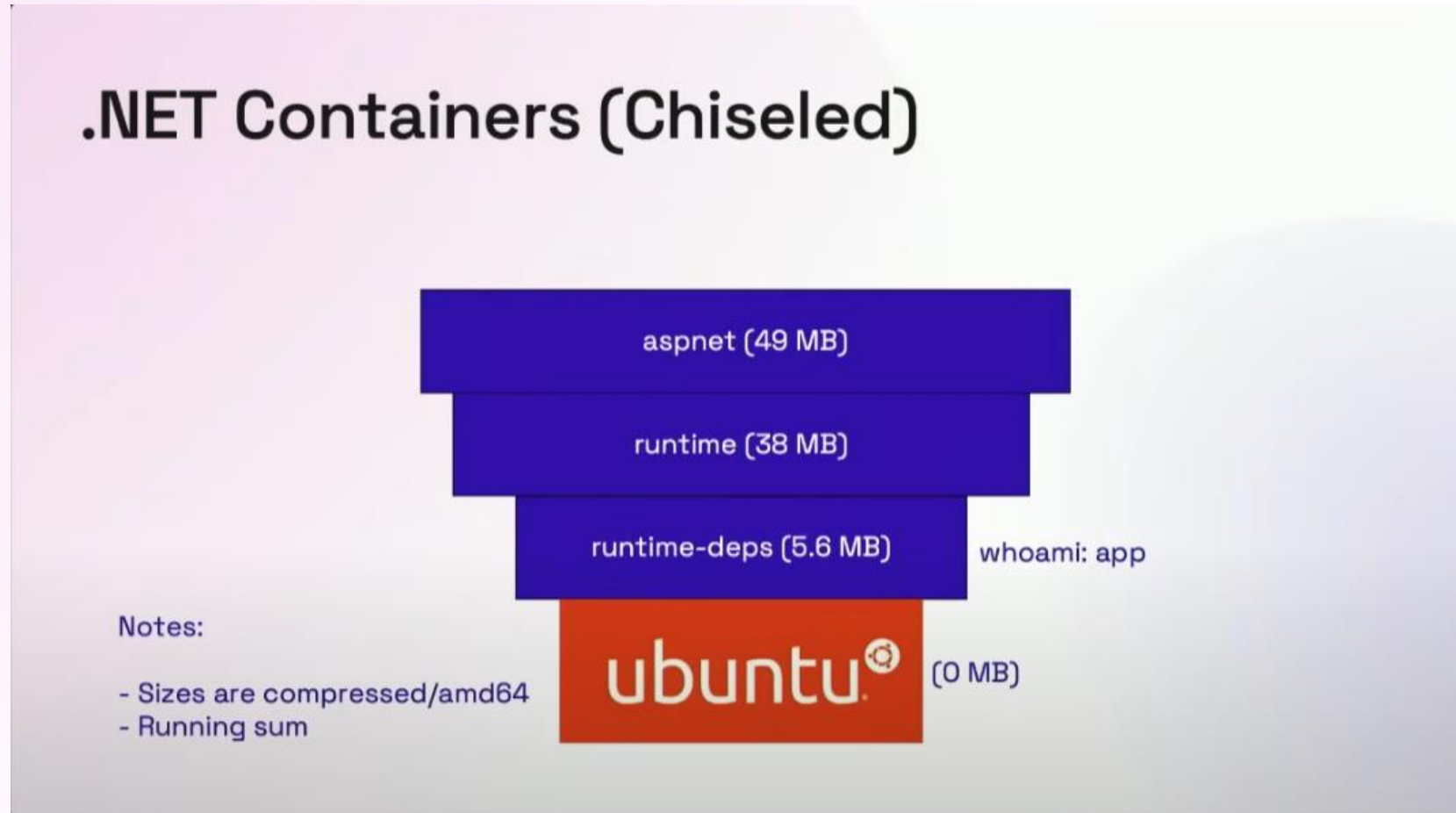
Reference: <https://www.youtube.com/watch?v=sclAwLrruMY>

.NET Containers (in general)



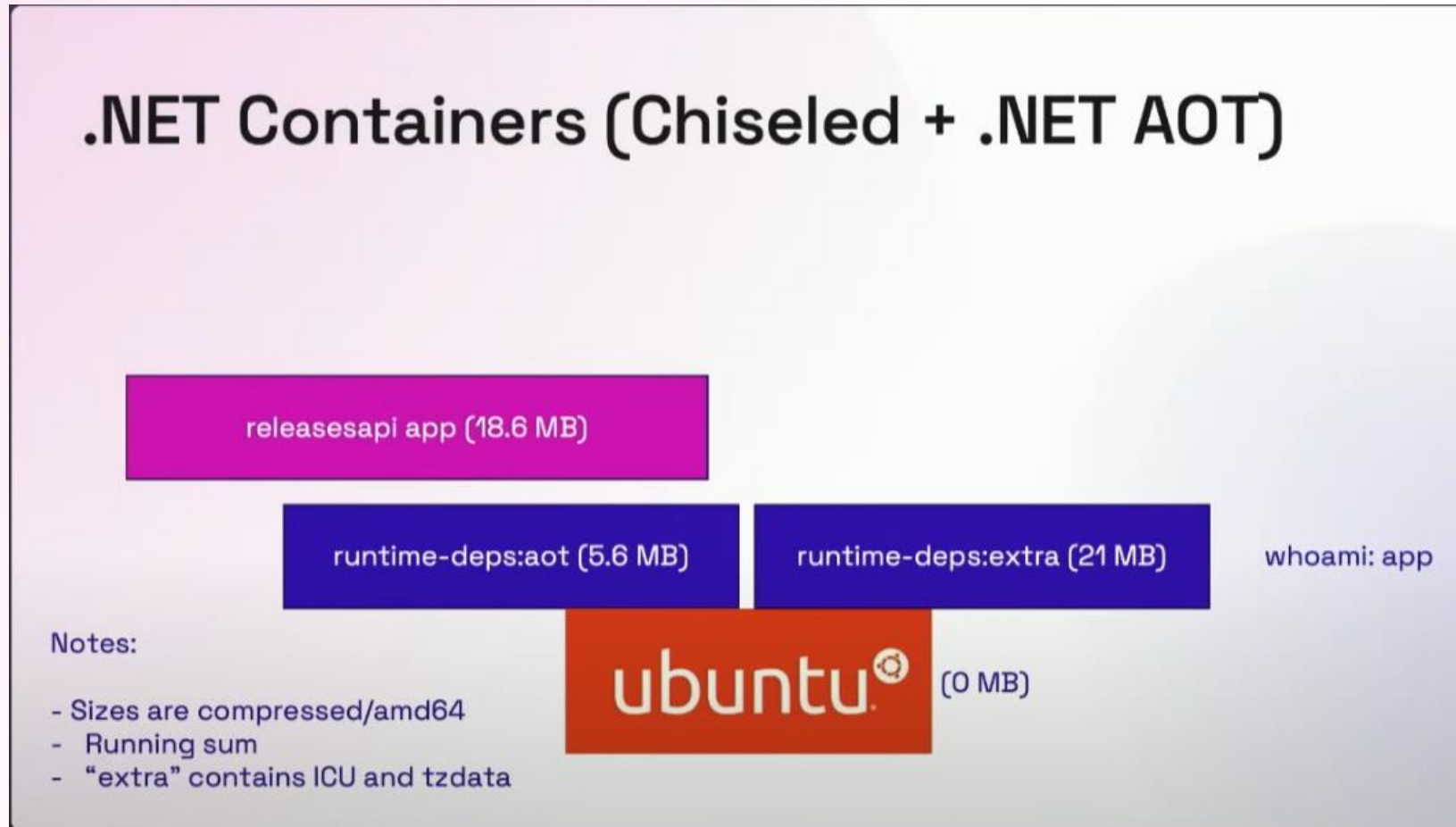
Reference: <https://www.youtube.com/watch?v=sclAwLrruMY>

.NET Container (Chiseled)



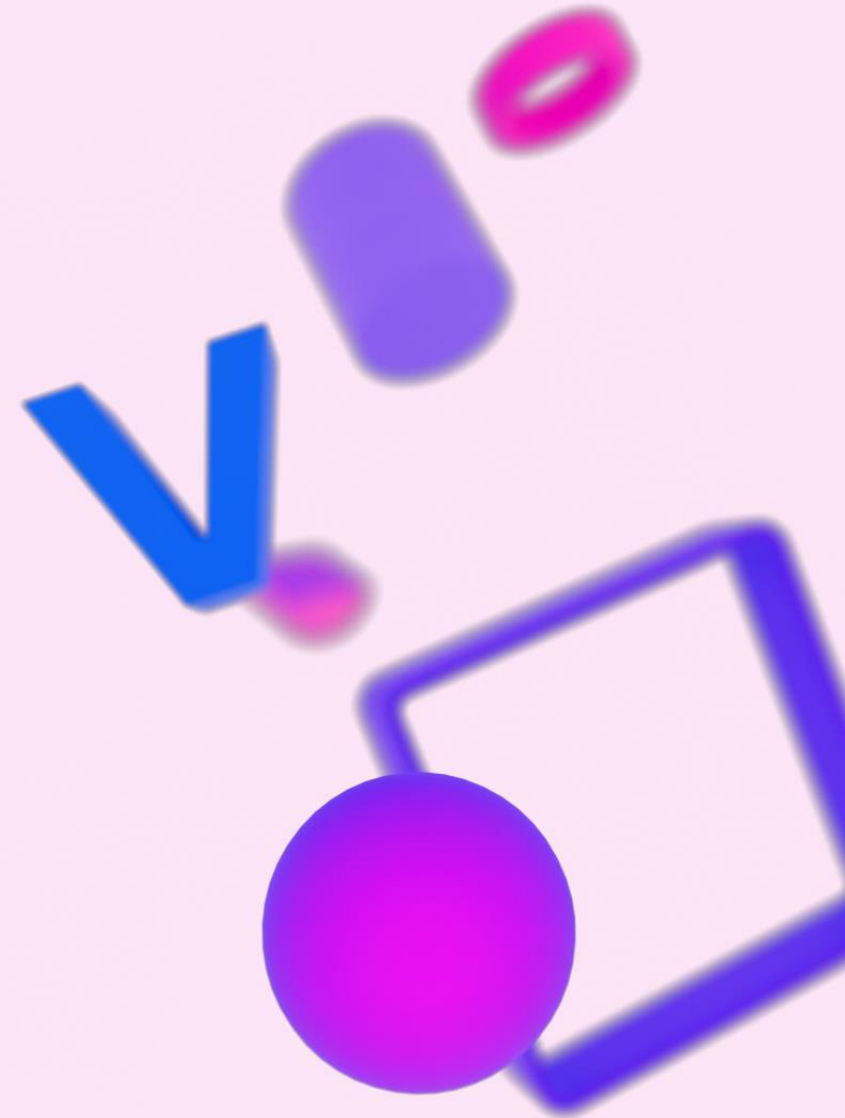
Reference: <https://www.youtube.com/watch?v=sclAwLrruMY>

.NET Containers (Chiseled + .NET AOT)



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

Demo



Reference

- [原生 AOT 的 ASP.NET Core 支援](#)
- [Tiny, fast ASP.NET Core APIs with native AOT | .NET Conf 2023](#)
- [.NET Containers advancements in .NET 8 | .NET Conf 2023](#)
- [Updates to Docker images in .NET 8](#)
- [How to make libraries compatible with native AOT](#)

Contact Info

Kevin Yang

FB Page: <https://www.facebook.com/CKNotepad>

Blog: <https://blog.kevinyang.net/>

