WEBASSEMBLY

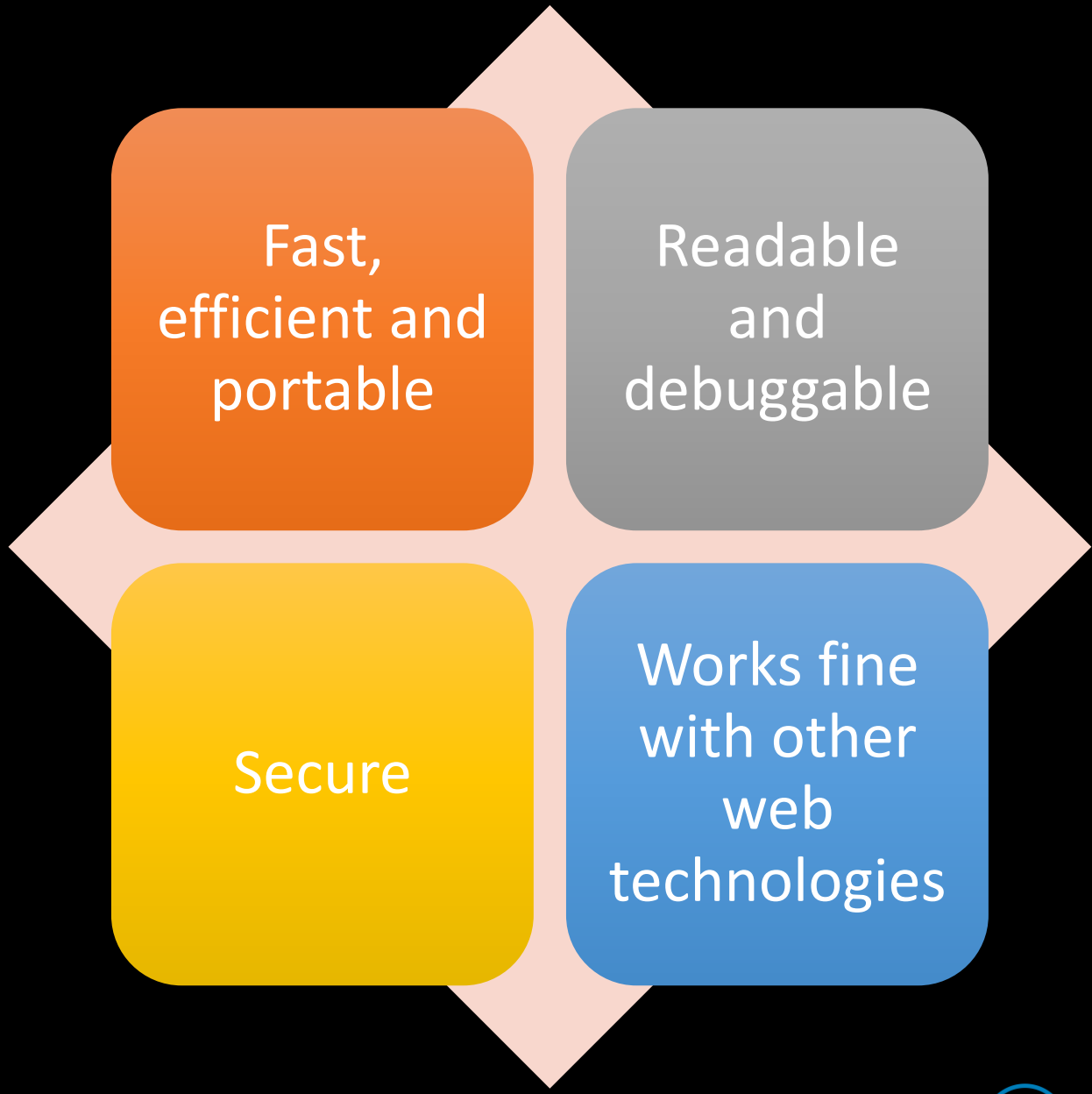
Agenda

- What?
- Why?
- What about Security?
- How?
- When to Use?
- Demo

What?

WebAssembly is a **new type of code that can be run in modern web browsers** — it is a low-level assembly-like language with a compact binary format that runs with near-native performance and provides languages such as C/C++ with a compilation target so that they can run on the web.

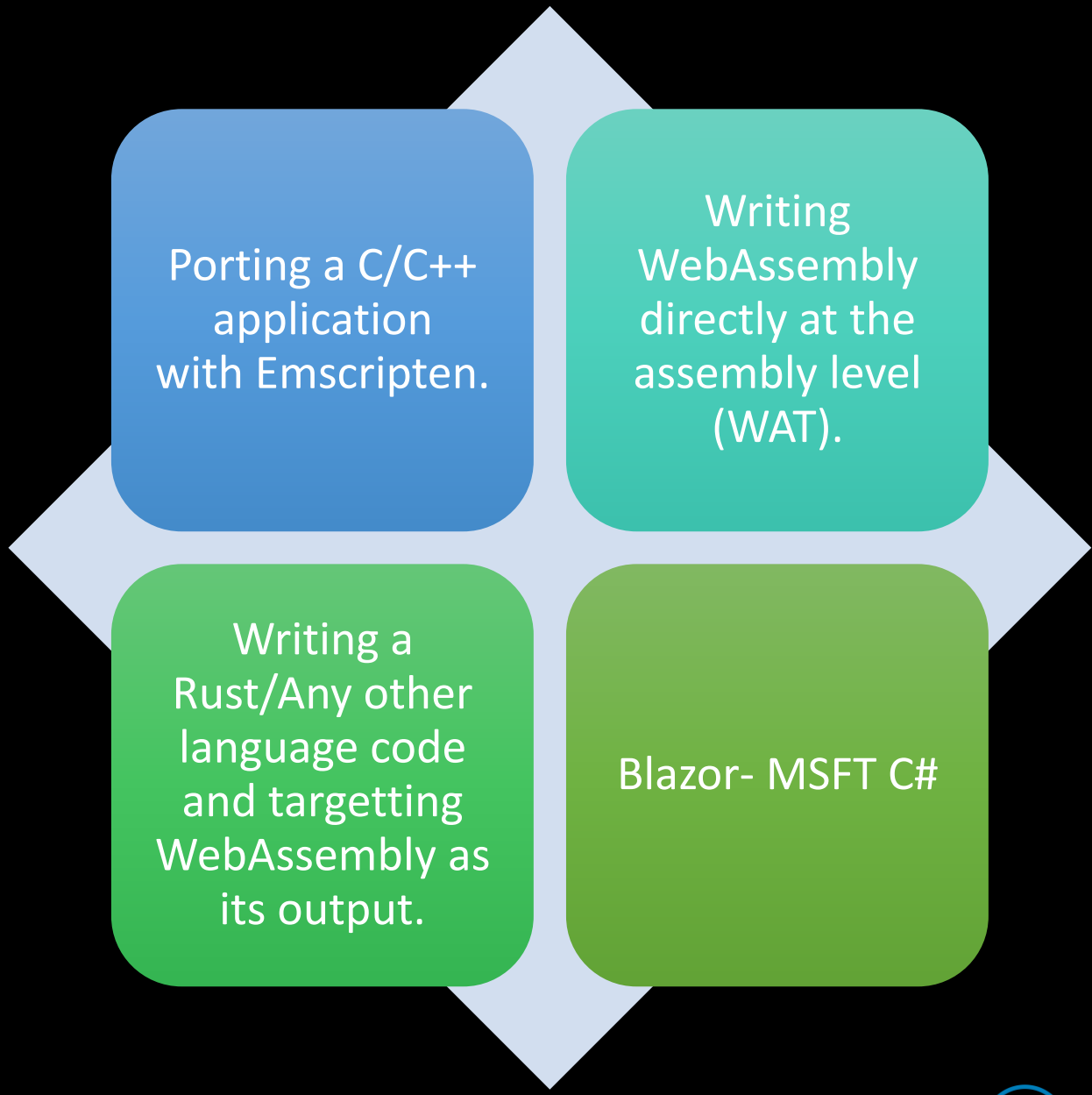
Why?


Fast, efficient and portable
Readable and debuggable
Secure
Works fine with other web technologies

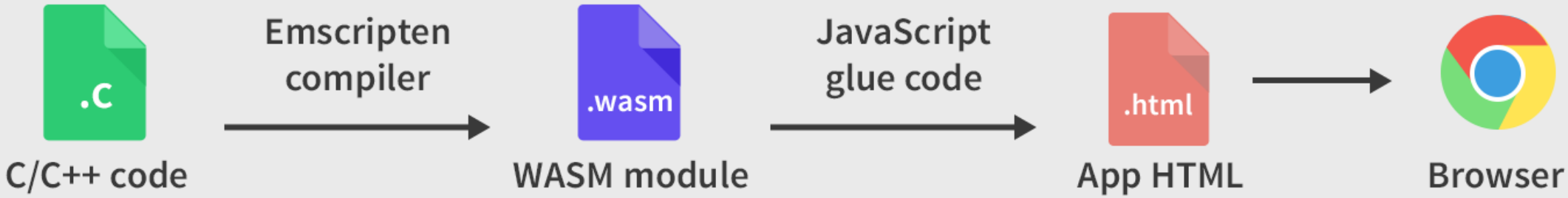
What about Security?

WebAssembly is specified to be run in a safe, sandboxed execution environment. Like other web code, it will enforce the browser's same-origin and permissions policies.

How?


Porting a C/C++ application with Emscripten.
Writing WebAssembly directly at the assembly level (WAT).
Writing a Rust/Any other language code and targetting WebAssembly as its output.
Blazor- MSFT C#

Porting a C/C++ application with Emscripten



Inside the browser

Use Cases

Better execution for languages & toolkits that are currently cross-compiled to the Web (C/C++, GWT)	Image / video editing	Games	Peer-to-peer applications (collaborative editing)
Music applications (streaming, caching).	Image recognition.	Live video augmentation (e.g. putting hats on people's heads).	VR and augmented reality (very low latency).
CAD applications.	Scientific visualization and simulation.	Interactive educational software, and news articles.	Platform simulation / emulation (ARC, DOSBox, QEMU, MAME).
Language interpreters and virtual machines.	POSIX user-space environment, allowing porting of existing POSIX applications.	Developer tooling (editors, compilers, debuggers).	Remote desktop.
VPN.	Encryption.	Local web server.	Common NPAPI users, within the web's security model and APIs.
	Fat client for enterprise applications (e.g. databases).		

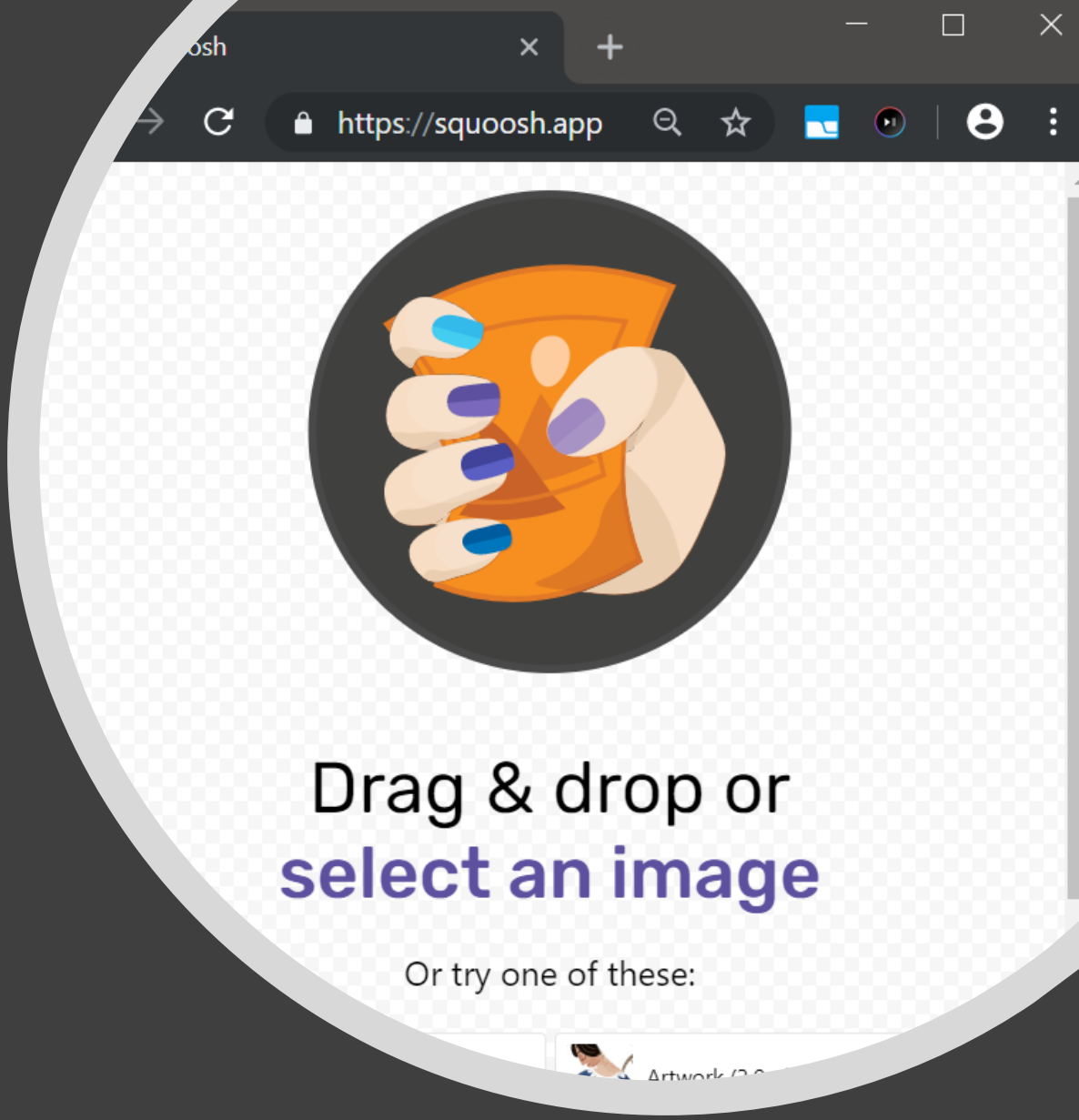
Outside the browser

Use Cases

- Game distribution service (portable and secure).
- Server-side compute of untrusted code.
- Server-side application.
- Hybrid native apps on mobile devices.
- Symmetric computations across multiple nodes

When to use?

- AutoCAD
- Unity game engine
- QT



Demo

Porting from C/C++

Two of the many options for creating WASM code are an online wasm assembler or Emscripten. There are a number of online WASM assembler choices, such as:

- WasmFiddle 
- WasmFiddle++
- WasmExplorer

References

https://developer.mozilla.org/en-US/docs/WebAssembly

WebAssembly (Google I/O ’19)
https://youtu.be/njt-Qzw0mVY

https://webassembly.org/

Thank you