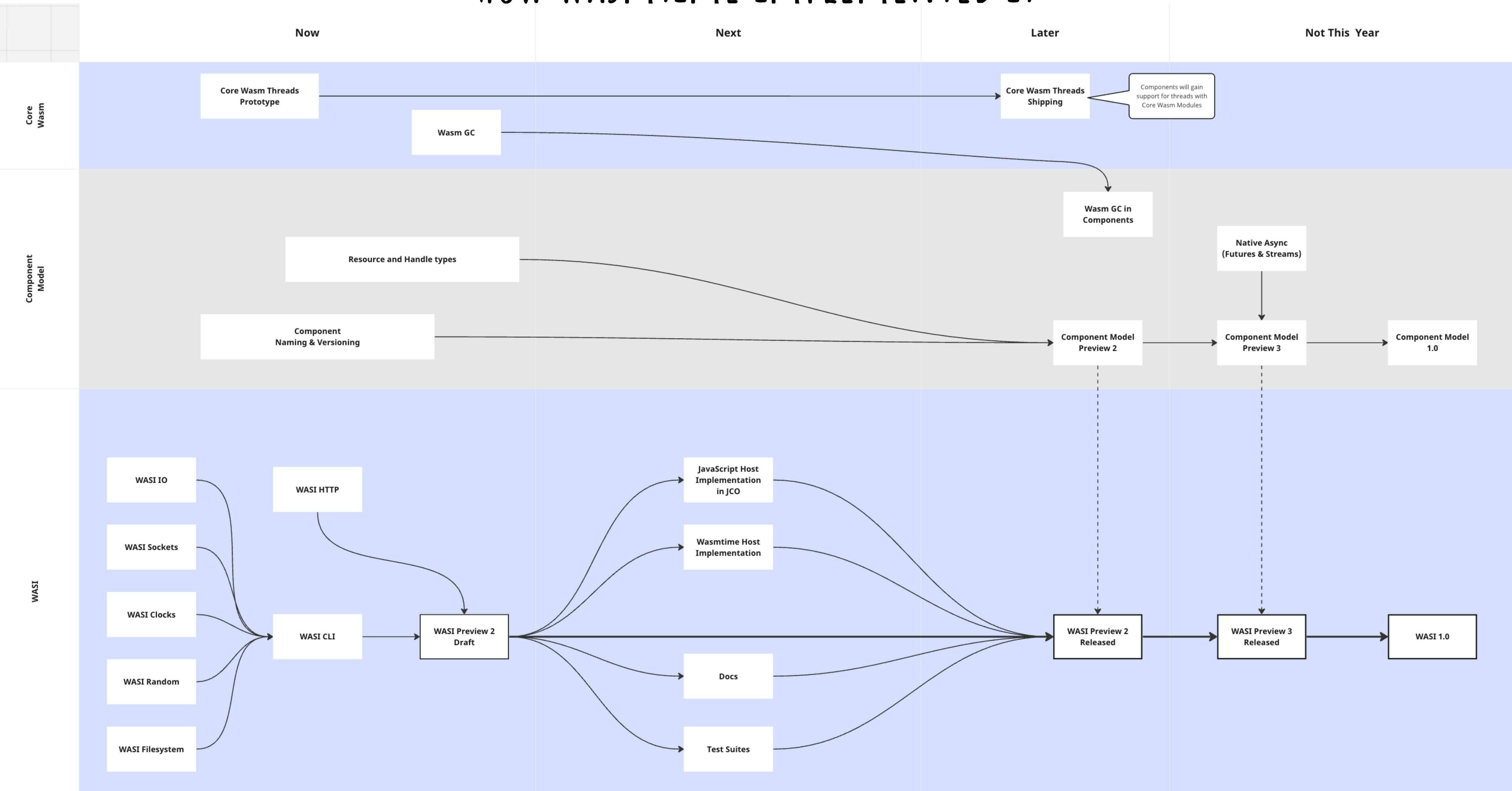
WASI Preview 2

- We talked about WASI and Preview 1
- Preview 2!
- WASI allows nested virtualization
- Towards a WASI 1.0

How Wasmtime implemented it



Now

Next

Later

Not This Year

Core Wasm

Component Model

WASI

Core Wasm Threads Prototype

Wasm GC

Core Wasm Threads Shipping

Components will gain support for threads with Core Wasm Modules

Wasm GC in Components

Resource and Handle types

Component Naming & Versioning

Native Async (Futures & Streams)

Component Model Preview 2

Component Model Preview 3

Component Model 1.0

WASI IO

WASI Sockets

WASI Clocks

WASI Random

WASI Filesystem

WASI HTTP

WASI CLI

WASI Preview 2 Draft

JavaScript Host Implementation in JCO

Wasmtime Host Implementation

Docs

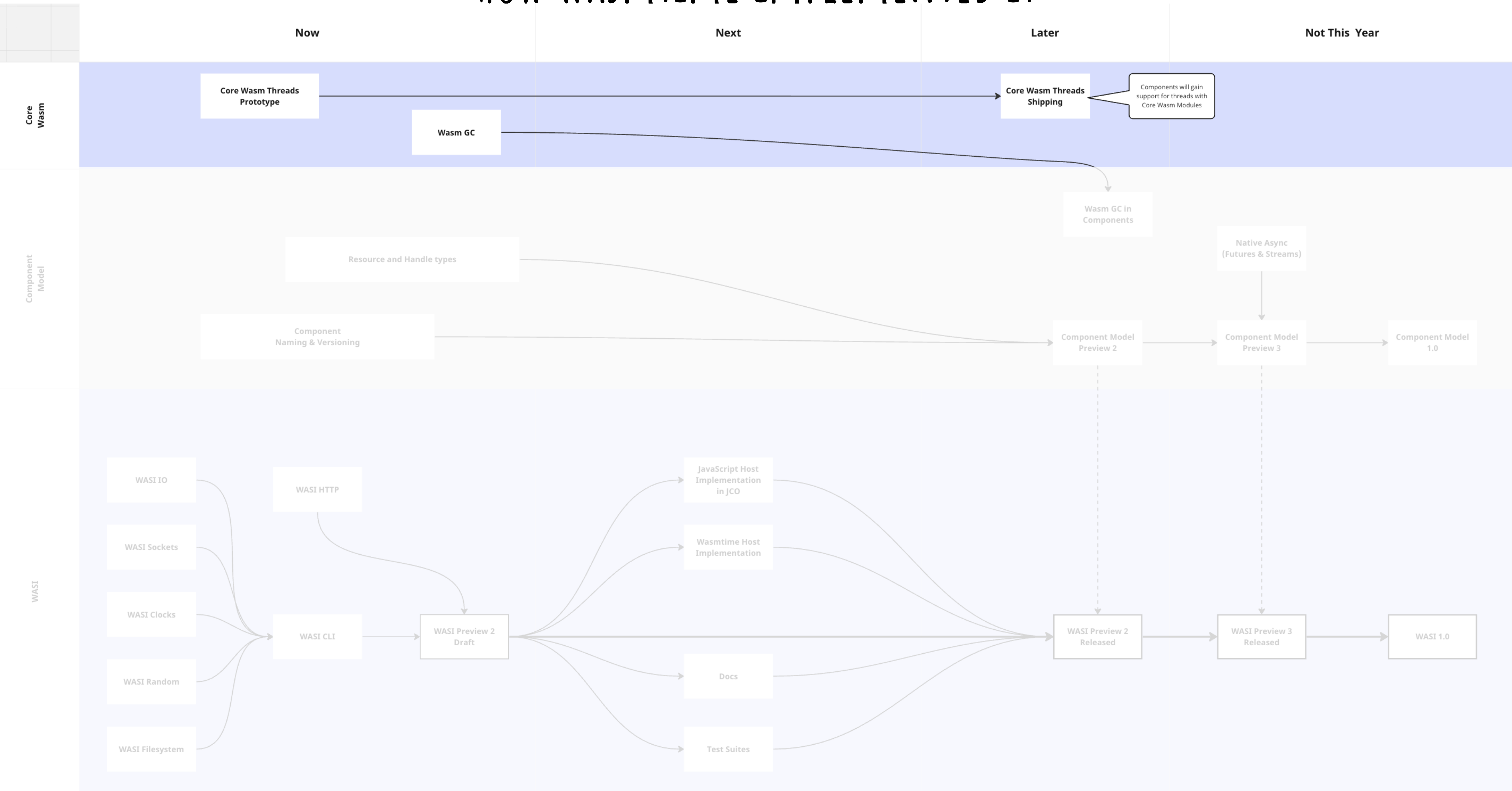
Test Suites

WASI Preview 2 Released

WASI Preview 3 Released

WASI 1.0

How Wasmtime implemented it



Now

Next

Later

Not This Year

Core Wasm

Core Wasm Threads Prototype

Wasm GC

Core Wasm Threads Shipping

Components will gain support for threads with Core Wasm Modules

Component Model

Resource and Handle types

Component Naming & Versioning

Wasm GC in Components

Native Async (Futures & Streams)

Component Model Preview 2

Component Model Preview 3

Component Model 1.0

WASI

WASI IO

WASI Sockets

WASI Clocks

WASI Random

WASI Filesystem

WASI HTTP

WASI CLI

WASI Preview 2 Draft

JavaScript Host Implementation in JCO

Wasmtime Host Implementation

Docs

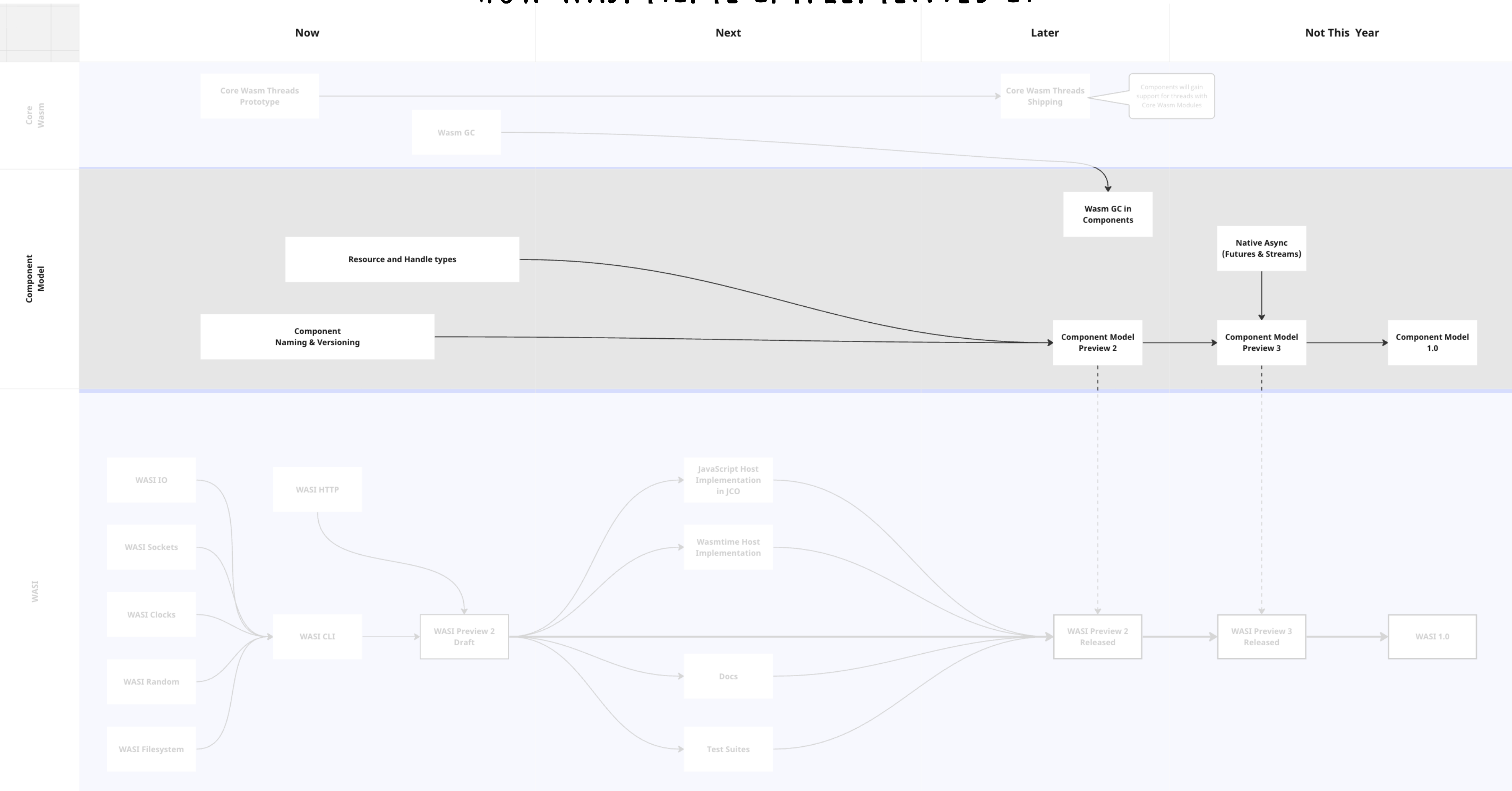
Test Suites

WASI Preview 2 Released

WASI Preview 3 Released

WASI 1.0

How Wasmtime implemented it



Now

Next

Later

Not This Year

Core Wasm

Core Wasm Threads Prototype

Wasm GC

Core Wasm Threads Shipping

Components will gain support for threads with Core Wasm Modules

Component Model

Resource and Handle types

Component Naming & Versioning

Wasm GC in Components

Native Async (Futures & Streams)

Component Model Preview 2

Component Model Preview 3

Component Model 1.0

WASI

WASI IO

WASI Sockets

WASI Clocks

WASI Random

WASI Filesystem

WASI HTTP

WASI CLI

WASI Preview 2 Draft

JavaScript Host Implementation in JCO

Wasmtime Host Implementation

Docs

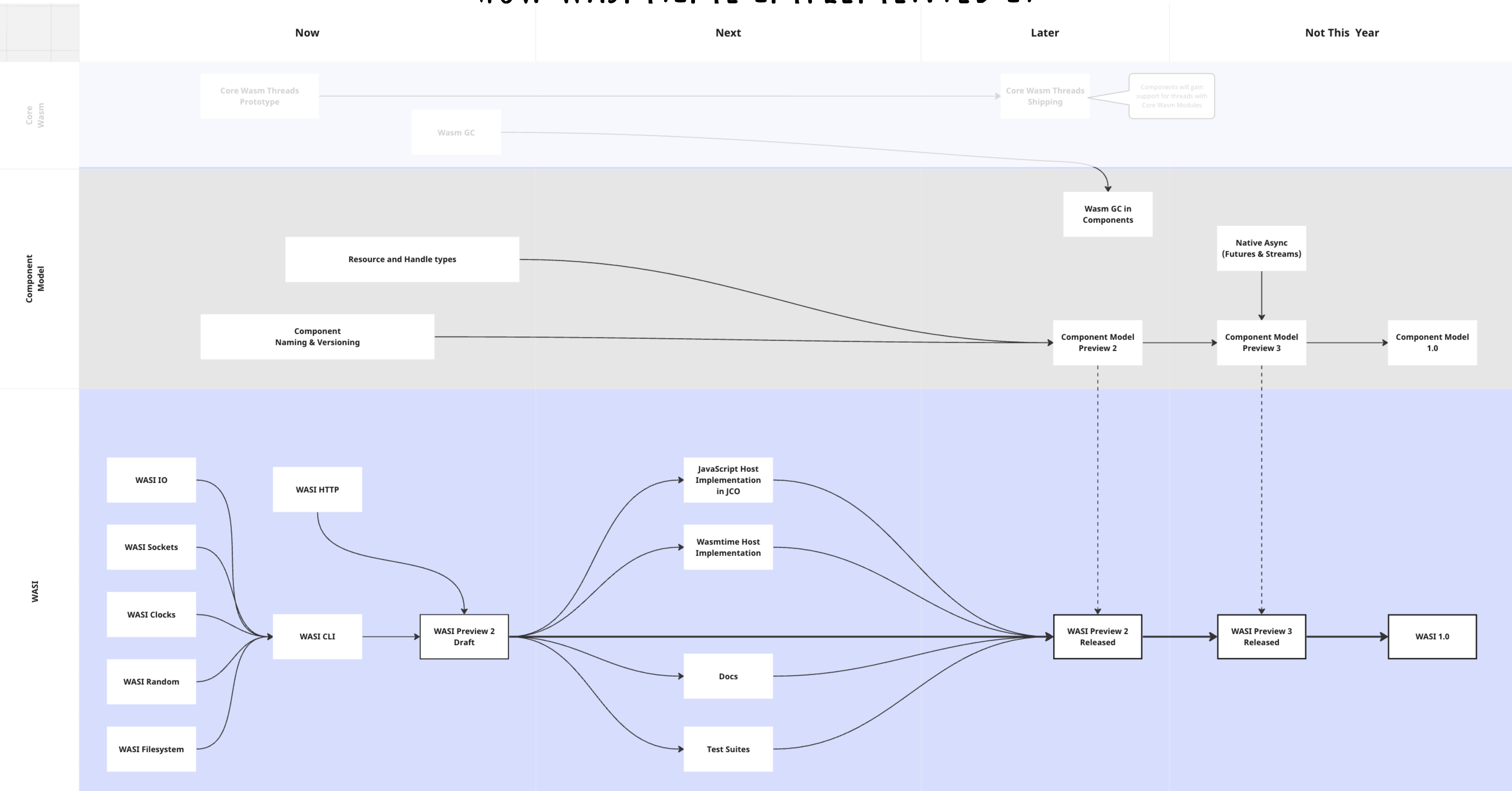
Test Suites

WASI Preview 2 Released

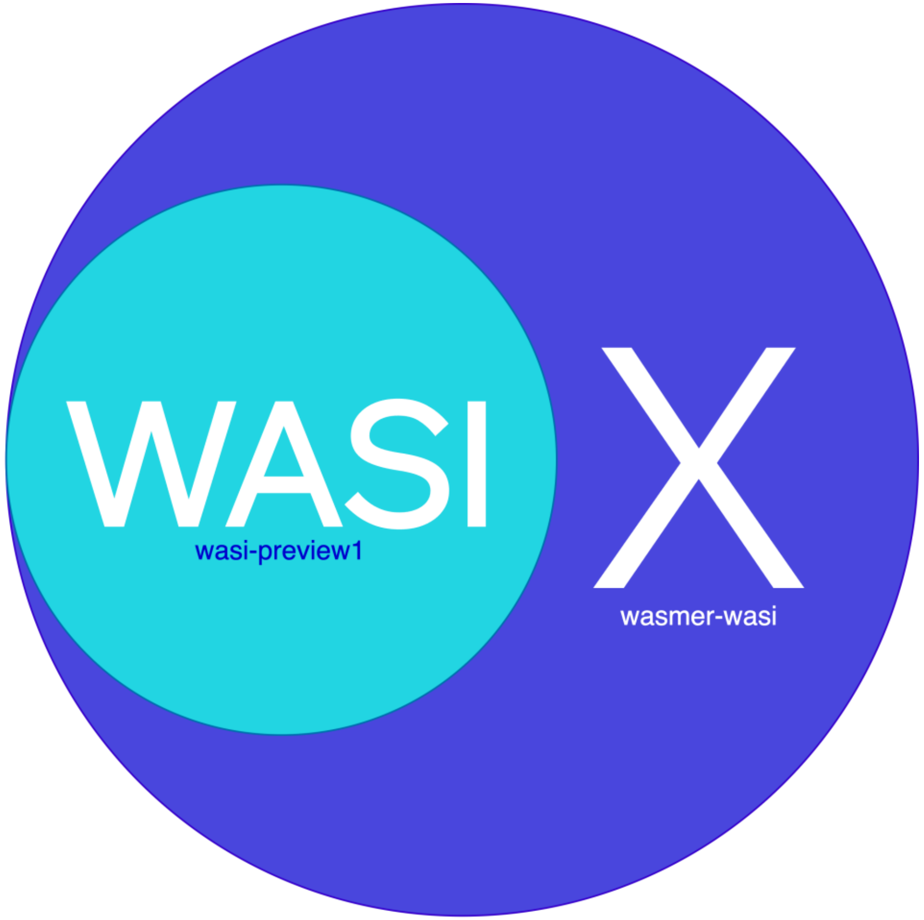
WASI Preview 3 Released

WASI 1.0

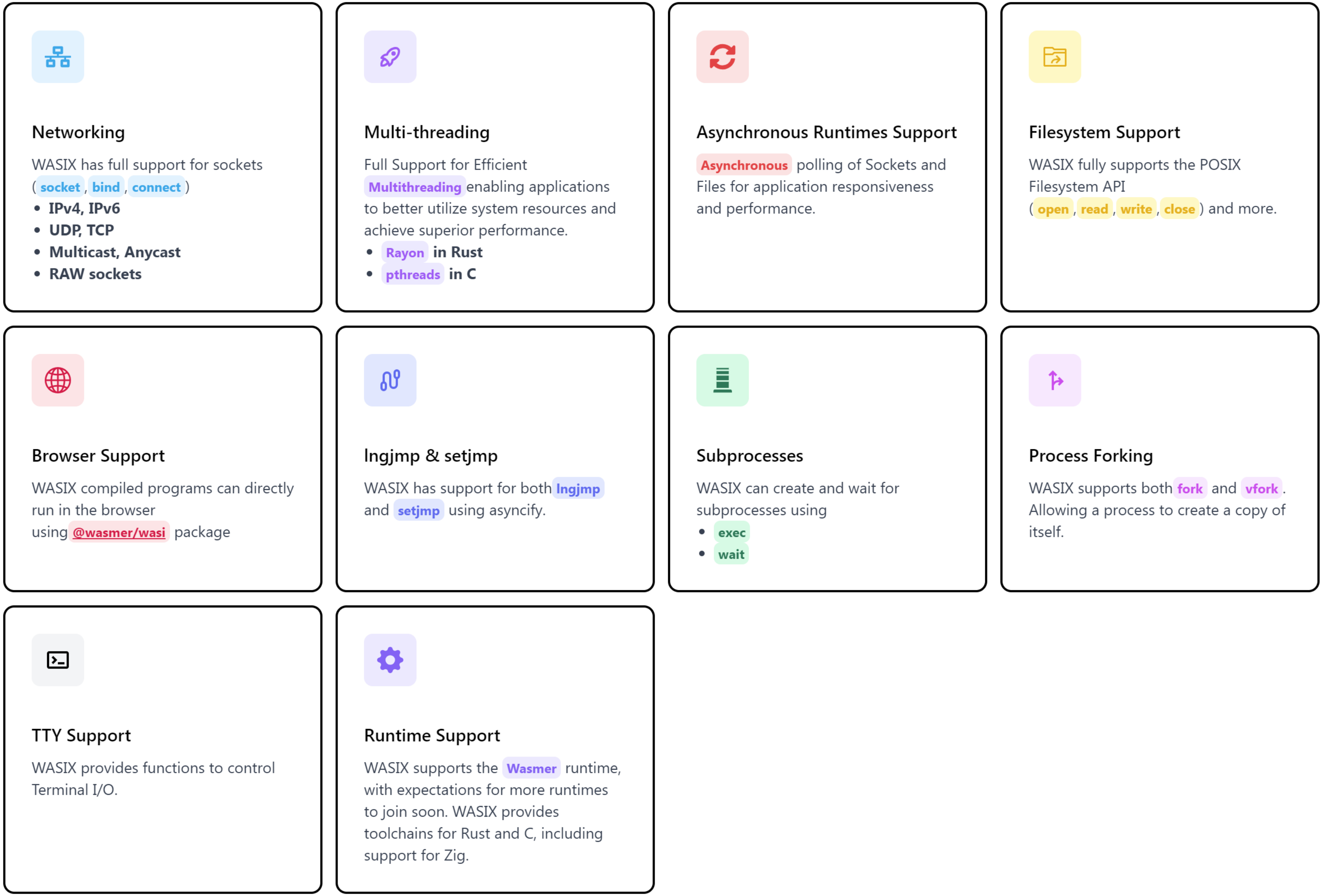
How Wasmtime implemented it


Now
Next
Later
Not This Year
Core Wasm
Core Wasm Threads Prototype
Wasm GC
Core Wasm Threads Shipping
Components will gain support for threads with Core Wasm Modules
Component Model
Wasm GC in Components
Native Async (Futures & Streams)
Resource and Handle types
Component Naming & Versioning
Component Model Preview 2
Component Model Preview 3
Component Model 1.0
WASI
WASI IO
WASI Sockets
WASI Clocks
WASI Random
WASI Filesystem
WASI HTTP
WASI CLI
WASI Preview 2 Draft
JavaScript Host Implementation in JCO
Wasmtime Host Implementation
Docs
Test Suites
WASI Preview 2 Released
WASI Preview 3 Released
WASI 1.0

WASIX



Source: WASIX



Networking

WASIX has full support for sockets (socket , bind , connect)

- **IPv4, IPv6**
- **UDP, TCP**
- **Multicast, Anycast**
- **RAW sockets**

Multi-threading

Full Support for Efficient Multithreading enabling applications to better utilize system resources and achieve superior performance.

- Rayon **in Rust**
- pthreads **in C**

Asynchronous Runtimes Support

Asynchronous polling of Sockets and Files for application responsiveness and performance.

Filesystem Support

WASIX fully supports the POSIX Filesystem API (open , read , write , close) and more.

Browser Support

WASIX compiled programs can directly run in the browser using @wasmer/wasi package

lngjmp & setjmp

WASIX has support for both lngjmp and setjmp using asyncify.

Subprocesses

WASIX can create and wait for subprocesses using

- exec
- wait

Process Forking

WASIX supports both fork and vfork . Allowing a process to create a copy of itself.

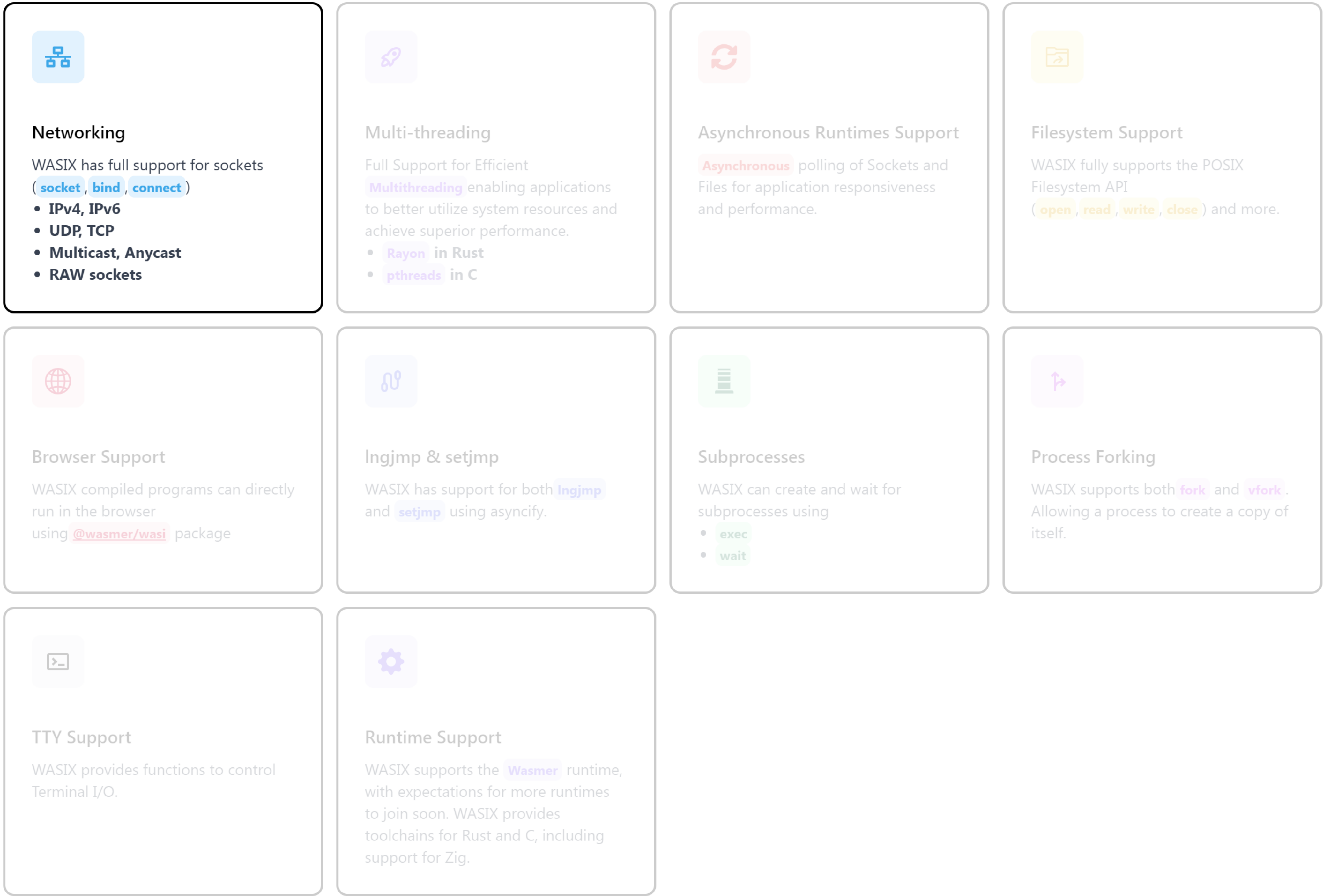
TTY Support

WASIX provides functions to control Terminal I/O.

Runtime Support

WASIX supports the Wasmer runtime, with expectations for more runtimes to join soon. WASIX provides toolchains for Rust and C, including support for Zig.

Source: WASIX



Networking

WASIX has full support for sockets (socket , bind , connect)

- IPv4, IPv6
- UDP, TCP
- Multicast, Anycast
- RAW sockets

Multi-threading

Full Support for Efficient Multithreading enabling applications to better utilize system resources and achieve superior performance.

- Rayon in Rust
- pthreads in C

Asynchronous Runtimes Support

Asynchronous polling of Sockets and Files for application responsiveness and performance.

Filesystem Support

WASIX fully supports the POSIX Filesystem API (open , read , write , close) and more.

Browser Support

WASIX compiled programs can directly run in the browser using @wasmer/wasi package

lngjmp & setjmp

WASIX has support for both lngjmp and setjmp using asyncify.

Subprocesses

WASIX can create and wait for subprocesses using

- exec
- wait

Process Forking

WASIX supports both fork and vfork . Allowing a process to create a copy of itself.

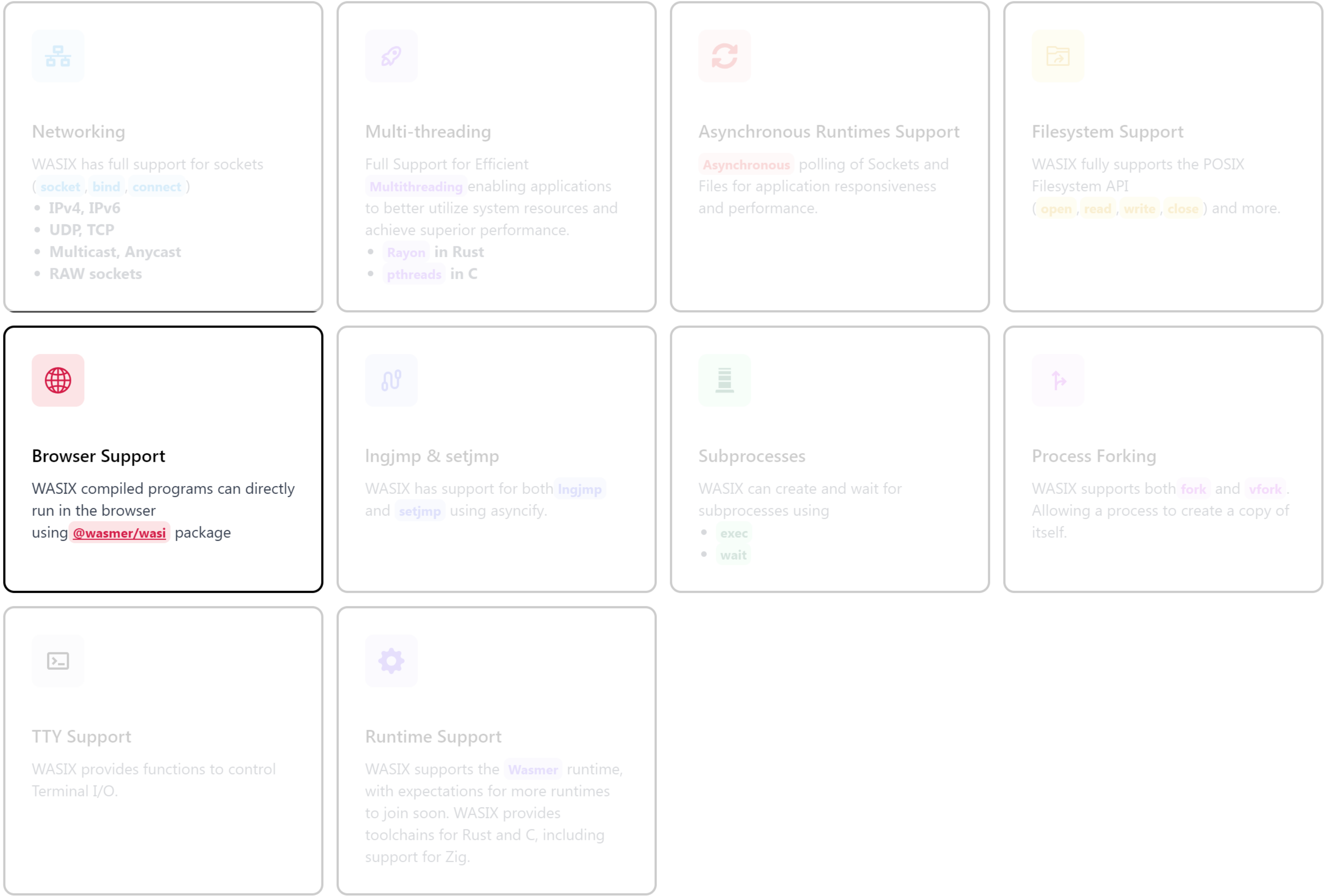
TTY Support

WASIX provides functions to control Terminal I/O.

Runtime Support

WASIX supports the Wasmer runtime, with expectations for more runtimes to join soon. WASIX provides toolchains for Rust and C, including support for Zig.

Source: WASIX



Networking

WASIX has full support for sockets (socket, bind, connect)

- IPv4, IPv6
- UDP, TCP
- Multicast, Anycast
- RAW sockets

Multi-threading

Full Support for Efficient Multithreading enabling applications to better utilize system resources and achieve superior performance.

- Rayon in Rust
- pthreads in C

Asynchronous Runtimes Support

Asynchronous polling of Sockets and Files for application responsiveness and performance.

Filesystem Support

WASIX fully supports the POSIX Filesystem API (open, read, write, close) and more.

Browser Support

WASIX compiled programs can directly run in the browser using @wasmer/wasi package

lngjmp & setjmp

WASIX has support for both lngjmp and setjmp using asyncify.

Subprocesses

WASIX can create and wait for subprocesses using

- exec
- wait

Process Forking

WASIX supports both fork and vfork. Allowing a process to create a copy of itself.

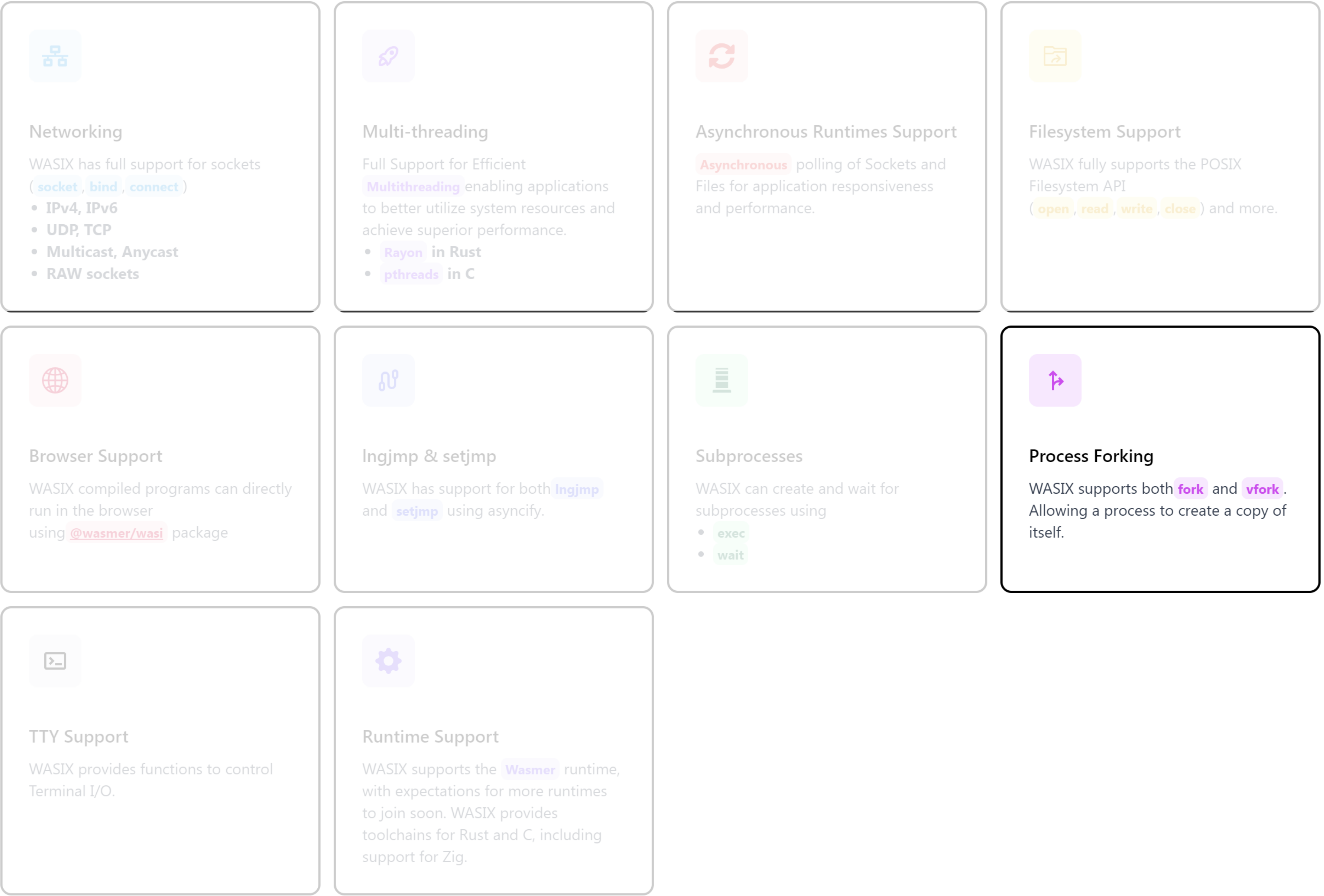
TTY Support

WASIX provides functions to control Terminal I/O.

Runtime Support

WASIX supports the Wasmer runtime, with expectations for more runtimes to join soon. WASIX provides toolchains for Rust and C, including support for Zig.

Source: WASIX



Networking

WASIX has full support for sockets (socket, bind, connect)

- IPv4, IPv6
- UDP, TCP
- Multicast, Anycast
- RAW sockets

Multi-threading

Full Support for Efficient Multithreading enabling applications to better utilize system resources and achieve superior performance.

- Rayon in Rust
- pthreads in C

Asynchronous Runtimes Support

Asynchronous polling of Sockets and Files for application responsiveness and performance.

Filesystem Support

WASIX fully supports the POSIX Filesystem API (open, read, write, close) and more.

Browser Support

WASIX compiled programs can directly run in the browser using @wasmer/wasi package

lngjmp & setjmp

WASIX has support for both lngjmp and setjmp using asyncify.

Subprocesses

WASIX can create and wait for subprocesses using

- exec
- wait

Process Forking

WASIX supports both fork and vfork. Allowing a process to create a copy of itself.

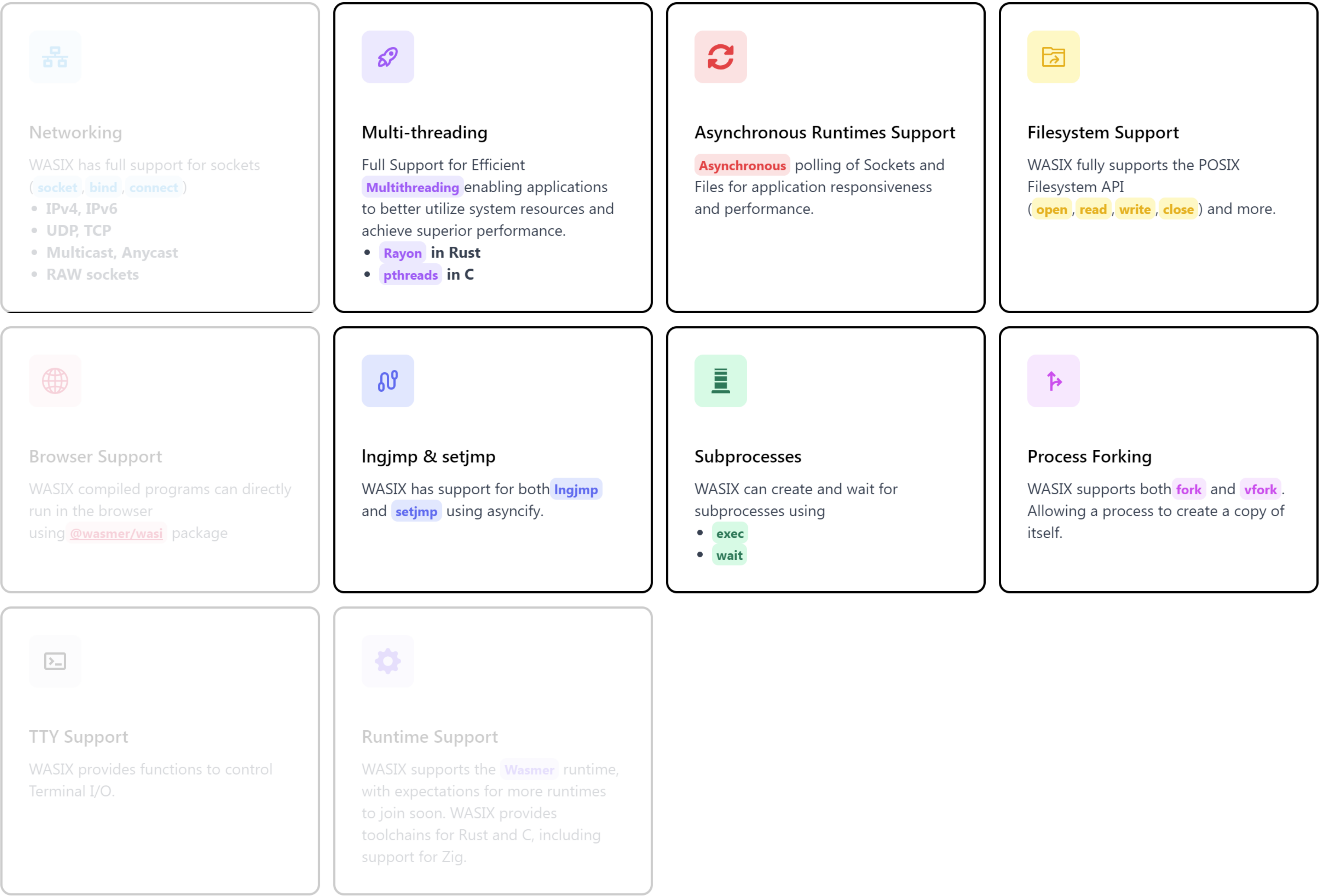
TTY Support

WASIX provides functions to control Terminal I/O.

Runtime Support

WASIX supports the Wasmer runtime, with expectations for more runtimes to join soon. WASIX provides toolchains for Rust and C, including support for Zig.

Source: WASIX



Networking

WASIX has full support for sockets (socket, bind, connect)

- IPv4, IPv6
- UDP, TCP
- Multicast, Anycast
- RAW sockets

Multi-threading

Full Support for Efficient Multithreading enabling applications to better utilize system resources and achieve superior performance.

- Rayon in Rust
- pthreads in C

Asynchronous Runtimes Support

Asynchronous polling of Sockets and Files for application responsiveness and performance.

Filesystem Support

WASIX fully supports the POSIX Filesystem API (open, read, write, close) and more.

Browser Support

WASIX compiled programs can directly run in the browser using @wasmer/wasi package

lngjmp & setjmp

WASIX has support for both lngjmp and setjmp using asyncify.

Subprocesses

WASIX can create and wait for subprocesses using

- exec
- wait

Process Forking

WASIX supports both fork and vfork. Allowing a process to create a copy of itself.

TTY Support

WASIX provides functions to control Terminal I/O.

Runtime Support

WASIX supports the Wasmer runtime, with expectations for more runtimes to join soon. WASIX provides toolchains for Rust and C, including support for Zig.

Source: WASIX