You are using Copilot Chat with **GPT-5** with standard access. Quality and performance may vary depending on service availability.[Learn more.](https://go.microsoft.com/fwlink/?linkid=2329839)

**Today**

**You said:**

Wanted to know this: in msys64 Should I use rustup update before cargo install-update -a or they both work independently, and I can run them in parallel?

**Copilot said:**

Copilot

Reasoned for 8s

Clarifying rustup and cargo commands

Explaining that rustup update and cargo install-update -a are independent commands. rustup update modifies the active toolchain, while cargo install-update -a rebuilds binaries with the chosen toolchain.

Explaining concurrent execution risks

Describing potential issues when running rustup update and cargo install-update -a concurrently, including race conditions, build failures, and mixed versions. Suggests a safe execution sequence: rustup update, check toolchain, then cargo install-update -a.

Summarizing recommended steps for rustup and cargo

Short answer: **do them sequentially—rustup update first, then cargo install-update -a**. They’re conceptually independent, but not completely isolated in practice, so **don’t run them in parallel**.

**Why this order?**

* **rustup update** updates your Rust toolchains (e.g., stable, nightly), along with rustc and cargo that come with the toolchain.
* **cargo install-update -a** (from the cargo-update plugin) recompiles and reinstalls the binaries you previously installed via cargo install, **using the currently active toolchain**.

If you run them at the same time, two things can go wrong:

1. **Race/locking issues:** While rustup is changing toolchain directories, cargo may try to build with tools that are being updated → intermittent failures.
2. **Inconsistent builds:** Some crates may build against the **old** toolchain while others build against the **new** one, depending on timing.

Running rustup update first ensures all your binaries get rebuilt with the latest compiler and standard library.