

PSR Naming Conventions

1. Interfaces MUST be suffixed by `Interface`: e.g. `Psr\Foo\BarInterface`.
2. Abstract classes MUST be prefixed by `Abstract`: e.g. `Psr\Foo\AbstractBar`.
3. Traits MUST be suffixed by `Trait`: e.g. `Psr\Foo\BarTrait`.
4. PSR-1, 2 and 4 MUST be followed.
5. The vendor namespace MUST be `Psr`.
6. There MUST be a package/second-level namespace in relation with the PSR that covers the code.
7. Composer package MUST be named `psr/<package>` e.g. `psr/log`. If they require an implementation as a virtual package it MUST be named `psr/<package>-implementation` and be required with a specific version like `1.0.0`. Implementors of that PSR can then provide `"psr/<package>-implementation": "1.0.0"` in their package to satisfy that requirement. Specification changes via further PSRs should only lead to a new tag of the `psr/<package>` package, and an equal version bump of the implementation being required.
8. Special lightweight utility packages MAY be produced alongside PSRs and interfaces and be maintained and updated after the PSR has been accepted. These MUST be under the vendor namespace `Fig`.