



Wrapping with Reflection

- **Thread-safe wrappers**
 - `synchronized_value`
 - `std::mutex/shared_mutex/spin_lock`
 - `crll::seqlock_object`
- **Value wrappers around shared objects**
 - Automatic reference counting (arc)
 - Copy-on-write objects (cow)
- **Async classes**



- P2996 - Reflection for C++26

Accepted 

- P3294 - Code Injection with Token Sequences

Hopeful for C++26  SOON

- P0707 - Metaclasses

Proposed 

C++29





Wrapping with Reflection

- P2996 - Reflection for C++26

Accepted ✓

- P3294 - Code Injection with Token Sequences

Hopeful for C++29 🙄

- P0707 - Metaclasses

Proposed ⚠



Implicit synchronized_value

```
class person
{
public:
    person() = default;

    std::string get_first_name() const
    {
        return first_name;
    }

    void set_first_name (std::string_view new_first)
    {
        first_name = new_first;
    }

    // Repeat for last_name

private:
    std::string first_name, last_name;
};
```