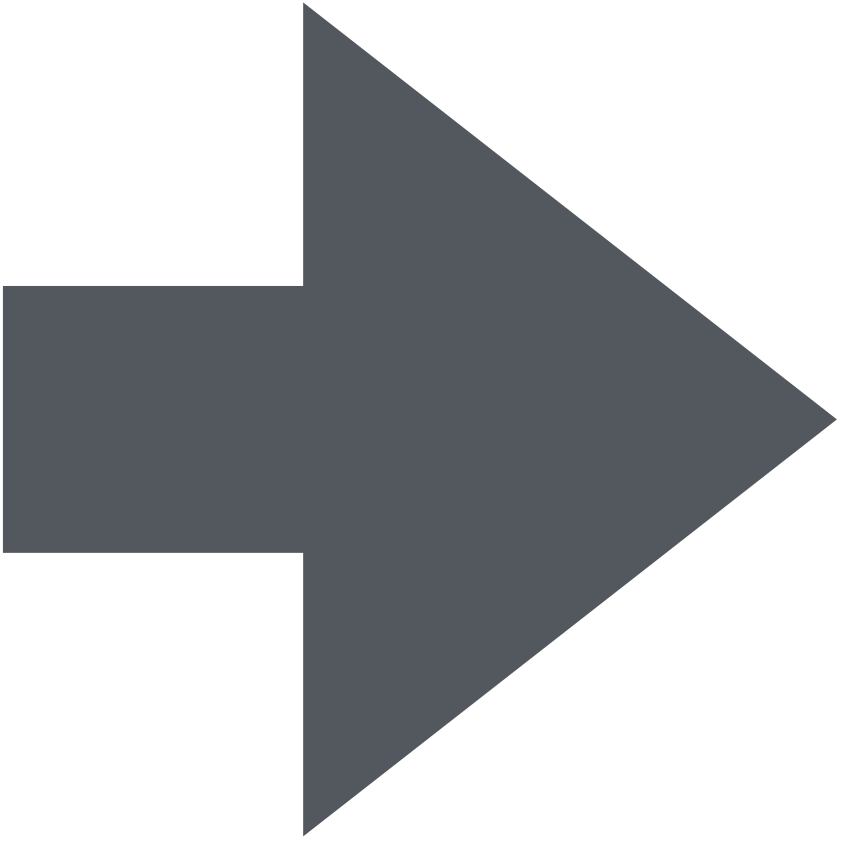
Wrapped `std::shared_ptr`

71

```
class(arc) 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;
};
```

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

    std::string get_first_name() const
    {
        return person_->get_first_name();
    }

    void set_first_name (std::string_view new_first)
    {
        person_->set_first_name (new_first);
    }

    // Repeat for last_name

private:
    class __person;
    std::shared_ptr<__person> person_;
};

template<>
struct is_send<person> : is_sync_v<person::__person> {};
```

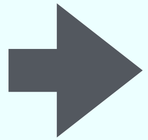


arc metaclass

Wrapped `std::shared_ptr`

arc metaclass

```
class(arc) 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;
};
```



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

    std::string get_first_name() const
    {
        return person_->get_first_name();
    }

    void set_first_name (std::string_view new_first)
    {
        person_->set_first_name (new_first);
    }

    // Repeat for last_name

private:
    class __person;
    std::shared_ptr<__person> person_;
};

template<>
struct is_send<person> : is_sync_v<person::__person> {};
```

Look familiar? Swift `class`es

```
class Person
{
    private var first_name: String = "";
    private var last_name: String = "";

    func get_first_name() -> String
    {
        return first_name
    }

    mutating func set_first_name (new_first: String)
    {
        first_name = new_first;
    }

    // Repeat for last_name
}
```

```
class(arc) person
{
public:
    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;
};
```