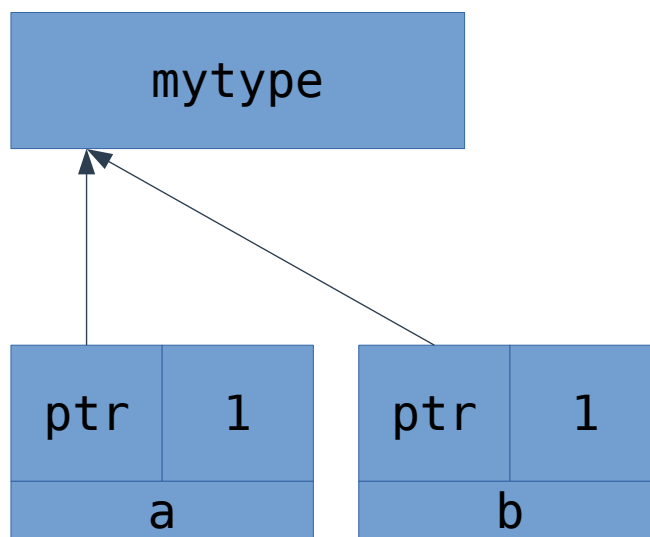


**shared\_ptr**

# Пример

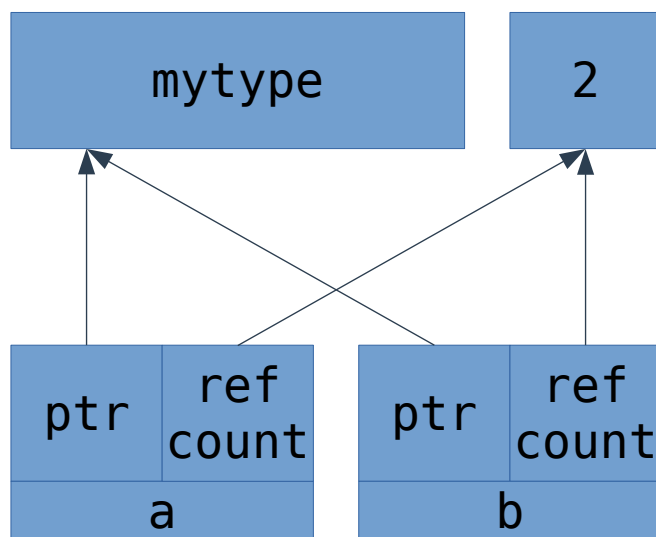
```
void foo()  
{  
    std::shared_ptr<mytype> a(new mytype());  
    std::shared_ptr<mytype> b = a;  
}
```

# Счетчик ссылок



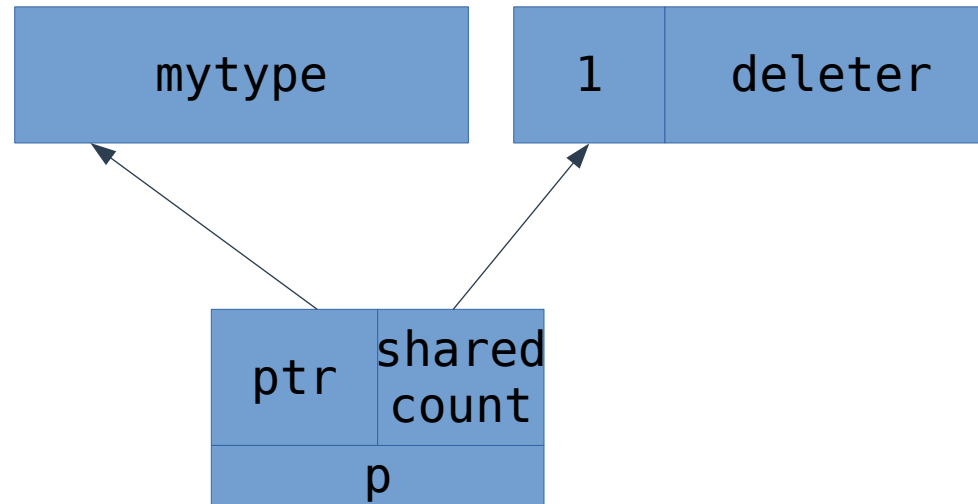
```
std::shared_ptr<mytype> a(new mytype());  
std::shared_ptr<mytype> b = a;
```

# Счетчик ссылок



```
std::shared_ptr<mytype> a(new mytype());  
std::shared_ptr<mytype> b = a;
```

# Custom deleter



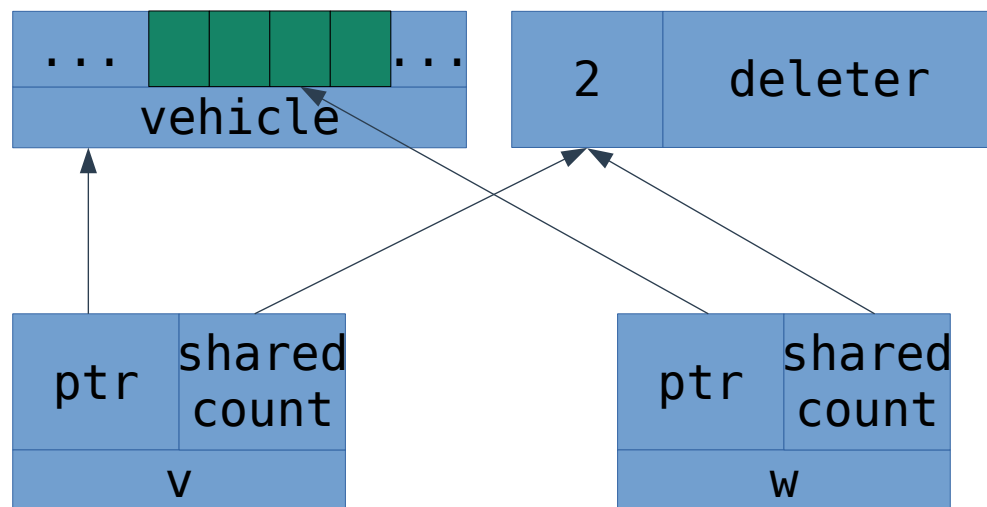
```
struct deleter
{
    void operator()(mytype* obj) const
    {
        delete obj;
    }
};
```

```
std::shared_ptr<mytype> p(new mytype(), deleter());
```

# Aliasing Constructor

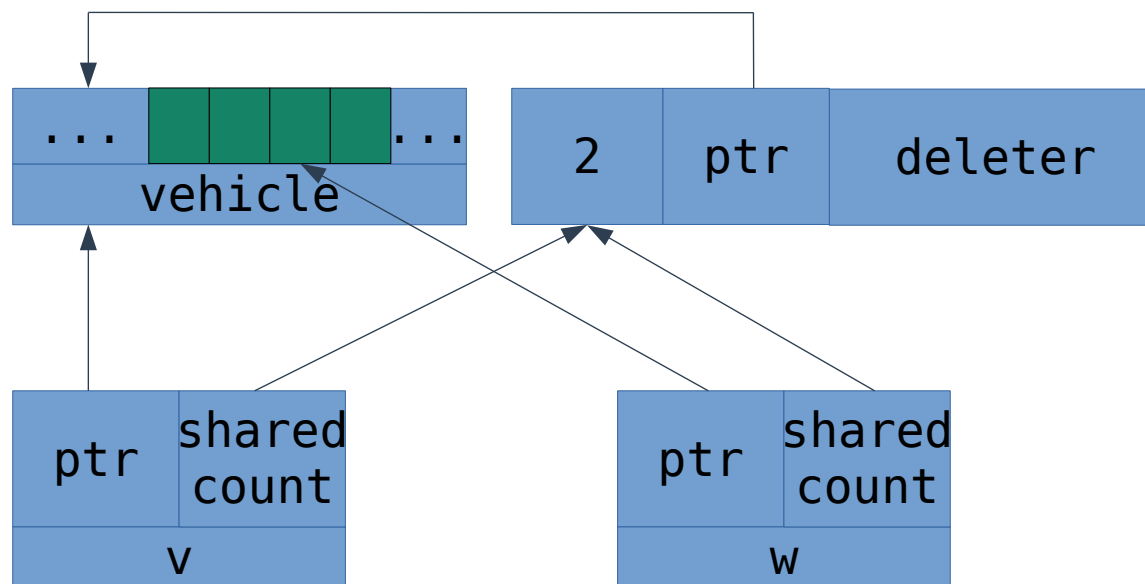
```
struct vehicle  
{  
    std::array<wheel, 4> wheels;  
};  
  
std::shared_ptr<vehicle> v(new vehicle);  
std::shared_ptr<wheel> w(v, &v->wheels[2]);
```

# Aliasing Constructor



```
std::shared_ptr<vehicle> v(new vehicle);  
std::shared_ptr<wheel> w(v, &v->wheels[2]);
```

# Aliasing Constructor



```
std::shared_ptr<vehicle> v(new vehicle);  
std::shared_ptr<wheel> w(v, &v->wheels[2]);
```

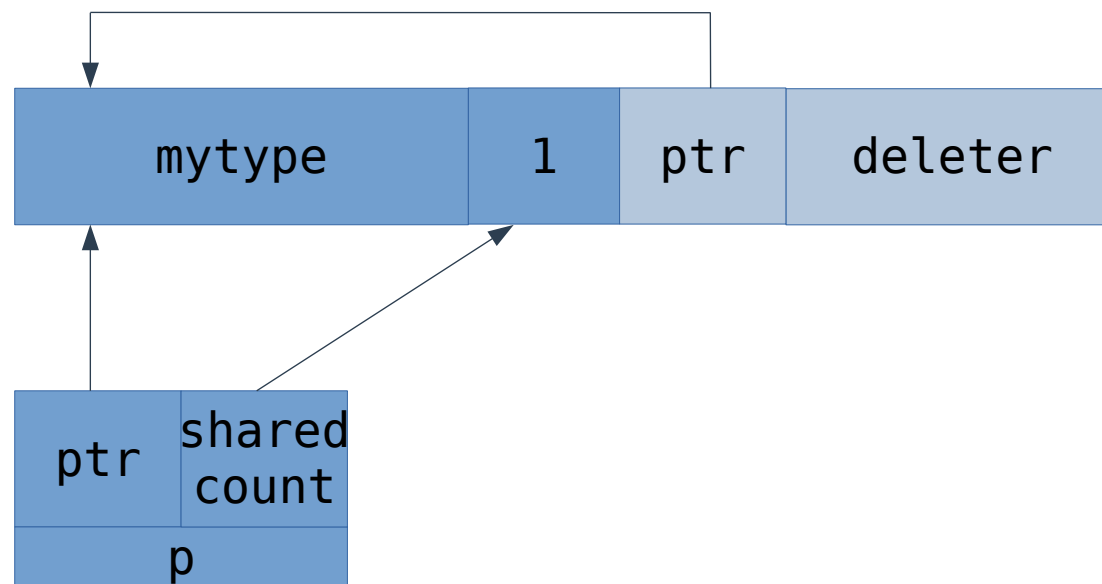


# std::make\_shared

```
struct mytype  
{  
    mytype(int, int, int);  
};
```

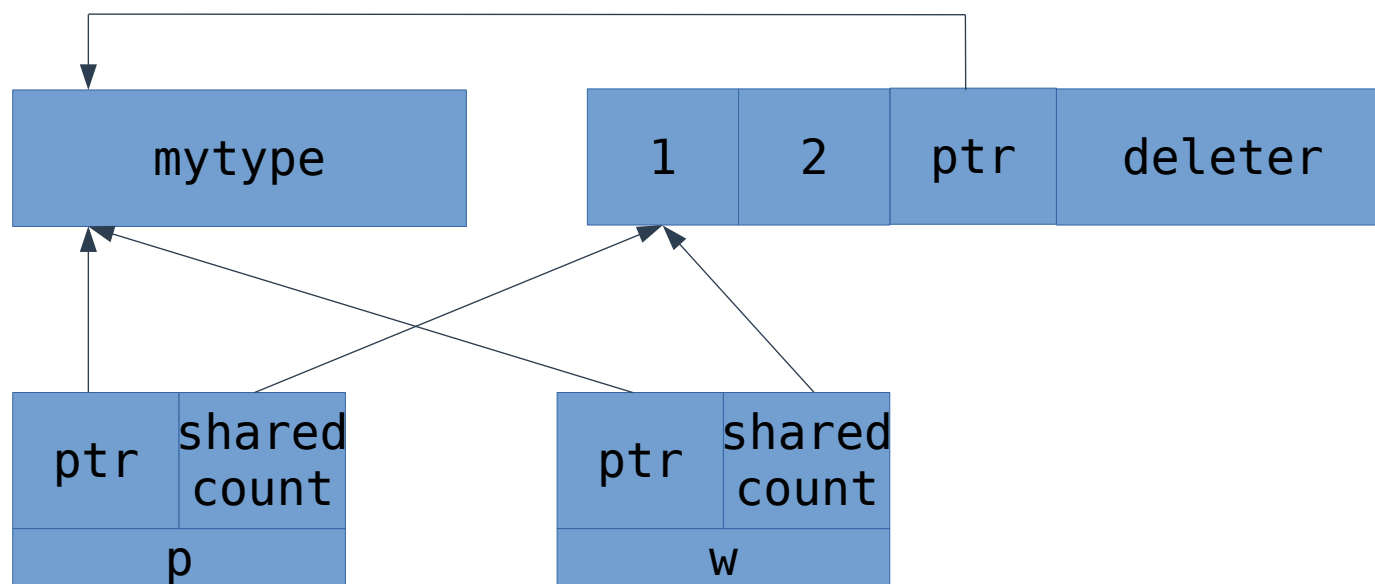
```
std::shared_ptr<mytype> p  
    = std::make_shared<mytype>(1, 2, 3);
```

# std::make\_shared



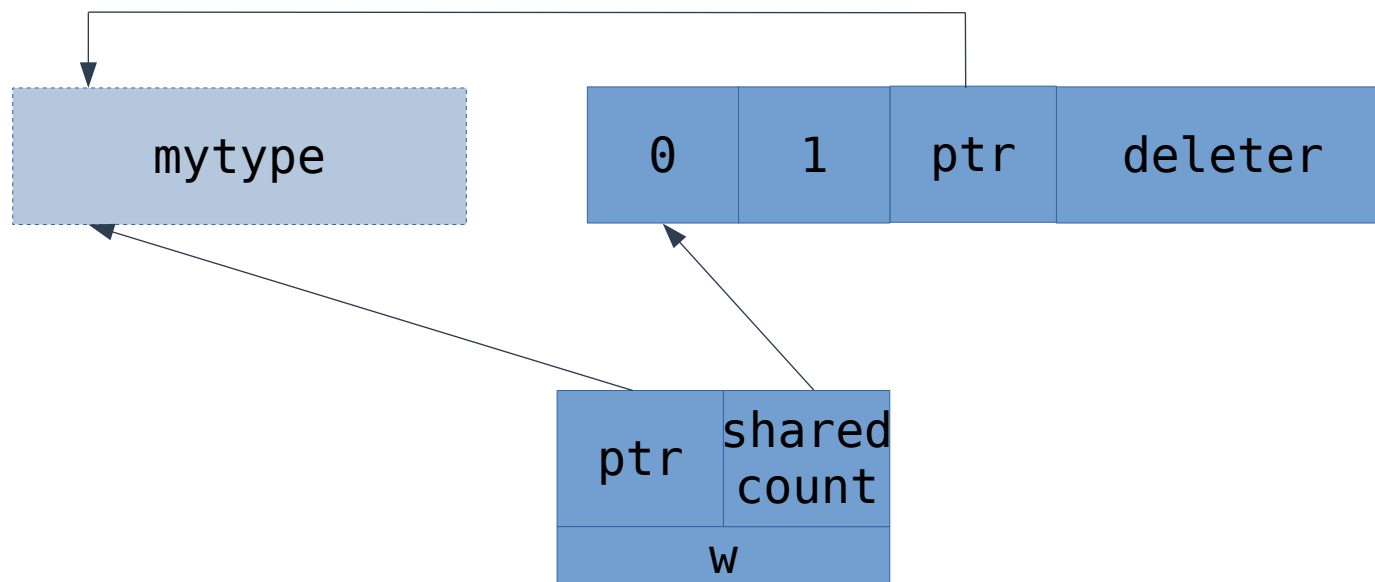
```
std::shared_ptr<mytype> p  
= std::make_shared<mytype>(1, 2, 3);
```

# std::weak\_ptr



```
std::shared_ptr<mytype> p(new mytype);  
std::weak_ptr<mytype> w = p;
```

# std::weak\_ptr



```
std::shared_ptr<mytype> p(new mytype);  
std::weak_ptr<mytype> w = p;  
p.reset();
```