

ProgramConfig



Program Config

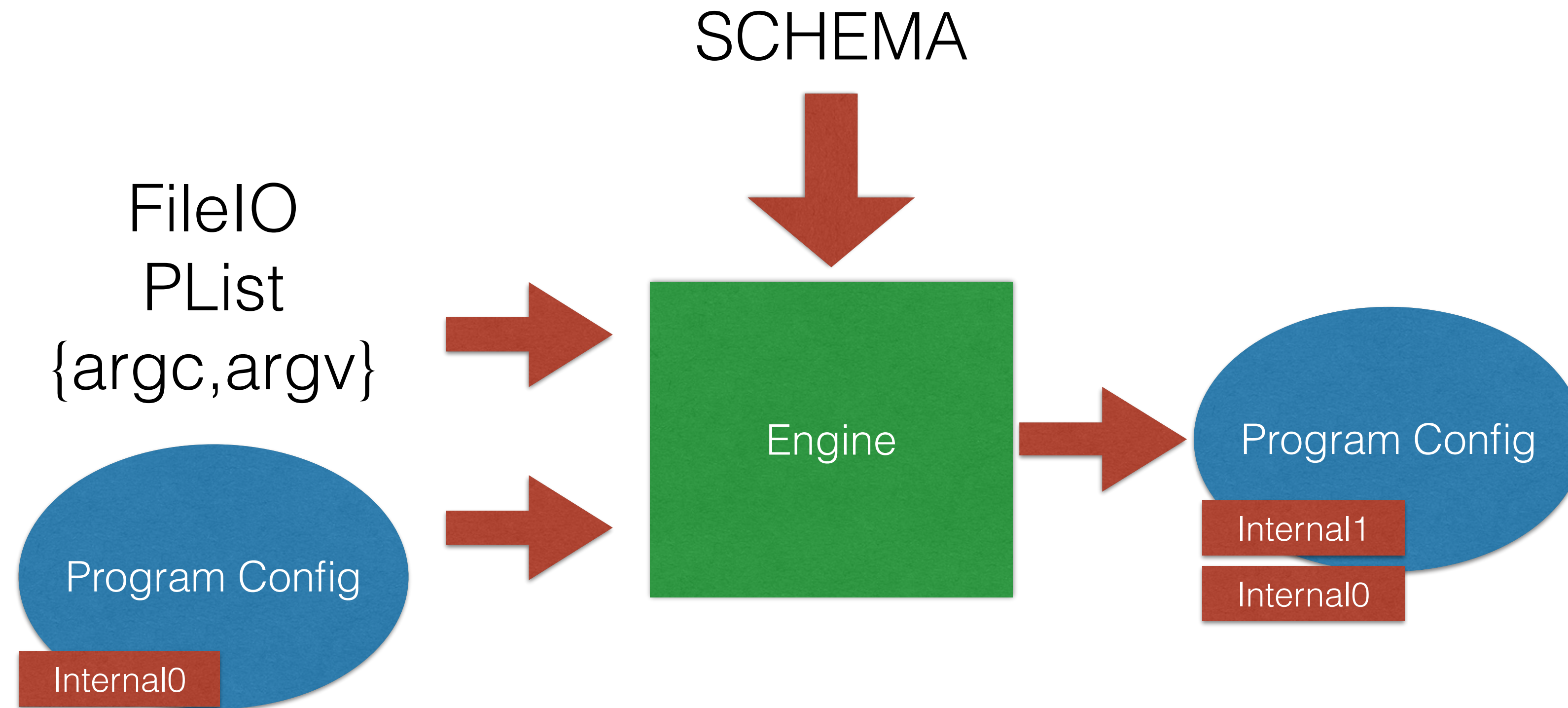
```
class ProgramConfig
{
public:
    template<ReturnType T>
    void GetValue(key_type_t key, T&
value);

    template<ReturnType T>
    T GetValue(key_type_t key);
    ...
};

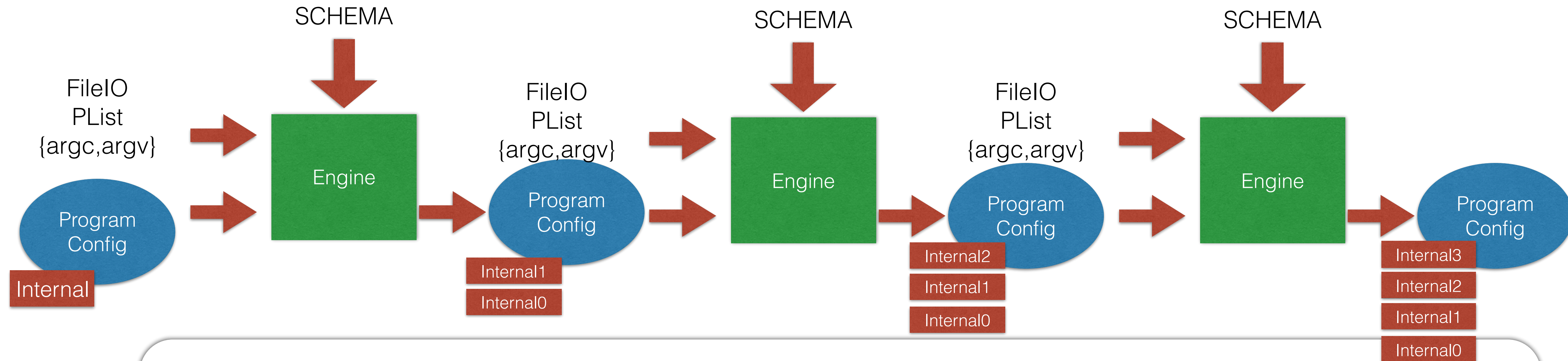
ProgramConfig pc;

auto timeout =
pc.GetValue<float>("TimeOut");
```

- Property List “like” object
- Keys -> Value
- Type conversion



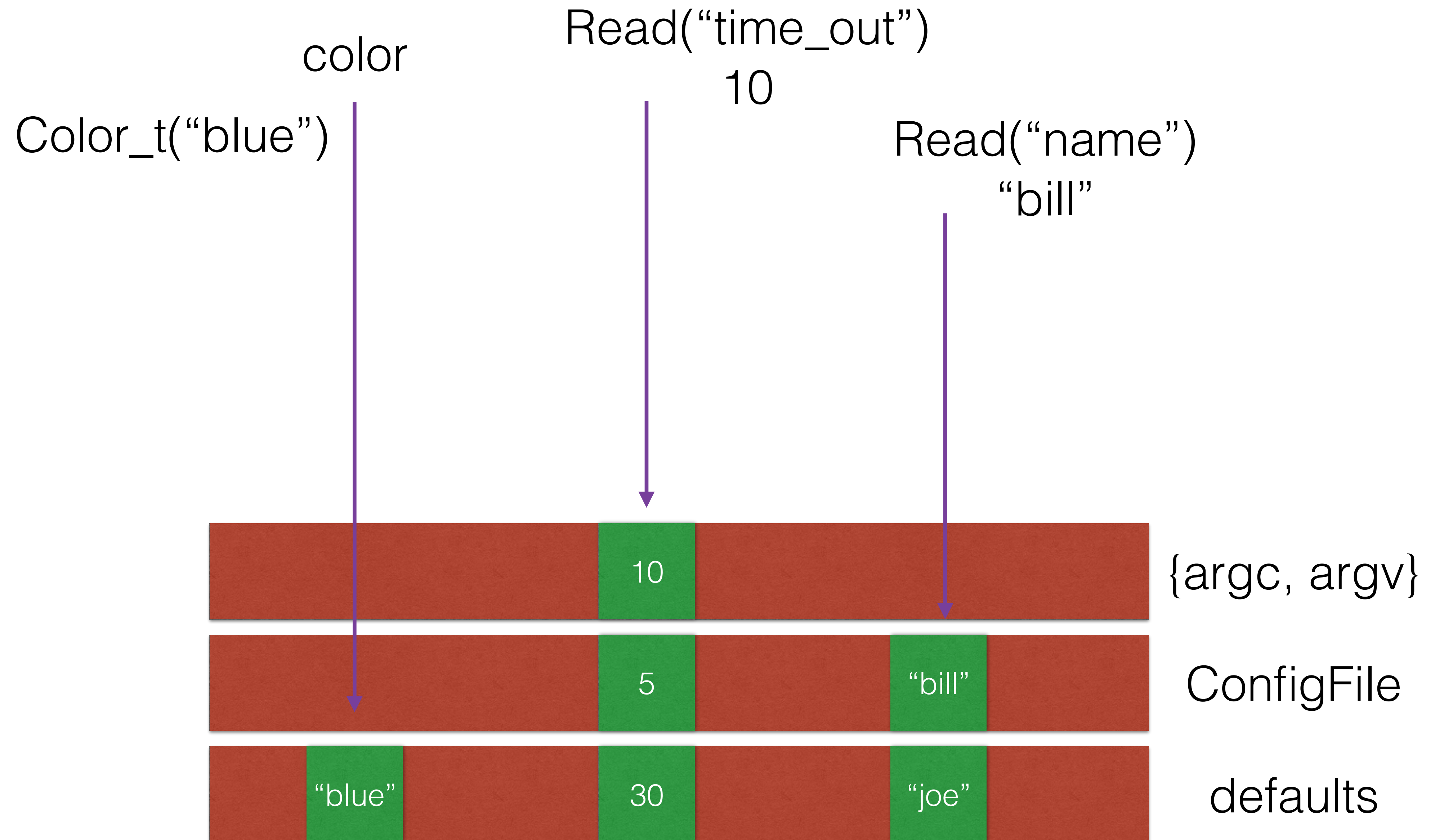
```
int main(int argc, char **argv)
{
    auto myProgramConfig = ProgramConfig::CreateProgramConfig(argc, argv, CMDLINE_SCHEMA);
}
```



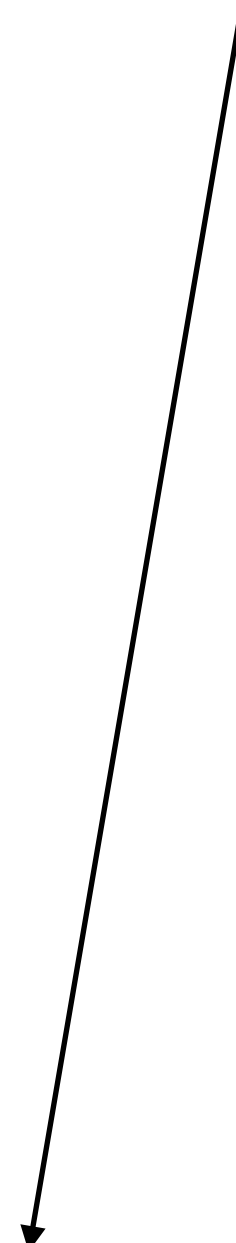
```
int main(int argc, char **argv)
{
    auto defaults = PropertyList{ { "time_to_run", "10" }, { "user_name", "Bob" } };

    using FILE_t = std::unique_ptr<FILE, std::function<int(FILE*)>>;
    FILE_t fromFile(fopen("myConfig.plist"), fclose);

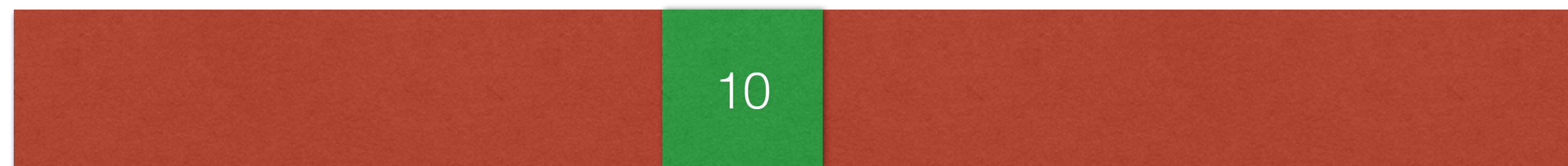
    auto myProgramConfig = ProgramConfig::CreateProgramConfig().
        CreateProgramConfig(defaults, CONFIG_SCHEMA).
        CreateProgramConfig(fromFile, CONFIG_SCHEMA).
        CreateProgramConfig(argc, argv, CMDLINE_SCHEMA);
}
```

Notification
“Something Changed”

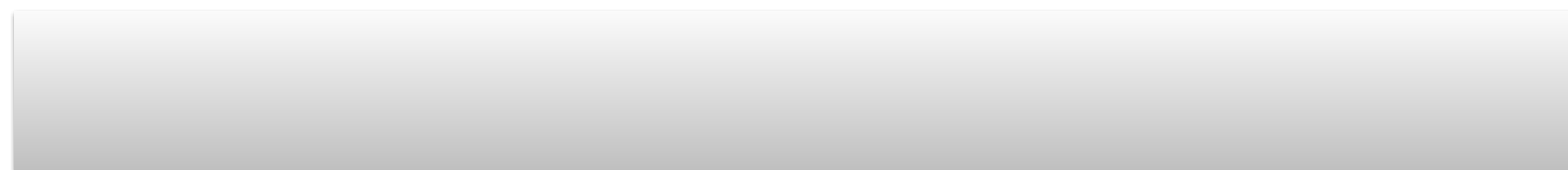


main_thread

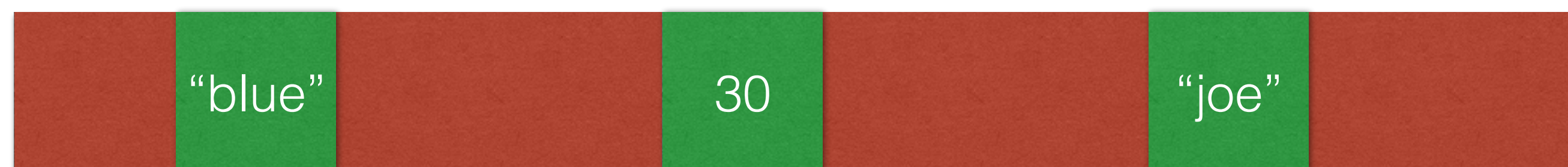


{argc, argv}

Invalidate(layer1);



ConfigFile

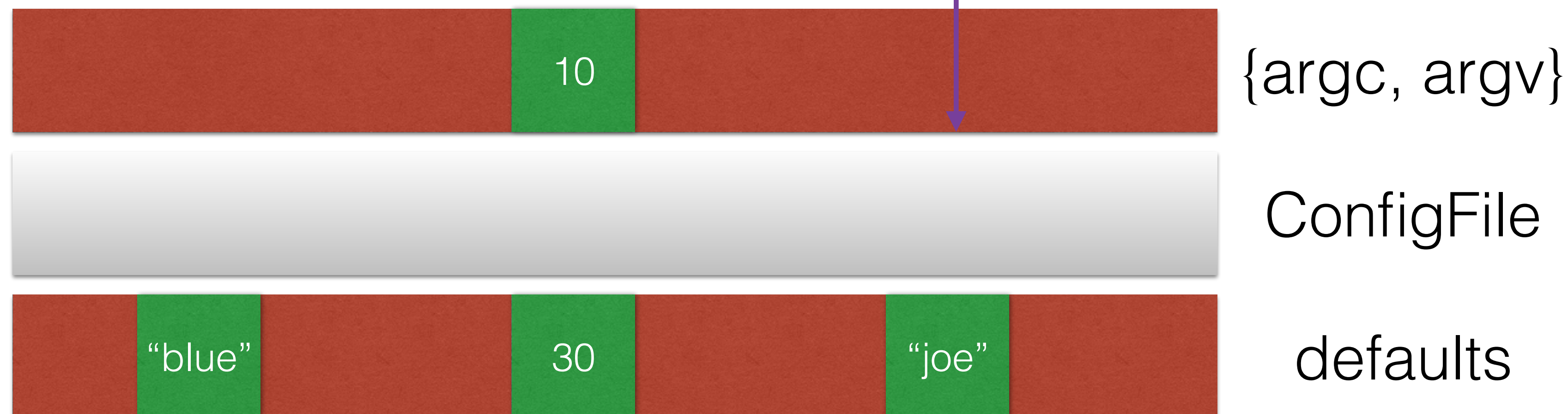


defaults

Notification
“Something Changed”

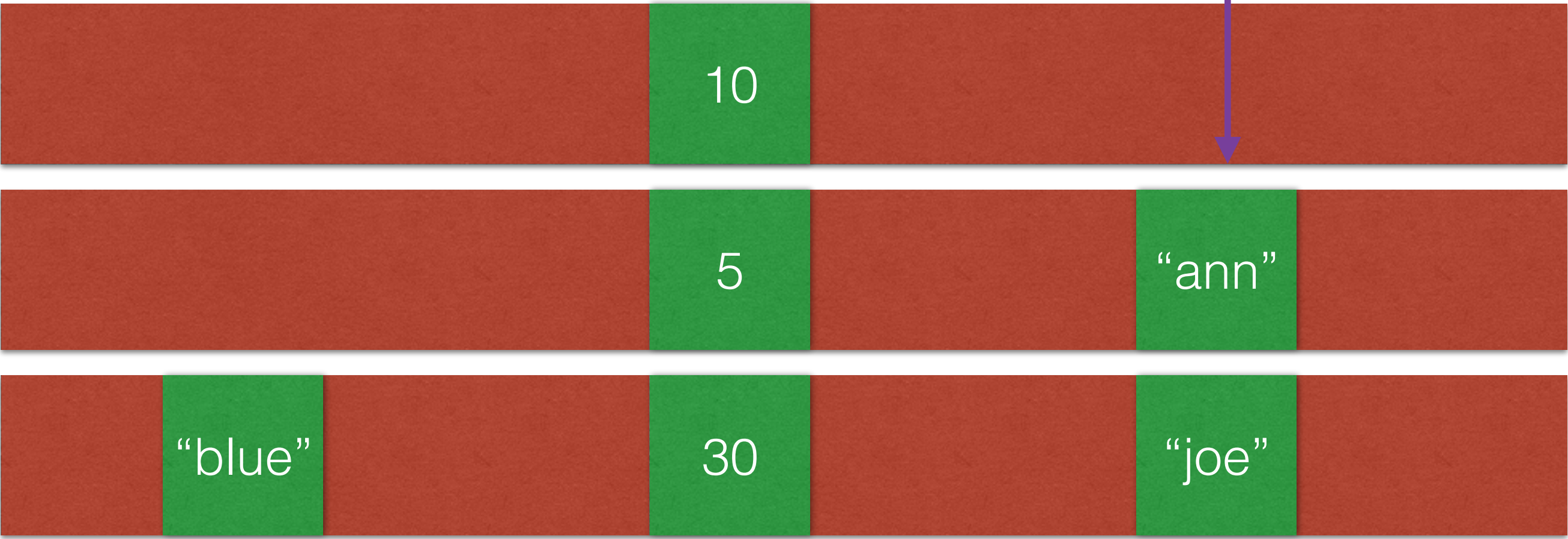
running_thread

Read(“name”)



Read("name")

"ann"



{argc, argv}

ConfigFile

defaults