

Writing DWARF-Services

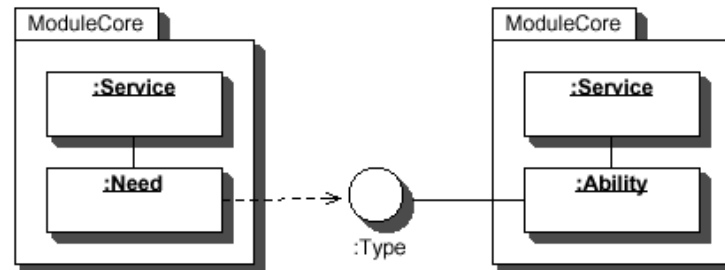


- About Services, Needs and Abilities
- So how do I create them
- Sample code
- Next steps

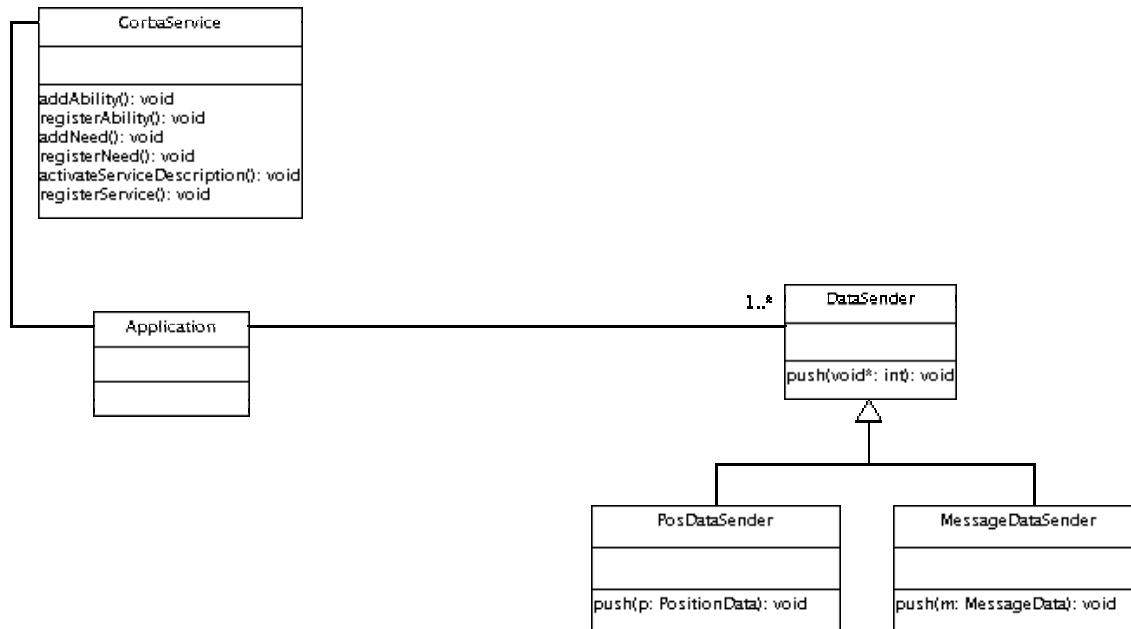
- SW Modules that fit into the DWARF framework
- Low-level services (id-tracker, optical tracker, display) communicating with high-level services (taskflow engine, world model)
- Service discovery and communication via Service Manager

Needs and Abilities

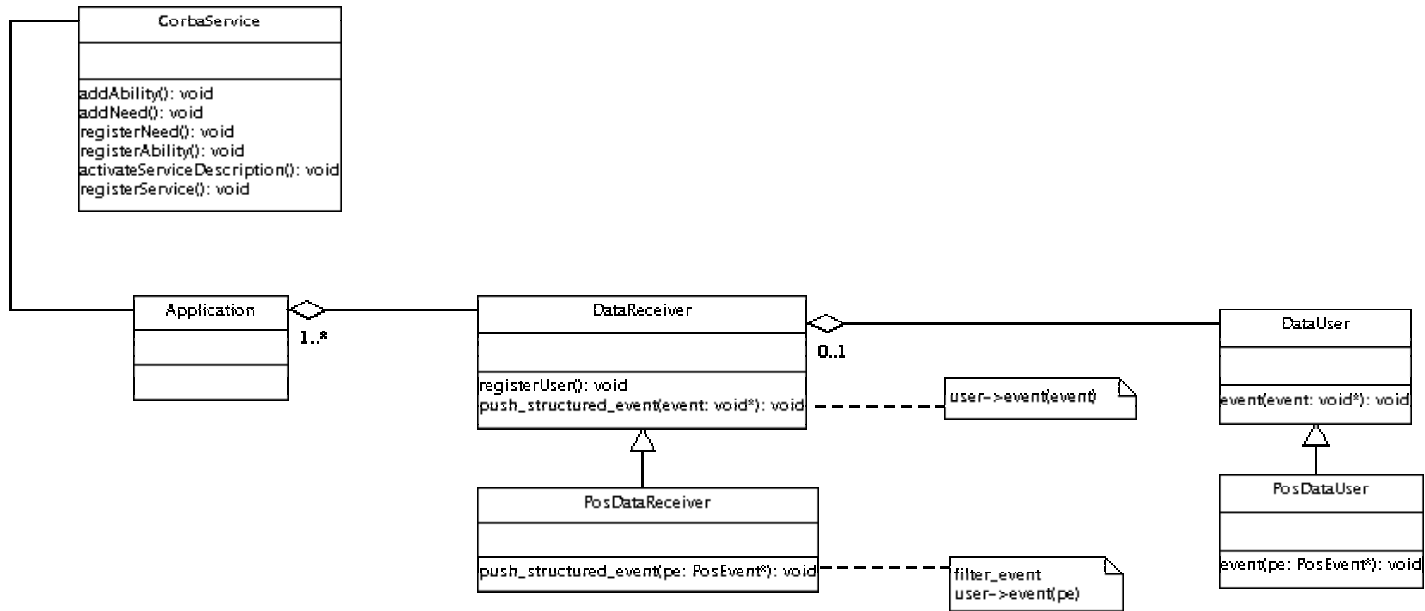
- Framework-compatible services describe abilities and needs
- Service Manager automatically connects matching needs and abilities by type
- Needs and abilities are described in the service description



Describing Abilities



Describing Needs



```
#include "corbaservice.h"
#include "posdatasender.h"
#include "datauser.h"
#include "posdatauser.h"
#include "posdatareceiver.h"

void main() {
    CorbaService s("corbaloc:localhost:39273/ServiceManager", "fakeposition");
    PosDataUser* user = new PosDataUser();

    s.addNeed("fakeposition", "PositionData", "NotifyStructuredPushSupply",
             "NotifyStructuredPushConsumer");
    s.activateServiceDescription();

    s.registerService();

    PosDataReceiver* consumer = new PosDataReceiver("fakeposition");
    consumer->setUser(user);

    s.registerNeed("fakeposition", consumer->_this());
    // application logic
}
```

Next steps

- Grab DWARF-Support package (OmniORB + Service Manager)
- Get sample programs (not yet available, monitor bboards for announcement)
- Write interface definitions (idl) for data types you need to send/receive
- Write your services

