

omniORB and CORBA

Dr Duncan Grisby

Apasphere Ltd



Agenda

about me

ORBA landscape

future directions

distribution of data

About me

Maintainer of omniORB

- C++ and Python CORBA implementation

Co-founder of Apasphere Ltd

- omniORB support, consulting

Co-founder and Chief Software Architect

Disclaimer

don't know much about Tango

- But I do know quite a lot about omniORB and CORBA

These are my personal opinions

State of CORBA

quietly successful

Web services hype has died down

Not much useful added since CORBA 2.6

Increasing complexity of specifications

CORBA future

CORBA will continue to be widely used

OMG will continue to add things to CORBA

- OMG has to justify its existence
- I doubt that many things will be useful
- Required simplification is unlikely

State of omniORB

Current: omniORB 4.1.4 / omniORBpy 3.4.

stable

robust

High performance

- Often 5 times faster than TAO

flexible configuration options

Py6 support

omniORB future

.2 branch starting

data marshalling efficiency

improved union footprint

MI?

selected CORBA 3 features?

python 3.x support?

Distribution of data

ORBA is request-response

- Point-to-point
- Streaming is problematic
- Multicast is an uncomfortable fit

event / notification service

- Could theoretically use multicast

OMG DDS

Some good concepts

- Typed interfaces

Really complex specification

- 260 pages core + 216 pages transport spec

- (66 pages for CORBA GIOP)

OMG DDS

IM / PSM complexity

- Separate from CORBA core, but intimately linked to it

Many many QOS policies

fine-grained OO design limits

omniORB Connection Management

Extension for high performance data
transport

Streams of oneway calls

- Private connections
- Limited threads to handle calls
- 'Safe' connection closure

omniORB Connection Management

```
interface Example {  
    oneway void push(in string val);  
};  
  
= makeRestrictedReference(  
    obj, 123, 1, 1, true, false, true);
```

Closing Thoughts

simplicity is undervalued

- Especially in standards organisations

better to provide simple building blocks

- Easier to understand

- Don't pay for what you don't need



Questions?

