"CORBA"DezSys-04

Python\_Client und C++\_Server

2014

Ari Aayvazyan - Samuel Schmidt

4AHIT - TGM

09.01.2014

Inhaltsverzeichnis

[AugabenStellung 2](#_Toc377051410)

[Resources 2](#_Toc377051411)

[Arbeitsaufteilung, aufwandschätzung & tatsächliche zeit 2](#_Toc377051412)

[Arbeitsdurchführung 3](#_Toc377051413)

[Testbericht 5](#_Toc377051414)

<https://github.com/aayvazyan-tgm/CORBA>

# AugabenStellung

Verwenden Sie das Paket ORBacus oder omniORB bzw. JacORB um Java und C++ ORB-Implementationen zum Laufen zu bringen.  
  
Passen Sie eines der Demoprogramme so an, dass Sie einen Namingservice verwenden, welches ein Objekt anbietet, das von jeweils einer anderen Sprache (Java/C++) verteilt angesprochen wird. Beachten Sie dabei, dass eine IDL-Implementierung vorhanden ist um die unterschiedlichen Sprachen abgleichen zu können.

Vorschlag: Verwenden Sie für die Implementierungsumgebung eine Linux-Distribution, da eine optionale Kompilierung einfacher zu konfigurieren ist.

### Resources

http://omniorb.sourceforge.net/  
http://www.microfocus.com/products/corba/orbacus/  
http://www.jacorb.org/  
http://omniorb.sourceforge.net/omni41/omniORB.pdf  
http://www.ing.iac.es/~docs/external/corba/book.pdf

# Arbeitsaufteilung, aufwandschätzung & tatsächliche zeit

Aayvazyan

* Python-Client
* Doku

Schmidt

* C++-Server
* Doku

|  |  |  |
| --- | --- | --- |
|  | Geschätzte Zeit | Tatsächliche Zeit |
| Aayvazyan Ari | 3h | 4h 30m |
| Schmidt Samuel | 3h | 4h |

# Arbeitsdurchführung

Resultate:

**C++ Client und Server:**

C:\Users\Samuel\Desktop\python.PNGC:\Users\Samuel\Desktop\omni.PNG

OmniORB 4.1.6 von <http://sourceforge.net/projects/omniorb/files/omniORB/> für die Demos und danach entpacken

C:\Users\Samuel\Desktop\entpacken.PNG

OmniORB verschieben und neuen build Ordner erstellen und dort *make* und *make install*

C:\Users\Samuel\Desktop\make.PNGC:\Users\Samuel\Desktop\install.PNGC:\Users\Samuel\Desktop\pyidl.PNG

**Python Client und Server:**

Um das python script auf debian zum laufen zu bekommen müssen folgende schritte durchgeführt werden:

Python-all-dev installieren

omniORB und omniORBpy konfigurieren & installieren.

Interface in IDL Spezifizieren

interface Echo {

string echoString(in string mesg);

};

In echo.idl speichern

Die IDL wird für c++ compiliert:

omniidl -bcxx echo.idl

Das selbe geschieht für python mit

omniidl -bpython echo.idl

Und nun wird auch noch die example\_echo.idl:

module Example {

interface Echo {

string echoString(in string mesg);

};

};

Für python kompilliert

omniidl -bpython example\_echo.idl

In /etc/omniORB.conf muss folgende zeile angepasst werden:

InitRef = NameService = corbaname::localhost

Nun lässt sich mittels python main.py das pythonscript starten

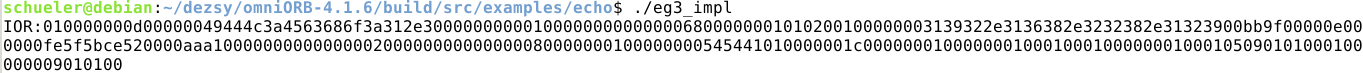
Niederlagen:

*Object reference is not an Example::Echo* bei Testfall mit Python Client und C++ Server.

# Testbericht

**C++ Client und Server:**

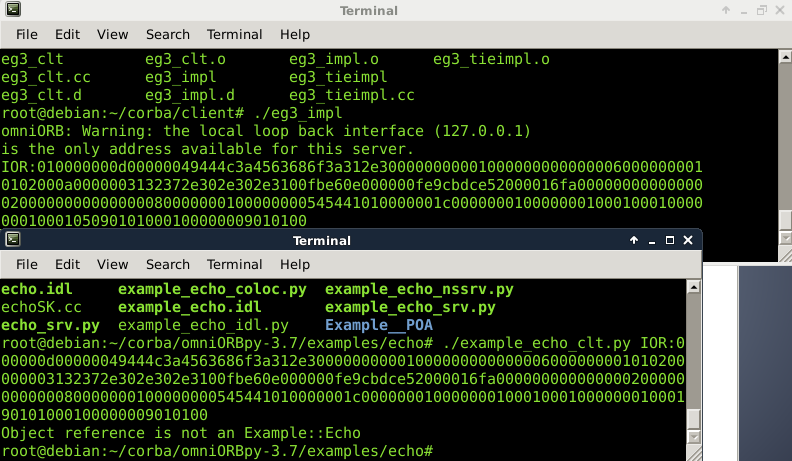
**Server:**



Client:



**Python Client und C++ Server:**



# Quellenangaben

<http://sourceforge.net/projects/omniorb/files/omniORB/>

<http://omniorb.sourceforge.net/>  
<http://www.microfocus.com/products/corba/orbacus/>  
<http://www.jacorb.org/>  
<http://omniorb.sourceforge.net/omni41/omniORB.pdf>  
<http://www.ing.iac.es/~docs/external/corba/book.pdf>