

Problem

Why do I get omniORB.CORBA.BAD_PARAM: Minor: BAD_PARAM_WrongPythonType, COMPLETED_NO exceptions when invoking CORBA methods of components?

Solution

This occurs when you try to pass Python objects of the incorrect CORBA type to CORBA objects as *in* or *inout* parameters. omniORB does not provide any nice mechanism for determining precisely which parameter is bad so you'll have to look at them one at a time. The IDL->Python mapping document, <http://www.omg.org/technology/documents/formal/python.htm>, is a good starting point for figuring out which parameter is incorrect.

Of particular interest to developers is the fact that it's possible to create CORBA objects (an instance of an IDL *struct* for example) using the wrong Python object types. Then if you try to pass that CORBA-based object to some real CORBA method as an *in* or *inout* parameter it will fail with the exception indicated above! This can be very hard to debug.

Related articles

- [How can more people do development with ACS on the same machine without disturbing each other?](#)
- [Which ports are used by ACS?](#)
- [Problems connecting to ACS servers on a remote machine: bad /etc/hosts](#)
- [Why does the getComponent method of ZLegacy/ACS.ContainerServices return an object of type None?](#)
- [Why are some of my print statements not showing up in the container output section of acscommandcenter?](#)