

CORBA Exceptions

Prof. Dave Bakken

Cpt. S 464/564 Lecture (part of it)

October 2, 2000

Administrative Items

- Handouts today
 - New Syllabus
 - Note **MIDTERM DATE CHANGE** and **Project 3+4** out/due (Dad's weekend)
 - Project 2 Description
 - Project 1 Solution Code (C++ and Java)
 - These slides (didn't quite make 8am on the web....)
- Note a few typos in Chapter 5 ...
 - Code Sample 5.8 (Java)

```
if (_balance < 50) {  
    throws AccountFrozen();  
}
```
 - Code sample 5.10 (IDL)

```
interface Account {  
    exception AccountFrozen {  
        int reason;  
    }  
}
```

Exceptions

- Errors happen.....
 - Significant % of real-world program code is for error and boundary checking
 - Doubly true for distributed programs!
- Error codes obfuscate a method's interface

```
typedef float SampleSet[10];  
float mean (in SampleSet samples);
```

becomes

```
typedef float SampleSet[10];  
typedef long ErrorCode;  
ErrorCode mean (in SampleSet samples, out float the_mean);
```

- Fundamentally ugly!
- Error codes have to be dealt with by each level of call in a deep nesting
 - Even if the caller does not know what to do with it
 - Lots of encoding and recoding of error messages up the calling chain...

Exceptions (cont.)

- Language support for exceptions allow errors to not be dealt with on the normal execution path – they are “exceptional” conditions!
 - Separate control path is provided for exceptions
- Advantages
 - Error conditions are dealt with at only at the level it makes sense to
 - Code can be written as if the methods called will work

Exceptionless C and Java (Local) Example

```
/* look up entry for "ButchTheCoug"  
/* in the "employee" table  
* On failure, "error" will be allocated  
* and filled in */
```

```
ErrorStruct *error = NULL;  
TableEntry *entry = lookup(  
    "ButchTheCoug",  
    "employee", &error);
```

```
if (entry == NULL)  
    return error;  
}
```

```
// Same as C example, but in Java
```

```
ErrorStruct error = new ErrorStruct;  
TableEntry entry = lookup(  
    "ButchTheCoug", "employee", error);
```

```
if (entry == NULL)  
    return error;  
}
```

```
// other variants are equally  
unsatisfactory...
```

(Note: this example was borrowed from "Java 1.2 by Example" by Jackson and McClellan)

Exception Support in Java (C++ is Similar)

- An exception is **thrown** by a method

```
public class MyException extends Exception {
```

```
    ...
```

```
}
```

```
...
```

```
public static void myMethod() throws MyException {
```

```
    ...
```

```
        throw new MyException(args);
```

```
    ...
```

```
}
```

- Exception must either
 - be caught by caller of myMethod(), or
 - Caller's method signature must indicate it throws MyException
- Thrown exception propagates up the calling chain until
 - It is "caught", or
 - Program exits

Exception Support in Java (cont.)

- **try/catch** block used by caller

```
try {  
    method1();  
    method2();  
    ...  
} catch (Exception e) {  
    System.err.println("Error: " + e.getMessage() );  
}
```

- Can also catch multiple exception classes (more specific first...)

Exceptions in CORBA

- Exceptions are a lot like IDL structs, but
 - No nesting
 - No inheritance
- CORBA System exceptions
 - Predefined by OMG (29 different ones: **COMM_FAILURE**, **INV_OBJREF**, ...)
 - Thrown by the ORB
- CORBA User Exceptions
 - Defined in IDL by programmer
 - Thrown by server application code
- CORBA exceptions work across programming languages
- See PROG Chapter 5 for more information on programming with them

VBJ Exception Example

```
// Bank.idl
module Bank {
  interface Account {
    exception AccountFrozen {
      string reason;
      long severity;
    };
    float balance() raises(AccountFrozen);
  };
};
```

VBJ Exception Example (cont.)

```
public class AccountImpl
  extends Bank.AccountPOA {

  public AccountImpl(float balance) {
    _balance = balance;
  }

  public float balance()
    throws AccountFrozen {

    if (_balance < 50) {
      throw AccountFrozen();
    }
    else
      return _balance;
  }
}
```

```
// client
...
try {
  ...
  float balance = account.balance();
  ...
  // catch user exceptions
} catch (Account.AccountFrozen e) {
  System.err.println("AccountFrozen rtn:");
  System.err.println(e);
  // catch CORBA system exceptions
} catch (org.omg.CORBA.SystemException
  sys_excep) {
  System.err.println("CORBA SysExcep:");
  System.err.println(sys_excep);
}
...
}
```