

JavaプログラミングのTip集

第3回

2007年6月10日
アイティエス

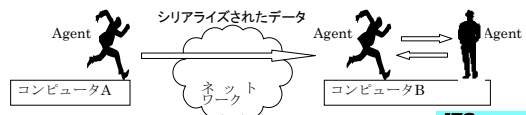


モバイル性に必要なファイル技術

シリアライズされたデータが他のコンピュータに移動する

ObjectOutputStream } 使用
ObjectInputStream }

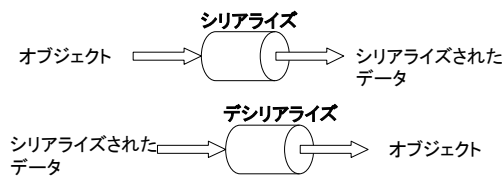
■ 最終的には
➤ データの圧縮・暗号化が必要 } ZipInputStream
ZipOutputStream



シリアライズとは

オブジェクトをストリームに読み書きする機能
シリアライズ→直列化

➤ ネットワーク上に流れるデータの形態



プログラムにみるシリアライズの構成・・・writeObject

```

FileOutputStream fo = null;
ObjectOutputStream oo = null;
try{
    fo = new FileOutputStream(fname);
    oo = new ObjectOutputStream(fo);
    oo.writeObject(obj); ← シリアライズのデータの書きだし
    oo.flush();
    oo.close();
}catch(IOException e){
    e.printStackTrace();
}
    
```

ここでシリアライズ



プログラムにみるデシリアライズの構成・・・readObject

```

Object getObject = null;
ObjectInputStream oi = null;
FileInputStream fi = null;
try {
    fi = new FileInputStream(fname);
    oi = new ObjectInputStream(fi);
    getObject = oi.readObject(); ← シリアライズのデータの読み込み
    oi.close();
    fi.close();
}catch(FileNotFoundException e){
    e.printStackTrace();
}catch(IOException e){
    e.printStackTrace();
}catch(ClassNotFoundException e){
    e.printStackTrace();
}
    
```

ここでデシリアライズ



シリアライズのデータフォーマット

