Subject: Re: broken pick semantics
Posted by crydev on Thu, 15 Nov 2012 21:59:53 GMT
View Forum Message <> Reply to Message

I have read those topics and I understand how and why you did that. Though the information from those topics still couldn't help me find out my problem. I fixed it though! I created a reference to the global variable, and the error did not pop up anymore.

The problem didn't reside in the GetCount() in the above post as you said, the problem resided in my synchroniser:

```
void MovieSynchroniser::Synchronise(Vector<String> pDirectories)
{
 for (int i = 0; i < pDirectories.GetCount(); i++)
 {
  GetFiles(pDirectories[i]);
 }
}
```

This function was called using the global variable as parameter. I changed the function definition into:

```
void MovieSynchroniser::Synchronise(Vector<String>& pDirectories)
```

It makes sense to me though. Because I used a normal value object as parameter it already copied the contents of the vector, which using pick semantics, resulted in the original vector being empty. By using a reference pointer to the original vector, the pick assignment is not used, avoiding this problem. Am I correct?