

---

Subject: BUG: EditKeys

Posted by [mrjt](#) on Thu, 29 Oct 2009 16:32:42 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

Using key group names with spaces causes RestoreKeys to fail. One symptom of this is that if the user clicks 'Cancel' on the EditKeys dialog then the key settings revert to application defaults.

My fix is to alter the processing of group names in Restore keys by using a ReadGroupName function (previously it used ReadId(), so stopped at spaces):

```
String ReadGroupName(CParser &p) {
    const char *start = p.GetPtr();
    const char *ptr = start;
    while (*ptr && *ptr != ';') {
        p.GetChar();
        ++ptr;
    }
    return String(start, ptr - start);
}
```

```
void RestoreKeys(const String& data)
{
    SetDefaultKeys();
    CParser p(data);
    try {
        while(!p.IsEof()) {
            try {
                p.PassChar('-');
                String group = ReadGroupName(p);
                p.PassChar(';');
            }
        }
    }
}
```

etc...

You can probably find a more elegant way though

---

---

Subject: Re: BUG: EditKeys

Posted by [mirek](#) on Fri, 30 Oct 2009 13:57:35 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

Thanks, a good observation.

Fixed this way:

```
String StoreKeys()
{
    String out;
    const VectorMap<String, Vector<KeyBinding> >& g = sKeys();
    for(int i = 0; i < g.GetCount(); i++) {
```

```
out << "- " << AsCString(g.GetKey(i)) << ";\r\n";
```

```
void RestoreKeys(const String& data)
{
    SetDefaultKeys();
    CParser p(data);
    try {
        while(!p.IsEof()) {
            try {
                p.PassChar('-');
                String group;
                if(p.IsId()) // backward compatibility
                    group = p.ReadId();
                else
                    group = p.ReadString();
            }
        }
    }
}
```

---