Subject: CParser do not check for invalid strings that span lines
Posted by mingodad on Sat, 12 Apr 2014 10:48:34 GMT
View Forum Message <> Reply to Message

Hello !

Working with witz templates I found error messages a bit misleading when I forgot to put an end quote on one string, following the problem I found several problems (it got better but the error message has a wrong column info on my case it is 2 bytes upfront):

1- CParser do not check for invalid strings that spam multiple lines (see fix bellow):
/////////

```
--- /home/mingo/upp/uppsrc/Core/parser.cpp
+++ /tmp/parser-0.cpp
@@ -327,7 +327,7 @@
    }
   }
   else {
-   if(*term < ' ') {
+   if(*term < ' ' || *term == '\n') {
     if(chkend) {
      ThrowError("Unterminated string");
      return result;
```

////////

2 - Witz parser do not ask to check string end, also when calling "Block()" recursivelly line count get reseted (fix bellow):
////////

```
--- /home/mingo/upp/uppsrc/Skylark/Compile.cpp
+++ /tmp/Compile-0.cpp
@@ -199,7 +199,7 @@
   if(p.Char('0'))
    c.value = int(p.IsNumber() ? p.ReadNumber(8) : 0);
   else
-   c.value = p.IsString() ? Value(p.ReadString()) : Value(p.ReadDouble());
+   c.value = p.IsString() ? Value(p.ReadString(true)) : Value(p.ReadDouble());
  }
  return result;
 }
@@ -356,16 +356,17 @@
  ExeBlock& blk = result.Create<ExeBlock>();
  const char *s = p.GetSpacePtr();
  const char *b = s;
- int line = 1;
```

```
  while(*s) {
   if(*s == '$') {
    if(s[1] == '$') {
     blk.AddText(b, s + 1);
-    p.Set(s + 2, NULL, line);
+    p.Set(s + 2, NULL, current_line);
     b = s = p.GetSpacePtr();
    }
    else {
     blk.AddText(b, s);
-    p.Set(s + 1, NULL, line);
+    p.Set(s + 1, NULL, current_line);
     if(p.Id("if")) {
      ExeCond& c = blk.item.Add().Create<ExeCond>();
      p.PassChar('(');
@@ -403,15 +404,15 @@
     else
      blk.item.AddPick(Prim());
     b = s = p.GetSpacePtr();
-    line = p.GetLine();
+    current_line = p.GetLine();
    }
   }
   else
   if(*s++ == '\n')
-   line++;
+   current_line++;
  }
  blk.AddText(b, s);
- p.Set(s, NULL, line);
+ p.Set(s, NULL, current_line);
  return result;
 }

////////

--- /home/mingo/upp/uppsrc/Skylark/Witz.h
+++ /tmp/Witz-0.h
@@ -194,7 +194,6 @@
  Vector<bool>  forvar;
  bool          optimized;
  int           count_node;
+ int     current_line;

  int ForVar(String id, int i);

@@ -231,7 +230,7 @@
```

```
  static void Register(const String& id, Value (*fn)(const Vector<Value>&, const Renderer *));

- Compiler(const char *code, const Index<String>& var) : p(code), var(var, 1) {
forvar.SetCount(var.GetCount(), false); }
+ Compiler(const char *code, const Index<String>& var) : p(code), var(var, 1) { current_line = 0;
forvar.SetCount(var.GetCount(), false); }
 };

 One<Exe> Compile(const char *code, const Index<String>& vars);

//////////
```

---