

Quoted Name Searching in Pyparsing with searchparser.py

September 17, 2010

The [searchparser.py](#) module has a flaw when it comes to quoted phrases with punctuation. Searching for something like this: "C. Montgomery Burns, Esq." results in a nasty stacktrace:

```
Traceback (most recent call last):
  File "searchparser.py", line 302, in
    if ParserTest().Test():
  File "searchparser.py", line 289, in Test
    r = self.Parse(item)
  File "searchparser.py", line 170, in Parse
    return self.evaluate(self._parser(query)[0])
  File "/var/lib/python-support/python2.5/pyparsing.py", line 1049, in parseString
    loc, tokens = self._parse( instring, 0 )
  File "/var/lib/python-support/python2.5/pyparsing.py", line 925, in _parseNoCache
    loc,tokens = self.parseImpl( instring, preloc, doActions )
  File "/var/lib/python-support/python2.5/pyparsing.py", line 2560, in parseImpl
    return self.expr._parse( instring, loc, doActions, callPreParse=False )
  File "/var/lib/python-support/python2.5/pyparsing.py", line 925, in _parseNoCache
    loc,tokens = self.parseImpl( instring, preloc, doActions )
  File "/var/lib/python-support/python2.5/pyparsing.py", line 2431, in parseImpl
    raise maxException
pyparsing.ParseException: Expected "" (at char 2), (line:1, col:3)
```

The reason is that words in `searchparser.py` are defined as consisting of just letters and numbers (or `alphanums`, in `pyparsing`-speak) in lines 94 and 95:

```
Word(alphanums)
```

The solution is to define a string which contains all the possible punctuation characters to expect in a quoted search string, and include it in the parser grammar for word. For example, for people's names, the likely punctuation characters to expect are:

```
punctuation = ",.'`&-"
```


So, using that definition at the start of the `parser(self)` method, we edit the lines for `operatorWord()` to look like this:

```
operatorWord = Group(Combine(Word(alphanums+punctuation) + Suppress('*'))).setResultsName('wordwildcard') |
\
    Group(Word(alphanums+punctuation)).setResultsName('word')
```

With that change, we can find Monty Burns using his exact name, periods and commas included.

"Mr Burns - Excellent"

Archived from the original at <http://denis.papathanasiou.org/>

 Bitcoin Donate: [14TM4ADKJbaGEi8Qr8dh4KfPBQmjTshkZ2](https://blockchain.info/address/14TM4ADKJbaGEi8Qr8dh4KfPBQmjTshkZ2)