Subject: Re: theide with libclang, first milestone finished
Posted by mirek on Sat, 09 Jul 2022 22:49:30 GMT
View Forum Message <> Reply to Message

Novo wrote on Sat, 09 July 2022 19:44mirek wrote on Sat, 09 July 2022 07:11
Now it is definitely a temporary solution as it is quite slow that way, hopefully we will find a fix soon.

I tested latest source code.
I do not see any changes.
Basically, source code where autocompletion doesn't work:

```
bool App::Parse(const String& file_name) {
 using namespace conllu;

 FileMapping s;
 if (!s.Open(file_name))
  return false;
 if (!s.Map(0, static_cast<dword>(s.GetFileSize())))
  return false;

 arr.Clear();
 sent.Clear();
 conlluTree.Clear();

 sentv.Clear();
 Parser p(reinterpret_cast<const char*>(s.Begin()), reinterpret_cast<const char*>(s.End()), sentv);

 while (p.Sentence()) ;

 for (const drtree::Sentence& v: sentv)
  sent.Add(AsString(v.GetWordV().GetCount()), v.GetText());

 return true;
}
```

In case of "arr.", "sent.", "conlluTree.", e.t.c. autocompletion doesn't work.
In case of "s." and "p." it does.

The only reason LibClang is needed is "name resolution". It is much harder to develop a name resolution algorithm than to make a C++ parser itself. If name resolution doesn't work for some reason, then there is no reason to use LibClang.

It works 100% with clang 14 in Windows, hence I think the problem is somewhere else than in clang. It works without preamble in clang 10 for me too. Too soon to try something else at this point.

Now this is one of troubles I expected before going clang - we are now victim of external incompatibilities. But what is done is done.

Quote:There are better solutions for parsing.

And these are?

Mirek
