Current lex/yacc's

- available cup's
  - `/home/java/cup/java_cup.jar`
  - `<WORKDIR>/src/javacup` as classes
  - `<WORKDIR>/src/javacup/docs/java_cup.var`
  - `<WORKDIR>/src/slime/sfcparser/java_cup.var`
- available lex's
  - `/home/java/JLex/JLex.jar`
  - `<WORKDIR>/src/JLex/` as classes
  - `<WORKDIR>/src/JLex/JLex.jar`
  - `<WORKDIR>/src/slime/sfcparser/JLex.jar`

Lex/Yacc (con't)

additionally: The history of JLex (but empty)

```
5         ./JLex/Archive/CVS
5         ./JLex/Archive/1.1.1
5         ./JLex/Archive/1.1.2
5         ./JLex/Archive/1.1.3
5         ./JLex/Archive/1.1.4
5         ./JLex/Archive/1.1.5
5         ./JLex/Archive/1.2
5         ./JLex/Archive/1.2.1
5         ./JLex/Archive/1.2.2
5         ./JLex/Archive/1.2.3
5         ./JLex/Archive/1.2.4
5         ./JLex/Archive/1.2.5
5         ./JLex/Archive/1.2.5-beta6
5         ./JLex/Archive/1.2.5-beta7
```

Classpath

Proposals

- as on our webpage `/home/java/cup/java_cup.jar` at the institutes pool, respectively, when working at home at the place where the user has copied the version
- any other comination of the checked-in cup/lex versions
- empty `CLASSPATH`
- Makefile with `javac -classpath ../..` in it

Why not check-in generated files?

Besides the arguments exchanged on the email-list:

- For instance: Yacc/Lex generated `java`-files, e.g. `SFCParser.java`:
  - sometime *merge conflicts*
  - look at `Error 36` + `Error 36.2`: The compilation errors of `SFCParser.java` are only *reproducible* (and thus *confirmable* by others and thus attackable! with specific combinations of CLASSPATH/Makefile version invokation of `make`. [a]

  Basically this means

Errors are not reproducible

[a] depending on the invokation, e.g., either the checked in version is taken or a new one is generated which differs from the generated one.