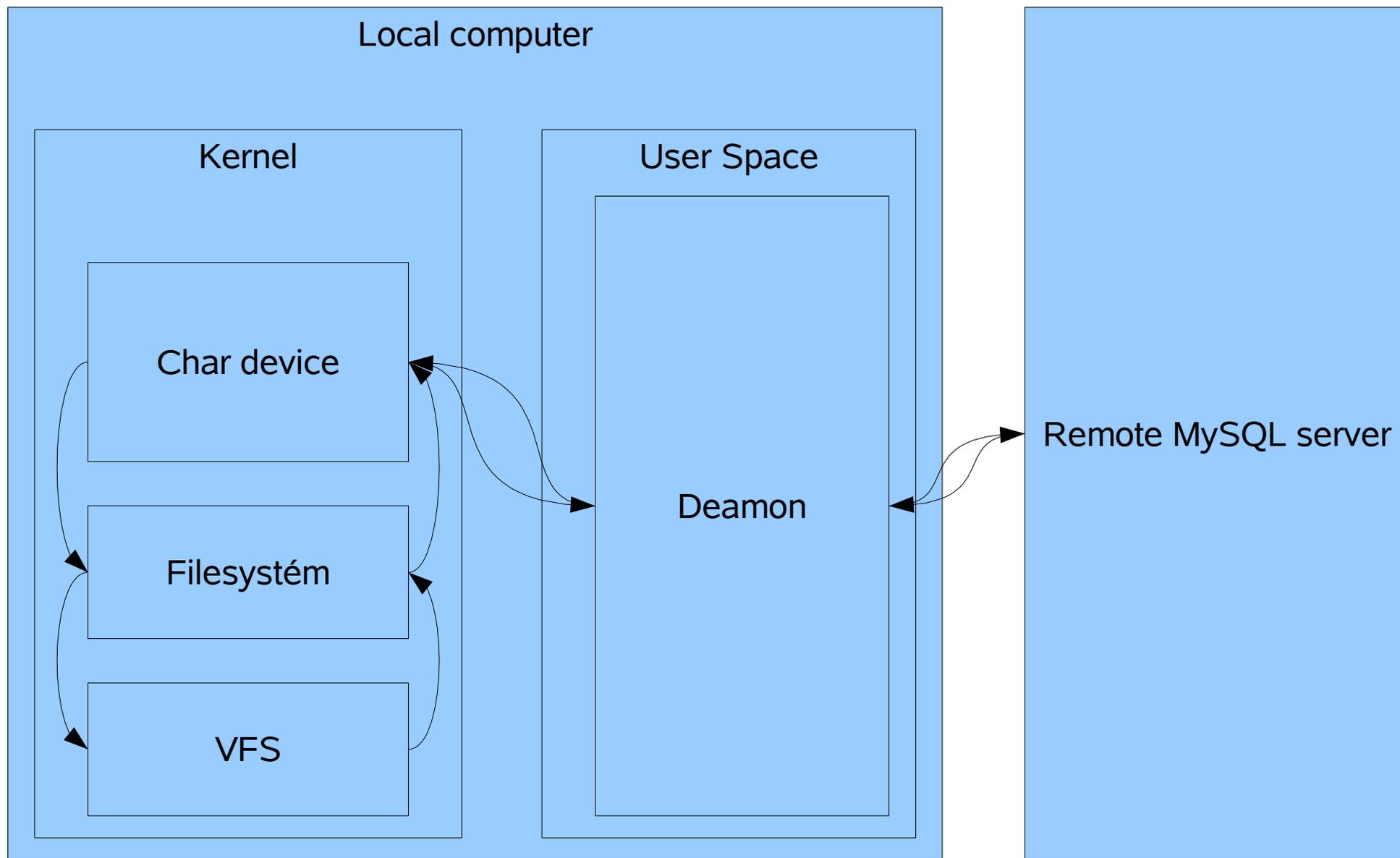


MySQL FS

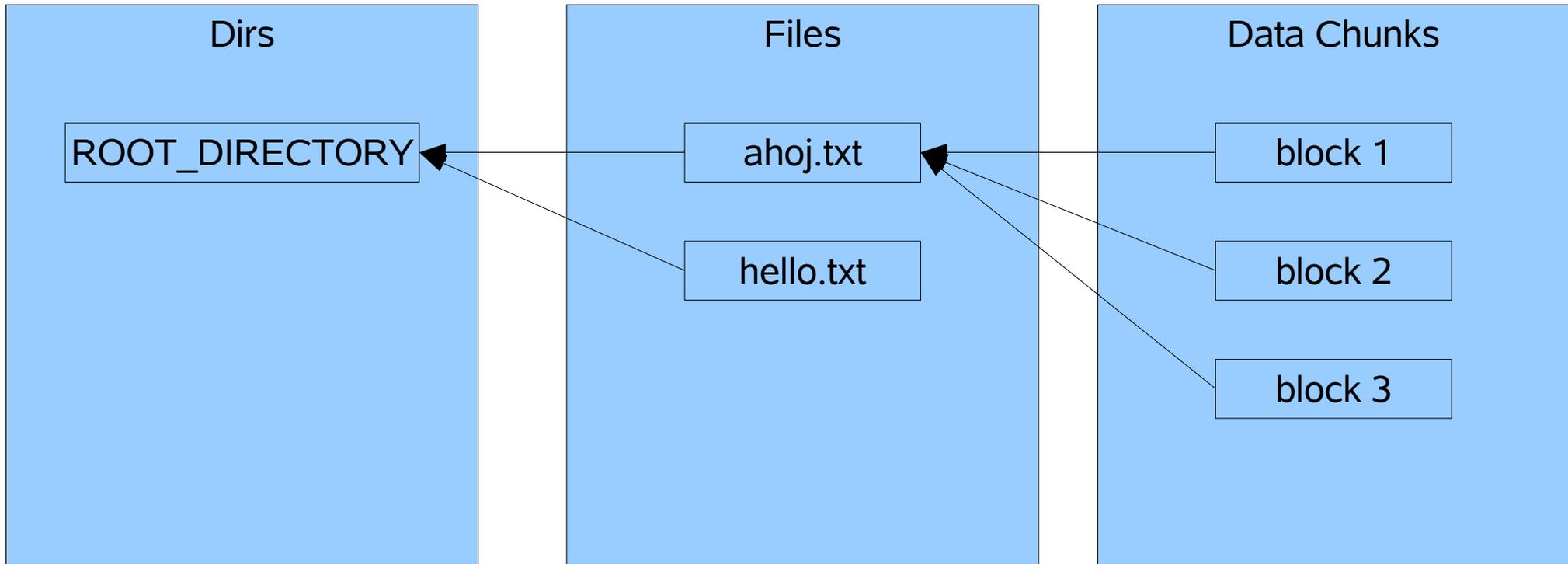
Linux Filesystem, storing data to MySQL database.
Written as a seminar work for „Operační systémy“
on
MFF UK, (Prague, CZ)

Michal Ficek
Michal Podzimek
Tomáš Pop

Principle



MySQL database schema



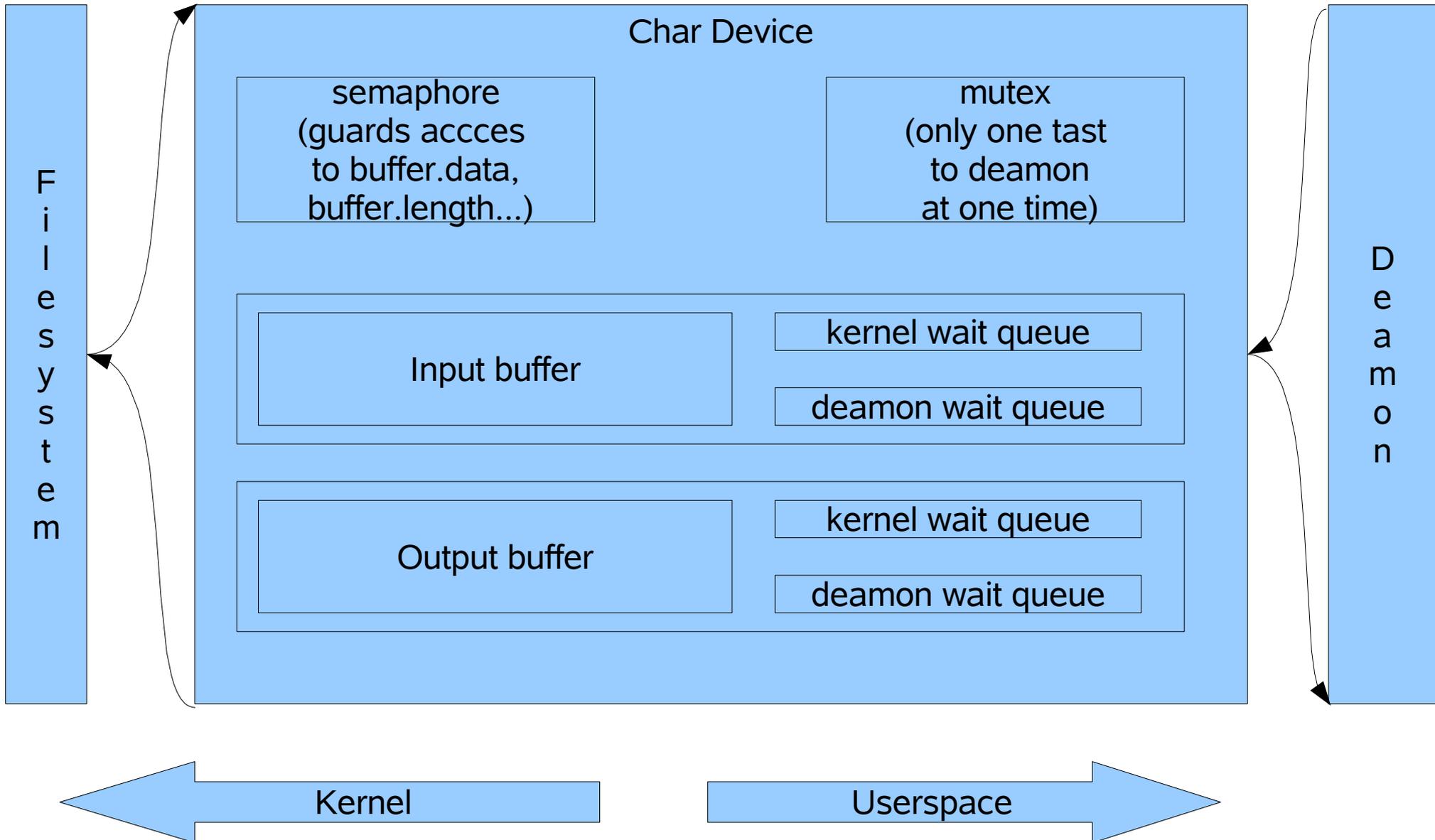
Protokol

- Protokol was written to be as simple as possible.
- Using char device to transef data from user space
- See doc/protokol.txt for more info
- Using strings (not symbolic constantns)

```
READ DIR - list files and dirs from dir, item_count - remaining items
```

```
-----  
FS:"MYSQLFS1.1 RE <dir_id>"  
MD:"MYSQLFS1.1 OK item_count filename/id/filename2/id/.../"  
MD:"MYSQLFS1.1 OK item_count filename/id/filename2/id/.../"  
ERRMD:"MYSQLFS1.1 ER DIR_NOT_EXISTS"  
ERRMD:"MYSQLFS1.1 ER DB_ERROR"  
-----
```

Char Device



Deamon

- User space part
- Based on standart mysql-devel library
- With kernel comunicates thru char device
- Kernel (char device) guarantees that only one reader (one deamon) can open.

Opened Questions

- Multiple mount means multiple char devices and multiple running daemons
- Database name, user and password are hardwired (will be changed)
- No file and dir attributes are stored with files (how to store user when filesystem can be mounted from multiple computers with different users?)

Kernel programming...

„Pýcha předchází bug“

- Kernel stack is 8kb!!!
 - something like example bellow can not work!! and is very hard to find.

```
foo() {  
    char string[5000];  
    char string2[5000];  
    ....  
}
```

Kernel programming...

„Pýcha předchází bug“

- Function `printk()` is a special function with a special behaviour
 - `cat /proc/kmsg` writes only some messages (not deterministically), use `dmsg`.
 - when reading `/proc/kmsg`, only messages not written to terminal are written to `/var/log/messages`