

# Projekt 1

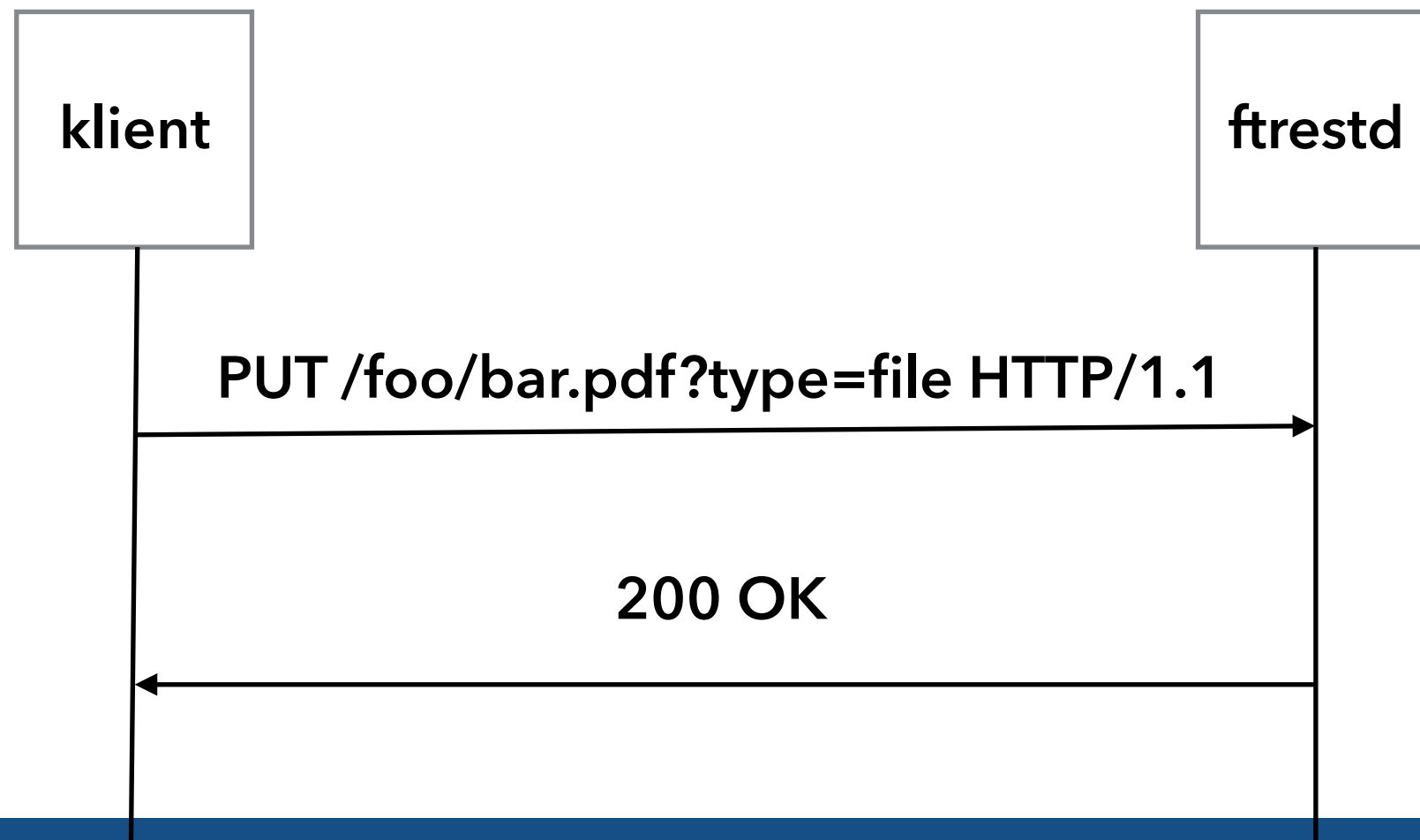
IPK2016L

# Zadání

- Napište klient/server aplikaci pro přenos souborů, který bude komunikovat pomocí HTTP a bude používat jednoduché RESTful API.

\$ftrest put http://server/foo/bar.pdf bar.pdf

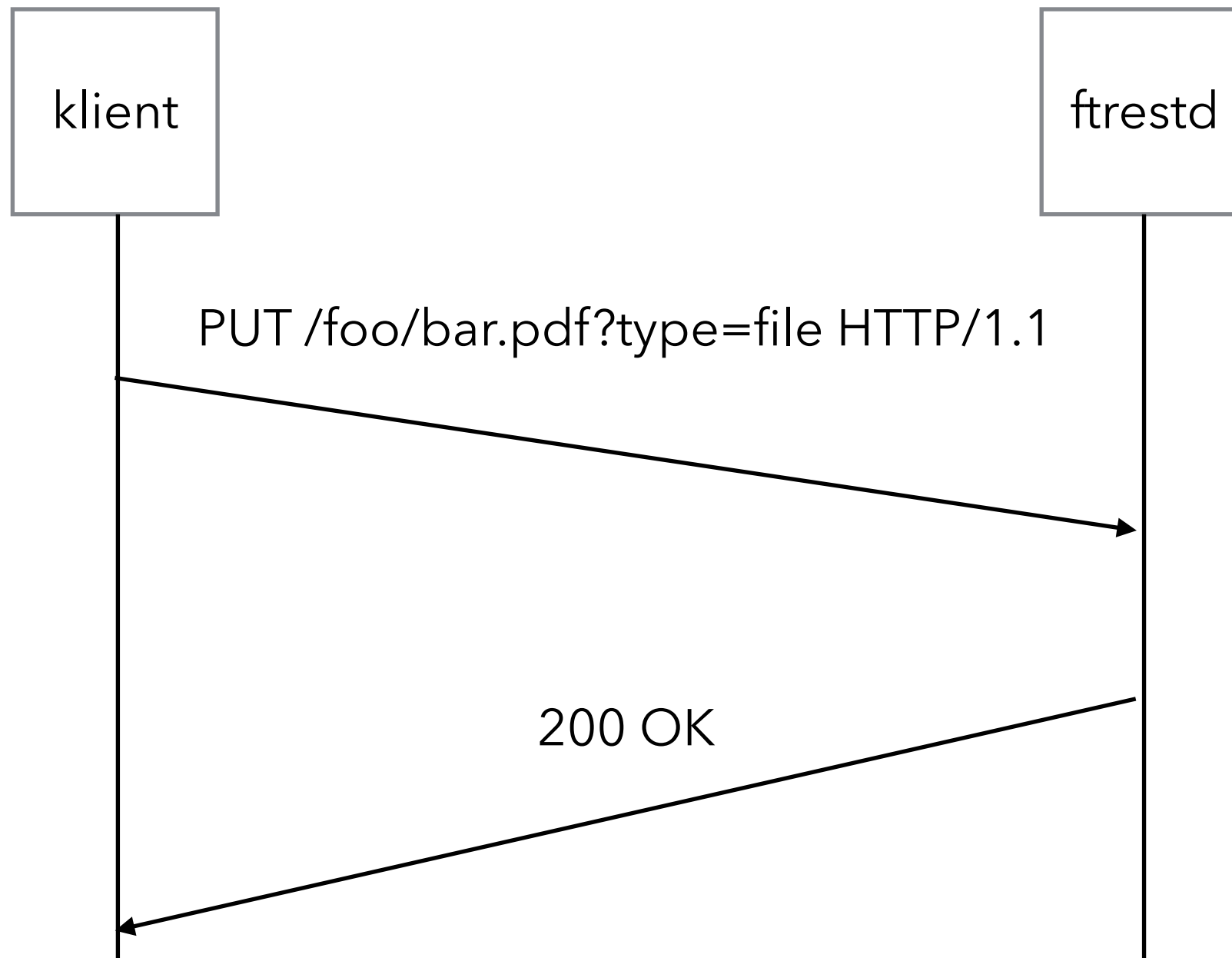
\$server



# Komunikace

```
$ftrest put http://server/foo/bar.pdf bar.pdf
```

```
$server
```



# RESTful Operace

- Nahrání souboru na server:  
**PUT `http://SERVER-NAME/USER-ACCOUNT/REMOTE-PATH?type=[file|folder]`**
- Stažení souboru ze serveru:  
**GET `http://SERVER-NAME/USER-ACCOUNT/REMOTE-PATH?type=[file|folder]`**
- Smazání souboru/adresáře:  
**DELETE `http://SERVER-NAME/USER-ACCOUNT/REMOTE-PATH?type=[file|folder]`**

# Nahrání souboru

**PUT /tonda/foo/bar/doc.pdf?type=file HTTP/1.1**

**Host: somehost.org**

**Date: Wed, 22 Feb 2017 15:50:00 CET**

**Accept: application/json**

**Accept-Encoding: identity**

**Content-Type: application/octet-stream**

**Content-Length: 21557**

**BINARY-CONTENT**

**HTTP/1.1 200 OK**

**Date: Wed, 22 Feb 2017 15:50:01 CET**

**Content-Type: application/json**

**Content-Length: 152**

**Content-Encoding: identity**

**JSON-RESPONSE**

# Server

Server se bude jmenovat ftrestd a bude spuštěn s volitelnými parametry:

**ftrestd [-r ROOT-FOLDER] [-p PORT]**

- -r ROOT-FOLDER specifikuje kořenový adresář, kde budou ukládány soubory pro jednotlivé uživatele, defaultní hodnota je aktuální
- -p PORT specifikuje port, na kterém bude server naslouchat

# Klient

Klient se bude jmenovat ftrest a bude mít dva povinné parametry následované (volitelným) parameterem:

**ftrest COMMAND REMOTE-PATH [LOCAL-PATH]**

- COMMAND je příkaz
- REMOTE-PATH je cesta k souboru nebo adresáři na serveru
- LOCAL-PATH je cesta v lokální souborovém systému, povinné pro put

# Příkazy

**del** smaže soubor určený REMOTE-PATH na serveru

**get** zkopíruje soubor z REMOTE-PATH do aktuálního lokálního adresáře či na místo určené pomocí LOCAL-PATH je-li uvedeno

**put** zkopíruje soubor z LOCAL-PATH do adresáře REMOTE-PATH

**ls** vypíše obsah vzdáleného adresáře na standardní výstup (formát bude stejný jako výstup z příkazu ls)

**mkd** vytvoří adresář specifikovaný v REMOTE-PATH na serveru

**rmd** odstraní adresář specifikovaný V REMOTE-PATH ze serveru



# Chyby

**"Not a directory."** když REMOTE-PATH ukazuje na soubor, ale je použita operace ls, mkdir, rmdir

**"Directory not found."** když REMOTE-PATH neukazuje na žádný existující objekt při použití operace lst, mkd, rmd

**"Directory not empty."** když REMOTE-PATH ukazuje na adresář, který není prázdný a je použita operace rmdir

**"Already exists."** když REMOTE-PATH ukazuje na adresář/soubor, který již existuje a je použita operace mkd

**"Not a file."** když REMOTE-PATH ukazuje na soubor, ale je použita operace del, get, put

**"File not found."** když REMOTE-PATH neukazuje na žádný existující objekt při použití operace del, get, put

**"Unknown error."** pro ostatní chyby.

# Příklady

Vytvoření adresáře foo na serveru běžícím na lokálním počítači a portu 12345:

```
$ ftrest mkd http://localhost:12345/tonda/foo/bar
```

Nahrání souboru doc.pdf na serveru do adresáře foo:

```
$ ftrest put http://localhost:12345/tonda/foo/bar/doc.pdf ~/doc.pdf
```

Stažení souboru doc.pdf do lokálního adresáře:

```
$ ftrest get http://localhost:12345/tonda/foo/bar/doc.pdf
```

Odstranění souboru doc.pdf:

```
$ ftrest del http://localhost:12345/tonda/foo/bar/doc.pdf
```

Odstranění adresáře foo:

```
$ ftrest rmd http://localhost:12345/tonda/foo/bar
```

# Implementace

- Použijte jazyk C/C++ a knihovnu BSD socketů
- Není možné použít další knihovny pro komunikace
- Je možné použít knihovny pro:
  - parsování/generování JSON budete-li potřebovat
  - kompresních algoritmů
- Je nutné vytvořit skutečný Makefile
- Je nutné napsat dokumentaci (použijte Markdown formát) a strukturu podle man stránek.

# Testování

- Implementace bude testována na standardní instalaci distribuce CentOS7.
- Můžete si vytvořit vlastní instalaci nebo použít virtuální image:  
<http://qwe.fit.vutbr.cz/igreg/centos7.ova>
- Přihlašovací údaje:  
user/user4lab  
root/root4lab
- Co nebude přeložitelné/funkční v tomto prostředí může být hodnoceno 0 body !!!

# Hodnocení

- Hodnotit se bude:
  - Funkčnost a úplnost implementace
  - (Ne)kvalita implementace
  - Kvalita dokumentace
- Extra bodové hodnocení je možné získat za různá rozšíření:
  - autentizace uživatele heslem a použití Authorization string v hlavičce
  - použití kódování pro přenos dat jiné než identity
  - rozšíření existujících operací (ls příkaz s uvedením filtru)
  - další operace (vzdálený přesun/kopírování souboru)

???