

---

# tinyFTP

---

64  
commits



21  
days

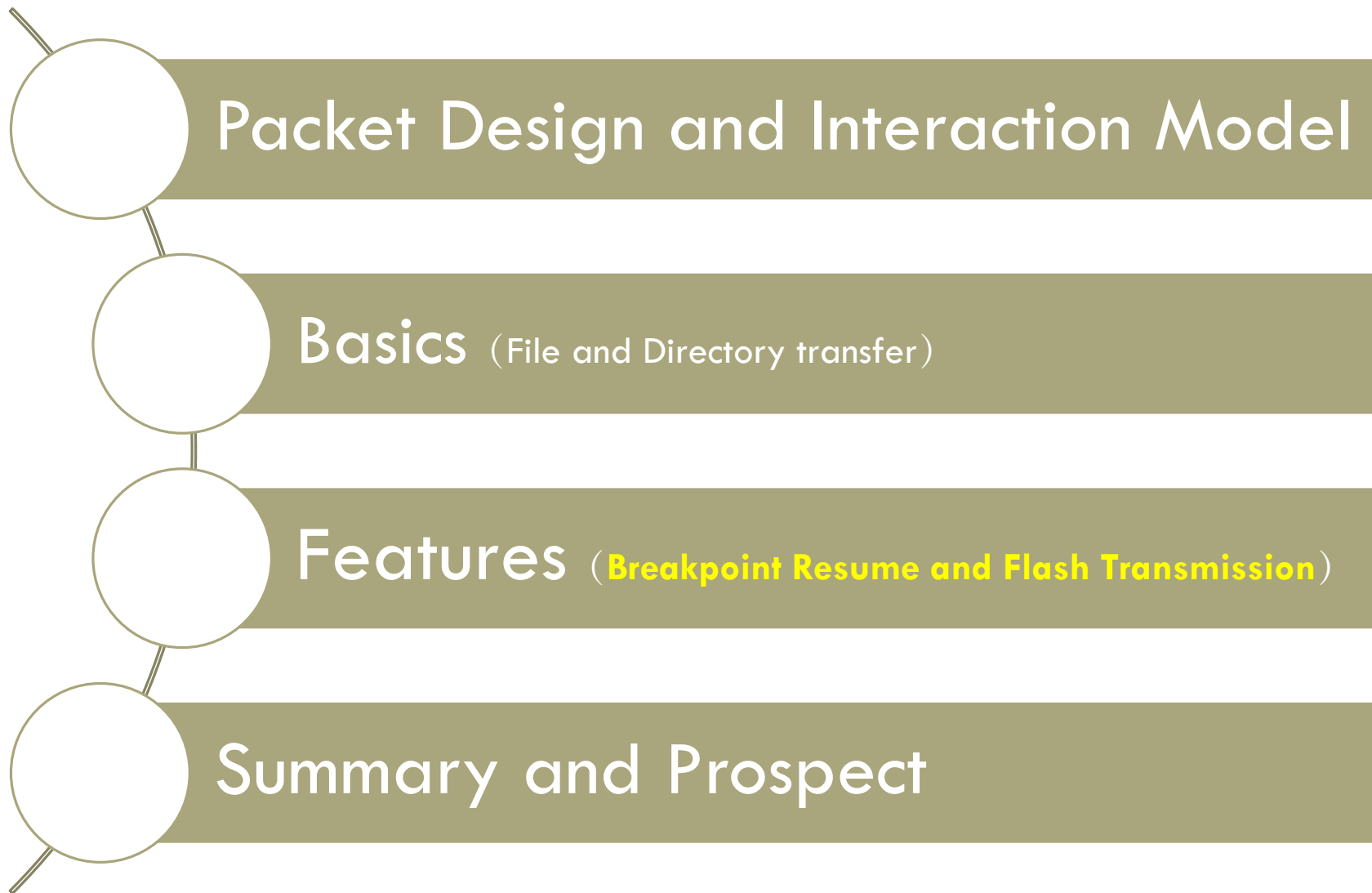
## FILE SERVER AND CLIENT

Wenchy

wenchy.zwz@gmail.com

# FOREWORD

- ✓ Single file (6G/2T) transfer
- ✓ Paged directory entries transfer (LS)
- ✓ Recursive or iterative transfer of directory
- ✓ User space isolation (CWD, sqlite3) , multithread concurrency
- ✓ Breakpoint resume (Slice + Filesize + MD5, sqlite3)
- ✓ Flash transmission (Filesize + MD5 + HardLink, sqlite3)



---

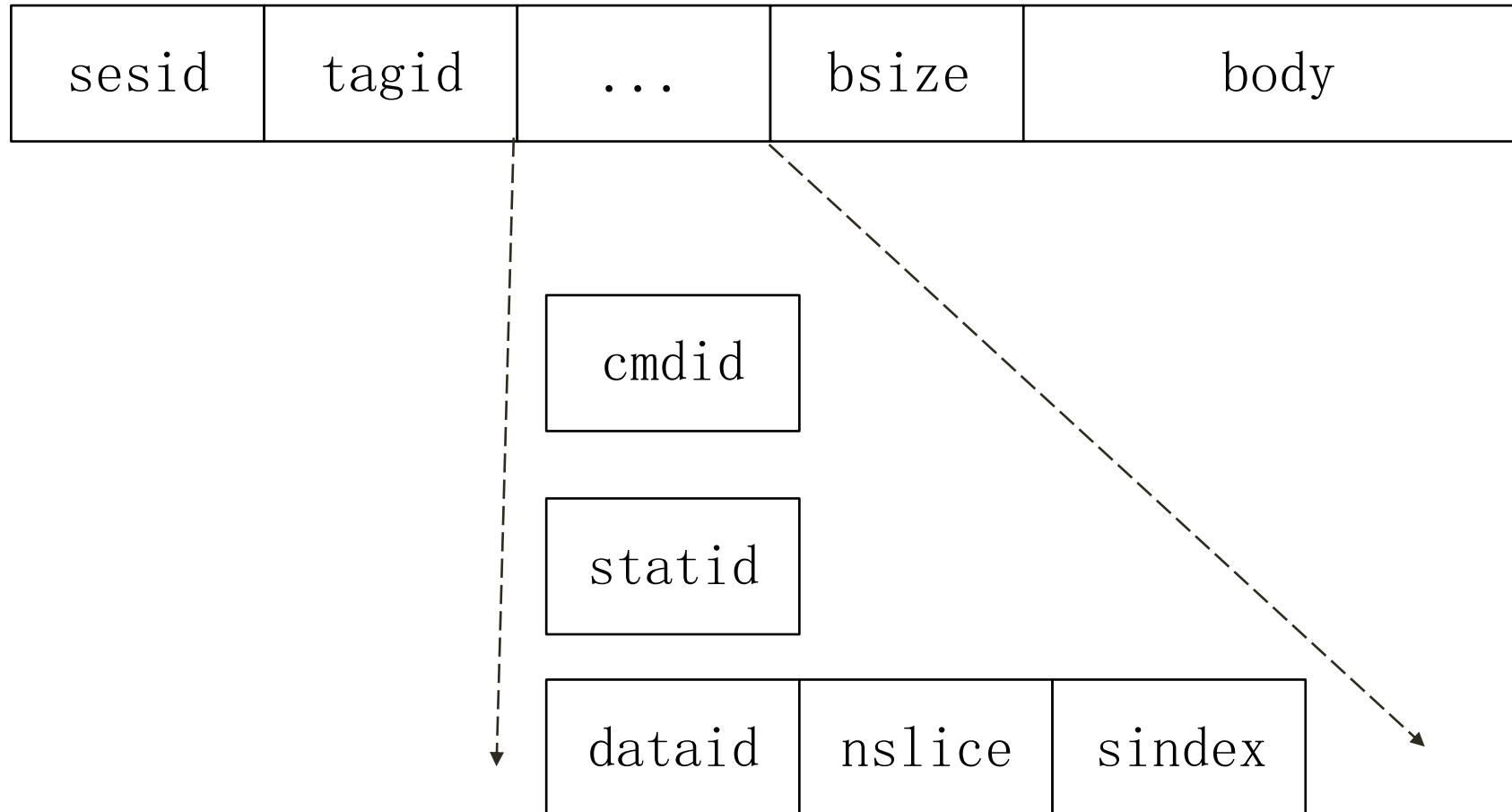
# tinyFTP

---

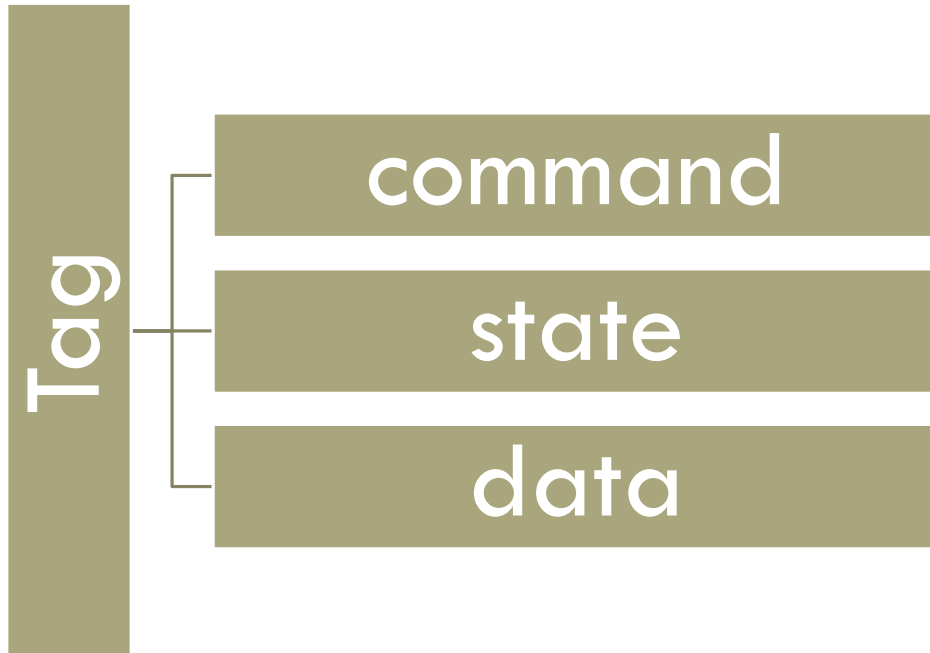
## PACKET DESIGN AND INTERACTION MODEL

Tag | Command  
Status  
Data

# PACKET FORMAT



# THREE FORMS OF TAG



PUT, GET, LS, RGET, RPUT, etc.

OK, ERR, TERM, EOT, EOF, etc.

FILE, LIST, TEXT, etc.

## Tag

```
typedef enum tagID
{
    // command
    TAG_CMD = 1,
    // status
    TAG_STAT,
    // data
    TAG_DATA
} TagID;
```

## Command

```
typedef enum cmdID
{
    USER = 1,
    PASS,
    USERADD,
    USERDEL,
    GET,
    PUT,
    LS,
    CD,
    RM,
    PWD,
    RGET,
    RPUT,
    ...
} CmdID;
```

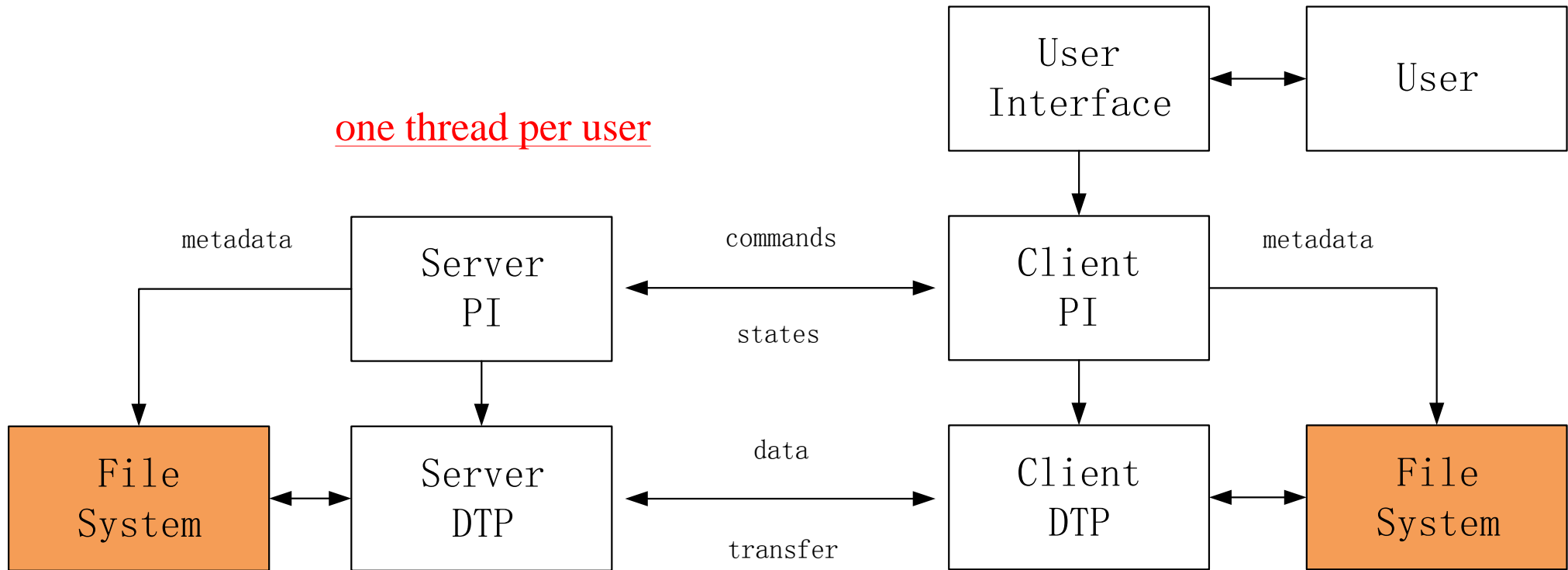
## State

```
typedef enum statID
{
    STAT_OK = 1,
    STAT_BPR, // breakpoint resume
    STAT_CFM, // confirm
    STAT_MD5, // md5sum
    STAT_PGS, // progress
    STAT_FAIL, // fail
    STAT_ERR, // error
    STAT_CTN, // continue
    STAT_TERM, // terminate
    STAT_SIZE, // size
    STAT_WAIT, // wait
    STAT_EOF, // end of file
    STAT_EOT // end of transfer
} StatID;
```

## Data

```
typedef enum dataID
{
    DATA_FILE = 1,
    DATA_TEXT,
    DATA_LIST,
    DATA_OTHER
} DataID;
```

# INTERACTION MODEL



**PI:** Protocol Interpreter

**DTP:** Data Transfer Process



---

# tinyFTP

---

## BASICS

- File
- Directory
- Directory Entries
- CWD

# USER SPACE ISOLATION

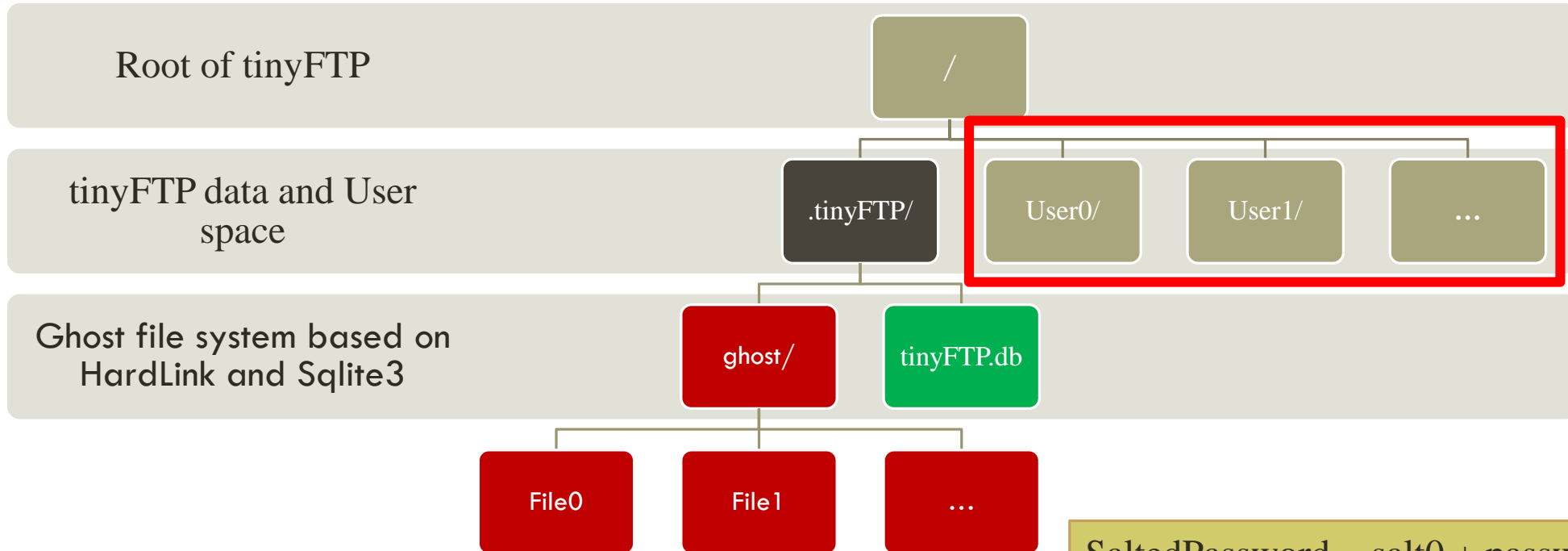
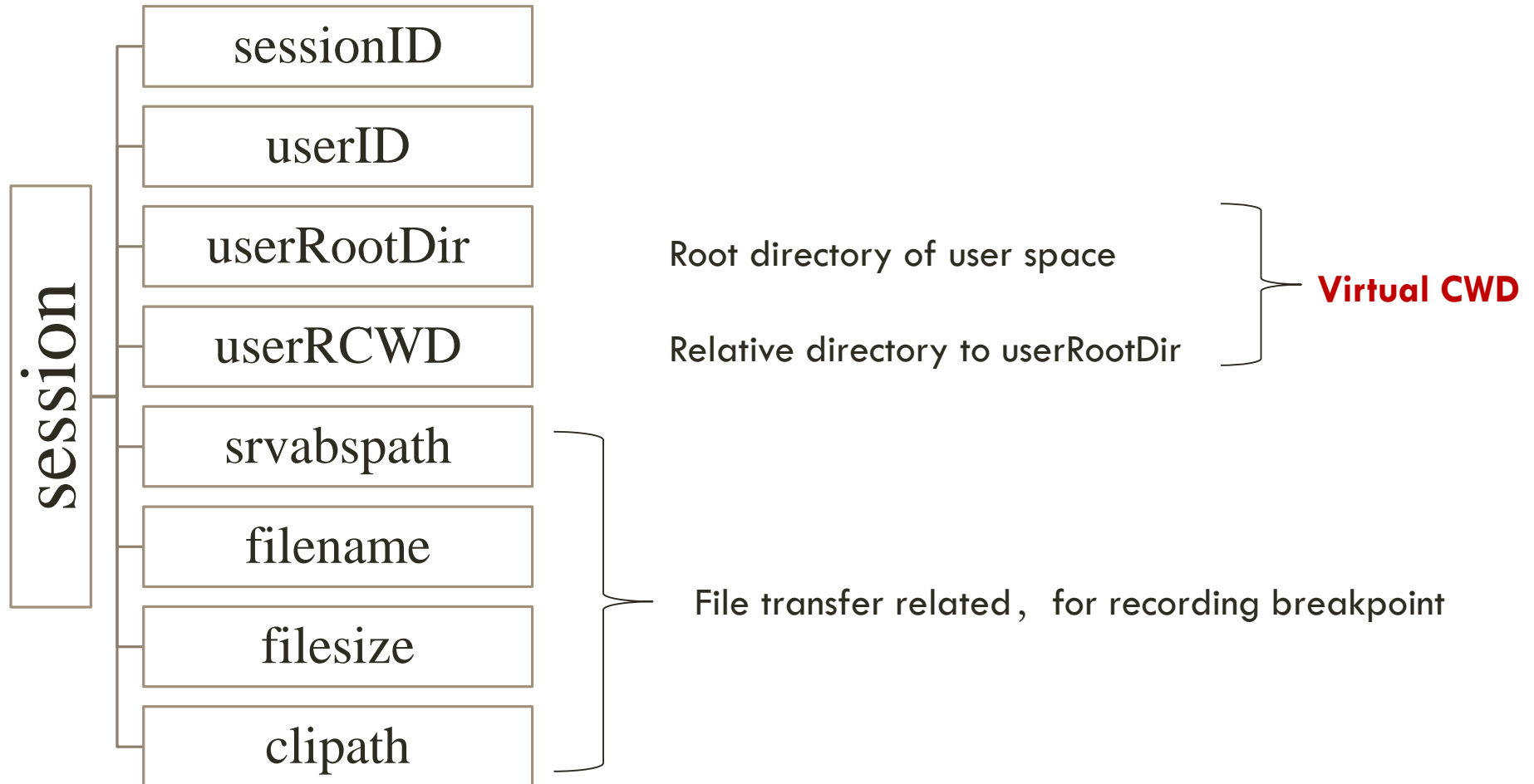


Table **USER**

ID	USERNAME	PASSWORD	RCWD	CREATE_AT	UPDATE_AT	STATE	...

$\text{SaltedPassword} = \text{salt0} + \text{password} + \text{salt1}$   
 $\text{USER.PASSWORD} = \text{MD5}(\text{SaltedPassword})$

# SESSION



Once one connection is closed, **Session** and **PrePacket** state require data persistence

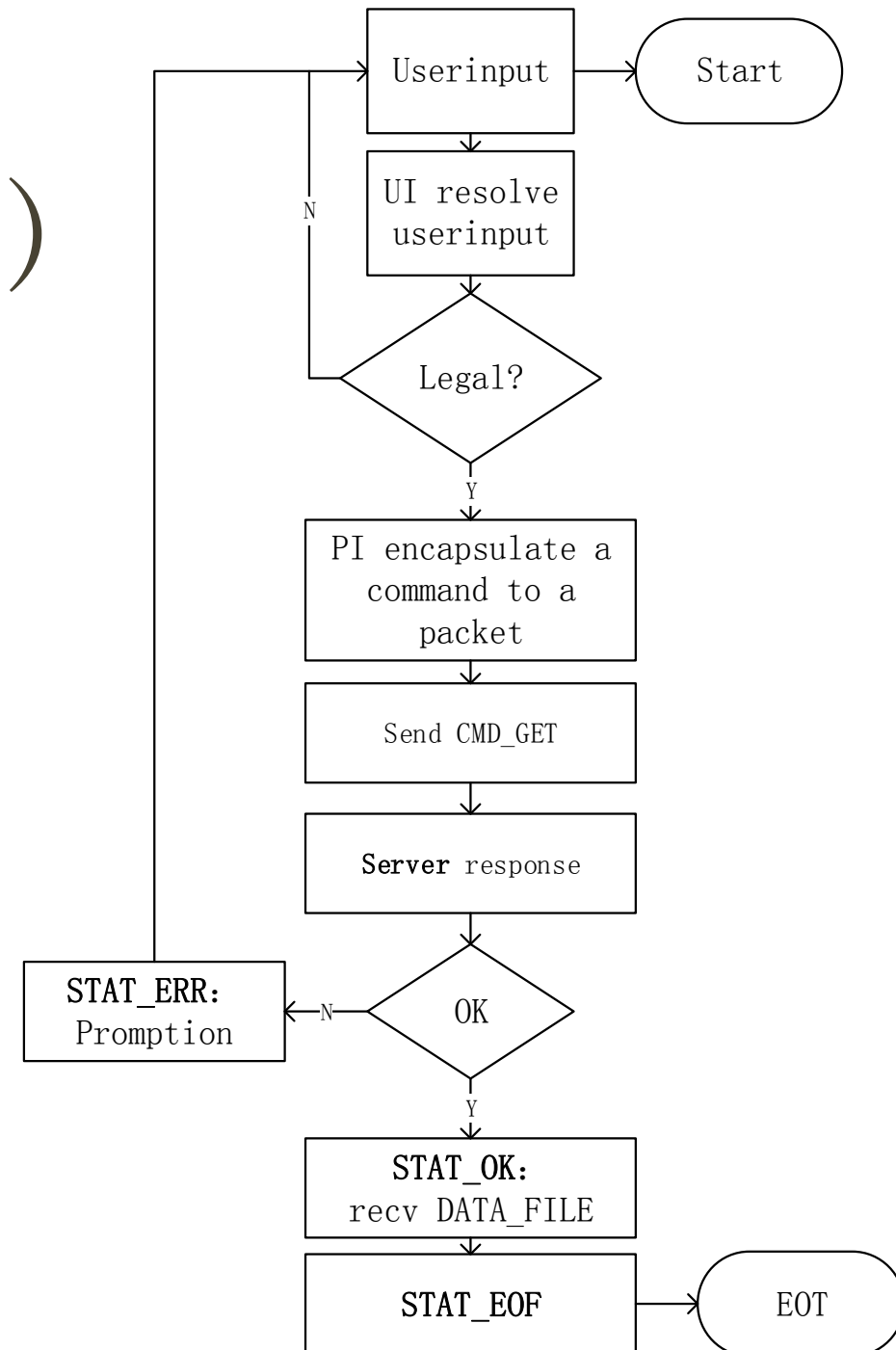
# SINGLE FILE TRANSFER (GET PUT)

Tested: **6G**

Theoretically: **2T**

NOTE:

1. File already exists ?
2. Different progress bar show of uploading and downloading
3. Disk space detection: Enough space to write?
4. File lock



# PAGED DIRECTORY ENTRIES TRANSFER (LS)

Thousands of directory entries, how to transfer elegantly ?

One page after one page, with user interaction (page down/up, quit)

# TRANSFER OF DIRECTORY (RGET RPUT)

**Recursive** or **Iterative** ?

✓ **Iterative**: More friendly, Clear and orderly;

✓ **Iterative**: First create a directory, then transfer files inside.

How to implement?

Reuse of PUT, GET, MKDIR commands

RPUT and RGET traverse directories and then issue corresponding commands orderly

# OTHERS

1. MKDIR      make a directory
2. RMDIR      remove a directory
3. RM          remove a file
4. CD          change CWD
5. PWD        print working directory
6. QUIT        close socket connection and exit
7. HELP        commands' usage
8. Corresponding local commands: LCD, LLS, etc.

---

# tinyFTP

---

## FEATURES

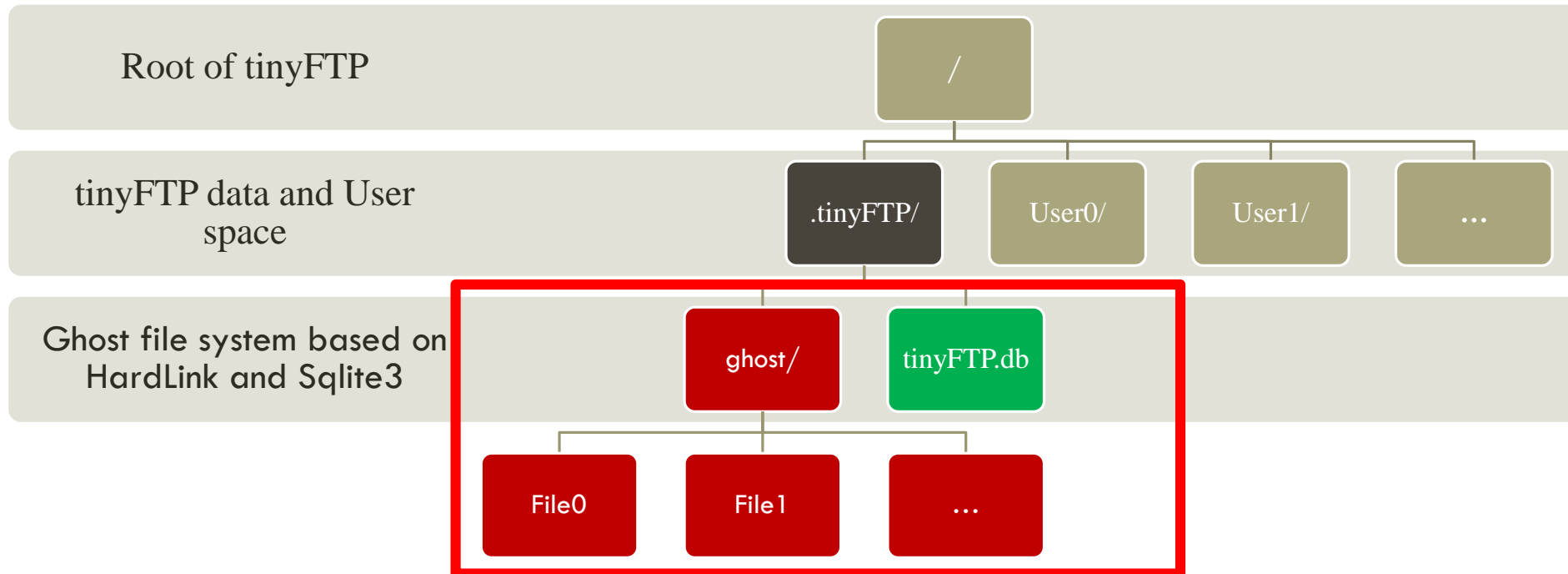
Breakpoint Resume  
Flash Transmission







# FLASH TRANSMISSION – GHOST FILE SYSTEM



Ghost file naming conventions: **timestamp\_md5sum\_filename**

Demo: `20150824101401_1dcec5a2febfaa144c882158eea77aa7_movie.mp4`

Target: guarantee uniqueness and convenient index

# FLASH TRANSMISSION – CONTROL FLOW



Table **FILE**

ID	ABSPATH	FILENAME	MD5SUM	INODE	SIZE	ACCESS	...

Once a file is uploaded, which is not recorded in database (MD5). It will be linked to a corresponding ghost file, then this file 's hard link count is 2. So, even this file is removed immediately, next transfer of it is flash transmission

---

# tinyFTP

---

SUMMARY AND PROSPECT

64

commits

21

days

# SUMMARY

## **Implemented :**

1. Basics: File and directory transfer, paged LS, user authentication, user space isolation
2. Features: breakpoint resume, flash transmission

## **Deficiencies :**

1. Unfinished breakpoint resume for GET command
2. Not robust program for network errors and exceptions
3. A command 's Transaction, state machine not clearly created
4. Not formalized server log (syslog)
5. Just finish hot data statistics, haven't prefetch hot data to RAM

# PROSPECT

- a) Server concurrency model : epoll
- b) Database: Mysql
- c) Hot data: prefetch into RAM memory
- d) Multithread download
- e) Deamon server

---

# tinyFTP

---

**Thanks**

Wenchy  
wenchy.zwz@gmail.com



# DEMONSTRATION

1. USER authentication, salting password
2. PUT progress bar(select), breakpoint resume, ghost file, hardlinks, flash transmission
3. RM remove a just uploaded file, then upload again (now is flash transmission)
4. GET hot data statistics
5. CD user space isolation, virtual CWD
6. RPUT
7. RGET