

SAKURAFUBUKI^{3rd} tutorial

Version 0.1-a

Maki Tsuchiya

maki@sfc.wide.ad.jp

Data Transfer Flow Chart in CLOSED MODE

[Sender]

[Receiver]

Member information advertising

Picture-1

Data information advertisement

Picture-2

Picture-3

Reporting Status

Data transmit

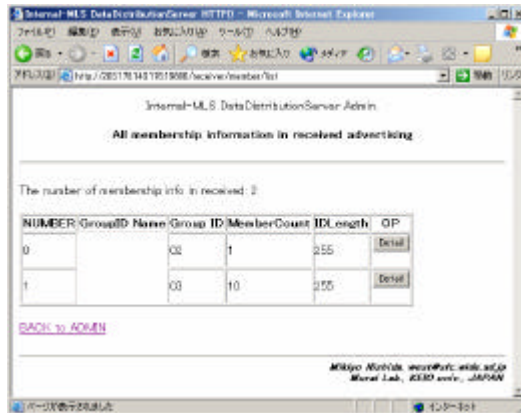
NACK(missing information)

Missing data retransmit

ACK(complete information)

Picture-4

Show list of received current member info (Picture-1)

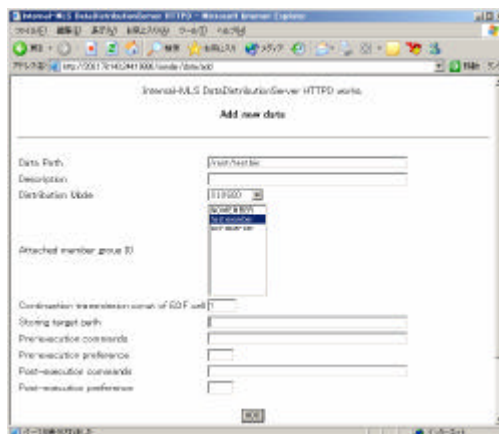


The screenshot shows a web browser window titled "Internal-MLS DataDistributionServer HTTPD - Microsoft Internet Explorer". The address bar shows the URL "http://203.178.143.101:8080/Receive/member/list". The page content includes the title "Internal-MLS DataDistributionServer Admin" and the heading "All membership information in received advertising". Below this, it states "The number of membership info in received: 2". A table displays the membership information:

NUMBER	GroupID	Name	Group ID	Member Count	ID Length	OP
0			03	1	255	Detail
1			03	10	255	Detail

Below the table, there is a link "BACK to ADMIN" and a footer with contact information: "Mitsuo Akita@kai.yamaguchi-u.ac.jp, Mitsuo Lab., KERO enviro., JAPAN".

Add New Data (Picture-2)



The screenshot shows a web browser window titled "Internal-MLS DataDistributionServer HTTPD - Microsoft Internet Explorer". The address bar shows the URL "http://203.178.143.101:8080/Receive/member/add". The page content includes the title "Internal-MLS DataDistributionServer HTTPD admin" and the heading "Add new data". The form contains the following fields:

- Dist Path:
- Description:
- Distribution Mode:
- Attached member zone (2):
- Condition transmission sort of GDF and:
- Storing target both:
- Pre-execution commands:
- Pre-execution preference:
- Post-execution commands:
- Post-execution preference:

At the bottom of the form, there is a "ADD" button.

Check receiving data by logfile (Picture-4)

- `tail -f /var/log/imlsdds.xxxxx.log`
 - INFO: [L255] :CELLSTORAGE: InitStorage Successful / status = 1
 - INFO: [L255] :RECVTRANSMIT: InitStorage Successful [/test.bin]
 - INFO: [L2] :RECVTRANSMIT: Not Set preProcCmd
 - INFO: [L255] :RECVTRANSMIT: initial multicast TTL:4
 - INFO: [L255] :[v4] ReceiverDataTransmitterProc() 21000
 - INFO: [L255] :RECVTRANSMIT: transmit status = 0
 - INFO: [L255] :SINP_CLIENT: Exec _sinpClientProc4
 - INFO: [L255] :SINP_CLIENT: Delete INFO
 - INFO: [L2] :CELLSTORAGE: **Data Complete '/test.bin'**

Lesson 1

- Send sample file :demonstration -

Preparation before receiving Files

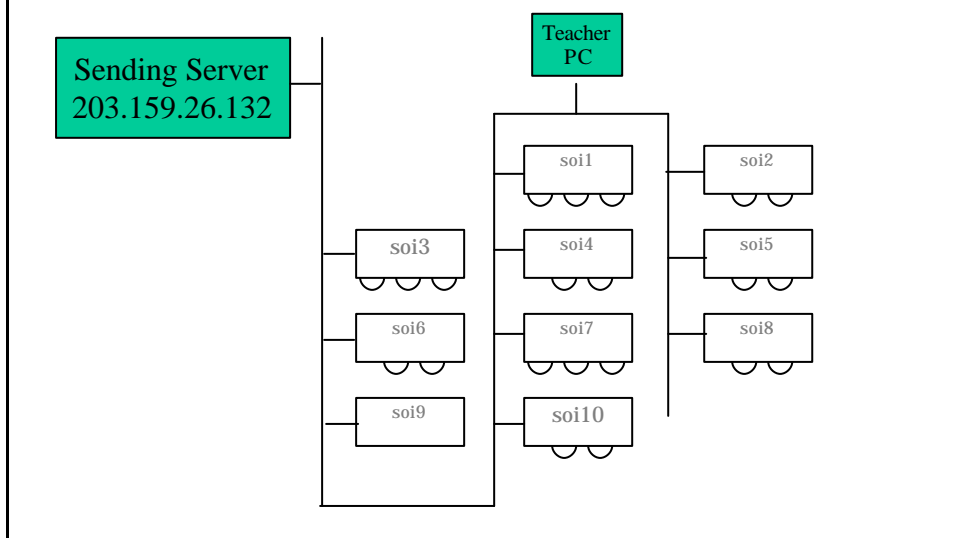
1. Edit recv.ini for your mcdd program
2. Restart your mcdd program
3. Verify mcdd program is working
 - Check the process
 - Check the Group information

Set up your host information to configure file

Change follow file

- /usr/local/implsdds/recv.ini
 - ReceiverMulticast IF4
 - Set ip address of each machine
 - ReceiverIdentity
 - Set Identity for this software(see next page)
 - TraceServerName
 - Set ip address of sender
 - LogMode
 - Set log mode to “0”

Today's environment



Restart mcdd program

Restart mcdd daemon

```
# killall mcdd
```

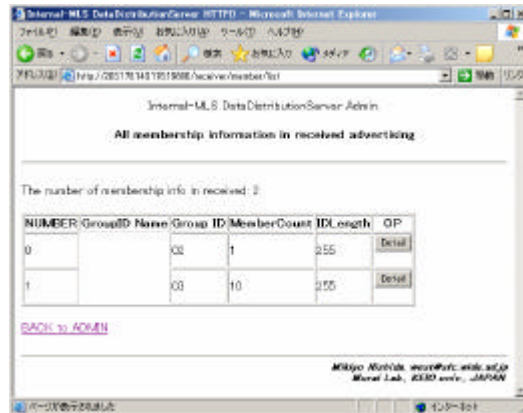
```
# /usr/local/imlsdds/mcdd -f /usr/local/imlsdds/recv.ini
```

Check program process

```
#ps -ax
```

```
5647 tty1 S 0:00 ./mcdd -f recv.ini
5648 tty1 S 0:00 ./mcdd -f recv.ini
5649 tty1 S 0:00 ./mcdd -f recv.ini
5650 tty1 S 0:00 ./mcdd -f recv.ini
5651 tty1 S 0:00 ./mcdd -f recv.ini
5652 tty1 S 0:00 ./mcdd -f recv.ini
5656 tty1 S 0:00 ./mcdd -f recv.ini
5657 tty1 S 0:00 ./mcdd -f recv.ini
```

Check Member info by Show list of member information



Receiving Data (1)

Sample Files :

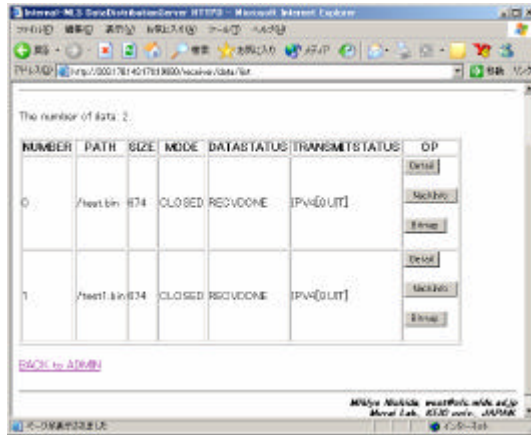
/soi/soi/class/20020032/materials/test.ppt

/soi/soi/picture/icon_pencil.gif

/soi/soi/picture/icon_tv.GIF

After receiving data,
check the file on your SOI Server.

“Show list of received current data”



Lesson 2

- Send soi archive file to your computers -

Prepare a “Post Process” script (one time only)

- Necessary to customize the SOI archive contents for your SOI server
- Install post process script

```
#mkdir /soi/soi/util
```

```
#cd /mnt/cdrom
```

```
#cp customize_lecture.pl /soi/soi/util
```

```
#chmod 775 /soi/soi/util/customize_lecture.pl
```

Receive Archived Contents

Sample Files:

```
/soi/soi/class/20020032/video/01/01.rm
```

```
/soi/soi/class/20020032/slides/01/01.tar
```

- After receiving data,
 - check the file on your SOI Server.
 - 01.tar is automatically extracted and customized.

Verify SOI Contents delivered

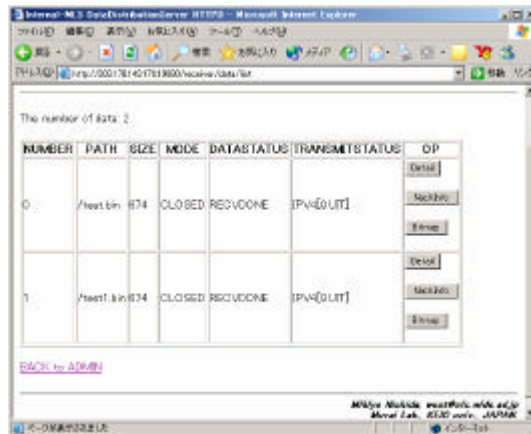
- See the SOI contents on your browser;
<http://YOURSERVER/class/20020032/slides/01/>
- See the video by clicking “start video”

Now Your Turn !

Thank you very much !

Appendix :
- Show list of received current data –

Show list of received current data (Picture-3)



The number of data: 2

NUMBER	PATH	SIZE	MODE	DATABSTATUS	TRANSMITSSTATUS	OP
0	/test bin	874	CLOSED	RECVDONE	[Pv]{0} [U]	Delete Refresh Erase
1	/test bin	874	CLOSED	RECVDONE	[Pv]{0} [U]	Delete Refresh Erase

[BACK to AT&M](#)

MODE

- **CLOSED** :Only specified member can receive data, and receiver can request missing data and should report receive status to sender.
- **OPENED** :Everyone can receive data, and only members of the group can request missing data and should report receive status to sender.
- **UNLIMITED** :One-way transmission. Everyone can receive data, but nobody can request missing data.

STATUS

- INIT :complete initialization
- LOADED :loaded data information. no get sent and received thread
- RECEIVING :data receiving(working receiving thread)
- RECVDONE :complete data receiving (exit receiving thread)
- RECVFAILURE :failed receiving data
- RECVREPAIR : repairing data

TRANSMIT STATUS

- INIT :complete initialization
- RECV : data receiving
- RECVSTOP :stop receiving
- REPAIRING :repairing data
- TURNEND :complete 1 term of sent and received data
- QUIT :complete thread