

```

#!/bin/bash
#module load dm17/tools #load this module to access dedicated commands for data-manager
such as: parallel, numfmt, etc.
export candidate=
export LOCK=
export MAIN_PATH=/kbin-irsnb/digit03ltp
#export ORIGINALDATA_PATH=/kbin-irsnb/data
export ORIGINALDATA_PATH=/media/jmherpers/BE-RBINS-BACKUP-028/LTP_transfer/data
export OUT_PYTHON_PATH=/media/jmherpers/BE-RBINS-BACKUP-028/LTP_transfer/out_python
export DATASET_FILE=dataset.param
export DATASET_PREZIP_FILE=dataset_prezip.param
export LOCAL_CSV_FILE=$MAIN_PATH/list_local.csv
export REMOTE_CSV_FILE=$MAIN_PATH/list_remote.csv
export XML_PATH=$MAIN_PATH/xml
export DATA_PATH=$MAIN_PATH/data
export SIP_PATH=$DATA_PATH/sip
export XML_PATH=$DATA_PATH/xml
export UPLOAD_PATH=$DATA_PATH/upload
export INGESTED_PATH=$DATA_PATH/ingested
export DOWNLOAD_PATH=$DATA_PATH/download
export CSVPLONE_PATH=$DATA_PATH/csv_plone
mkdir -pv $XML_PATH/ $SIP_PATH/ $XML_PATH/ $UPLOAD_PATH/ $INGESTED_PATH/ $DOWNLOAD_PATH/
$CSVPLONE_PATH/
chown -c jmherpers $DOWNLOAD_PATH/
export DATA_STR=kbin-irsnb_data_NAS_
export PATH_TO_DATA_IN_MNT=/media/jmherpers/BE-RBINS-BACKUP-004/ltp_storage/
export PATH_TO_DATA_IN_MNT=/mnt/NAS/XRE/
export PATH_TO_DATA_IN_MNT=/media/jmherpers/BE-RBINS-BACKUP-004/XRE-LTP/
export PATH_TO_DATA_IN_MNT=/media/jmherpers/Digital_Collection_Disk_2_Clone/ltp_storage/
export PATH_TO_DATA_IN_MNT=/media/jmherpers/Back-up_Stacking_Entomology/ltp_storage/
export PATH_TO_DATA_IN_MNT=/mnt/NAS/rx/
export PATH_TO_DATA_IN_MNT=/mnt/ltp_storage/OK_for_LTP
#upload
#export UHOST=ltp-ingest.belspo.be
#export UURL=ftp://$UHOST:21/
export UHOST=ltp-acc.belspo.be
export UHOSTDIR=/serviceIrsnb-Kbin/VERS/CollectionIrsnb-Kbin/
export UURL=sftp://$UHOST:21/$UHOSTDIR
export ULOGIN= #see ~/.netrc
#download
export DHOST=ltp.belspo.be
export DURL=https://$DHOST/mediahaven-rest-api/resources
export DLOGIN= #to fill
export DPASSWORD= #to fill
#-----
create_lock()
{
    LOCK="${1:-$0.lock}"
    test -n "$LOCK" || return
    if mkdir -p "$LOCK" 2>/dev/null #see http://mywiki.woledge.org/BashFAQ/045
    then
        echo "$(date +%b %e %H:%M) lock $LOCK created" >&2
    else
        echo "lock $LOCK already exists" >&2
        exit 13
    fi
    #lock removing automatical called here
    trap 'remove_lock; trap - EXIT; exit' INT TERM EXIT #see http://www.davidpashley.com/articles/writing-robust-shell-scripts/
}
export -f create_lock
#-----
remove_lock()
{
    test -n "$LOCK" || return
    rm -rf "$LOCK"
}

```

```

    echo "$(date +%b %e %H:%M) lock $LOCK removed" >&2
    LOCK=
}
export -f remove_lock
#-----
encode_sip()
{
    sed -e 's|^/|'| -e 's|//|/|g' -e 's|/|_|g' <<< "$1"
}
export -f encode_sip
#-----
second2human()
{
    local seconds=${1:-0}
    seconds=${seconds%.*}
    [[ $seconds =~ ^[0-9]+$ ]] || return
    if [ $seconds -lt 60 ]
    then
        printf "%ds" $seconds
    elif [ $seconds -lt 3600 ]
    then
        printf "%dm%02ds" $((seconds/60)) $((seconds%60))
    elif [ $seconds -lt 86400 ]
    then
        printf "%dh%02dm" $((seconds/3600)) $((seconds/60%60))
    elif [ $seconds -lt 604800 ]
    then
        printf "%dd%02dh" $((seconds/86400)) $((seconds/3600%24))
    elif [ $seconds -lt 31536000 ]
    then
        printf "%dw%02dw" $((seconds/604800)) $((seconds/86400%7))
    else
        printf "%dy%02dw" $((seconds/3153600)) $((seconds/604800/7%52))
    fi
}
export -f second2human
#-----
byte2human()
{
    numfmt --to=iec ${1:-0}
}
export -f byte2human
#-----
human2byte()
{
    numfmt --from=iec ${1%B}
}
export -f human2byte
#-----
get_speed()
{
    local bytes=${1:-0} seconds=${2:-1}
    seconds=${seconds%.*}
    test "$seconds" -lt "1" && seconds=1
    printf "%s in %s: %'%.0fGB/d %'%.0fGB/h %'%.0fMB/s %'%.0fbps\n" \
        $(byte2human $bytes) \
        $(second2human $seconds) \
        $(bc -l <<< "( $bytes/10^9)/($seconds/86400)" ) \
        $(bc -l <<< "( $bytes/10^9)/($seconds/3600)" ) \
        $(bc -l <<< "( $bytes/10^6)/ $seconds" ) \
        $(bc -l <<< "(8*$bytes )/ $seconds" )
}
export -f get_speed
#-----
md5()
{

```

```
    local md5=$(md5sum "$1")
    echo ${md5:0:32}
}
export -f md5
#-----
get_zip_on_ltp_ingest()
{
    lftp -u $ULOGIN $UHOST -e "set ftp:ssl-allow off; set ftp:passive-mode off; cache
flush; nlist *zip; quit" \
    | sort
}
export -f get_zip_on_ltp_ingest
```