

SureStore E Disk Array XP256 / XP512 Business Copy XP and/or Continuous Access XP in an HP e3000 Environment

This paper describes the usage and procedures used to successfully implement Business Copy and/or Continuous Access in an HP e3000 environment.

White Paper by
Walter McCullough
November 18, 2000

Product Description Business Copy XP

Business Copy is an XP256/XP512 feature that allows for copy on demand of selected disk volumes. These volumes should be grouped as User Volumes, an MPE/iX term. These copied disks can then be split off from their masters and presented to another HP e3000 machine for offloading operations such as backup or application testing.

Business Copy volumes are created by the HP storage specialist through the SVP and managed from the internal XP256/XP512 PC or from a PC configured as the Remote Control PC or through an instance of the RAID Manager XP executing under MPE/iX.

Continuous Access XP

Continuous Access is the feature that allows the XP256 to continuously maintain a copy of a selected number of disk volumes on another XP256/XP512 disk cabinet, located locally or remotely.

This feature is somewhat similar to Business Copy in that offloading backup operations or application testing can be performed. The added value is that the user can use this feature to maintain a complete working set of data that could be used on the remote computer in the event of a catastrophic site outage at the primary location.

Environment

Business Copy and Continuous Access is supported on MPE/iX release 5.5 with PPT07 and patch MPEKXL9 and MPEKXT8. It is also supported on MPE/iX release 6.0 with patch MPEKXL9 and MPEKXT8. RAID Manager XP is supported on 6.5 post express 2.

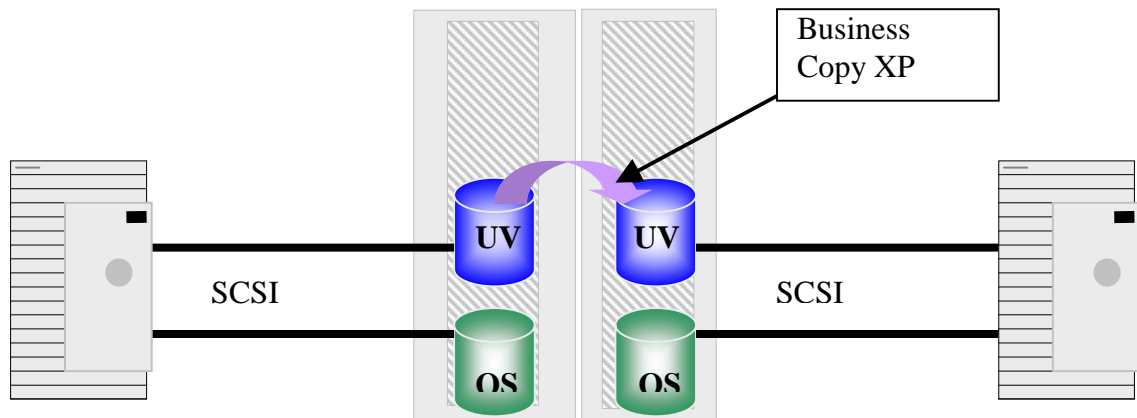
For more information on patches and versions, please contact your support engineer.

Basic Operation

Business Copy and Continuous Access have virtually the same procedures for setup. System A will “own” data located on a primary user volume set. A running copy of this data will be maintained on a Business Copy of the volume set.

At a customer defined time, the user will VSCLOSE the primary user volume set and then split the primary user volume set from the Business Copy volume set through the RAID manager.

After the split has been performed the user can then VSOPEN the primary user volume set onto System A and continue operations. System B can now VSOPEN the split off Business Copy volume set and proceed with its operations.



System A Procedures

Setup System A

The first time setup requires is as follows:

1. Create the LUNs and paths you will be using on the XP256 and then using Sysgen or IOCONFIG create the LDEVs associated with the paths and LUNs. Make sure to use HPDARRAY as the ID type. We will use the LDEV 60 through 62 for our examples.

Test that MPE/iX recognizes the new LDEVs by either rebooting and running ODE Mapper or if you used IOCONFIG to issue the commands, use DSTAT ALL.

```
(MPE/iX Prompt):dstat all
  LDEV-TYPE      STATUS      VOLUME (VOLUME SET - GEN)
-----
60-OPEN-3       UNKNOWN
61-OPEN-3       UNKNOWN
62-OPEN-3       UNKNOWN
```

(MPE/iX Prompt):

2. Now create the User Volume Set that will be used to offload processing. The volume set name for this example will be **PRODUCTION_DATA**. Use the **VOLUTIL** utility program to create the user volume set named **PRODUCTION_DATA**. Once the volume set is created, **VSCLOSE** and then **VSOPEN** the volume set before proceeding onward.
3. Issue the **DSTAT ALL** command and the following should appear:

```
(MPE/iX Prompt):dstat all
  LDEV-TYPE      STATUS      VOLUME (VOLUME SET - GEN)
-----
60-OPEN-3       MASTER      MEMBER1      ( PRODUCTION_DATA-0 )
61-OPEN-3       MEMBER      MEMBER2      ( PRODUCTION_DATA-0 )
62-OPEN-3       MEMBER      MEMBER3      ( PRODUCTION_DATA-0 )
```

4. Next, create the directory and accounting structure on the system that will use the User Volume. Be sure to remember to use the proper syntax and parameters **ONVS** and **HOMEVS** when creating the groups. You are now ready to restore or create the files that will reside on the user volume set **PRODUCTION_DATA**.

From the Remote Control PC, configure the LUNs and paths for the Business Copy volumes and start the copy operation. This takes about 1 Gbytes per minute depending on your configuration. Once this operation has completed, you may proceed.

At this point you should have a system configured with a user volume set named **PRODUCTION_DATA** and a directory and accounting structure that is accessible from System B. You should also have a copy of this volume set maintained by Business Copy. To access the files and directories when this volume set is “moved” over to System B, you need to have the identical directory and accounting structure on System B that is already on System A.

BULDACCT

To access those files on the Business Copy volumes you must first create the accounts and groups on System B, that reference the directory structure located on the user volume. Re-issue the NEWACCT and NEWGROUP commands on System B or using BULDACCT to create a job stream that will create the directory for you.

1. Execute the buldacct utility program and for this example three accounts exist on the PRODUCTION_DATA volume set. SALESDB, ORDERDB and STOCKDB.

```
:RUN BULDACCT.PUB.SYS;INFO="SALESDB,ORDERDB,STOCKDB%VSACCT=PRODUCTION_DATA"
```

This will create 2 files BULDJOB1 and BULDJOB2. These files create the directory structure and re-create the UDC linkages. Store these files off and restore them to System B. (*DSCOPY can also be used*)

System B Procedures

5. Either re-issue the NEWACCT commands that pertain to creating the accounts on the system volume set and only those NEWGROUP commands that you used the parameter HOMEVS or restore the BULDJOB1 and BULDJOB2 files onto System B and stream those jobs.
6. From Sysgen or IOCONFIG issue the commands to configure the LDEVs of the Business Copy volumes. Be sure that the copy has completed.

If you used Sysgen to configure the volumes you will need to reboot the system. If IOCONFIG was used, then it will try to mount the volumes as you add the LDEVs.

Once the volumes are configured and the directory structures are intact, issue the dstat all command.

```
(MPE/iX Prompt):dstat all
  LDEV-TYPE   STATUS      VOLUME (VOLUME SET - GEN)
-----
60-OPEN-3    LONER-RO    MEMBER1      ( PRODUCTION_DATA-0 )
61-OPEN-3    LONER       MEMBER2      ( PRODUCTION_DATA-0 )
62-OPEN-3    LONER       MEMBER3      ( PRODUCTION_DATA-0 )
```

The master volume should say LONER-RO, which means that it is closed and access is defined as Read-Only. MPE/iX will not let you mount a Read-Only disk because it is not supported on MPE/iX. The Read-Only attribute is set by MPE/iX when it encounters an active copy mode from an XP256.

Business Copy Procedure

On System A VSCLOSE PRODUCTION_DATA;NOW This will quies I/Os to the volume set and flush all files and data structures resident to that volume set to disk.

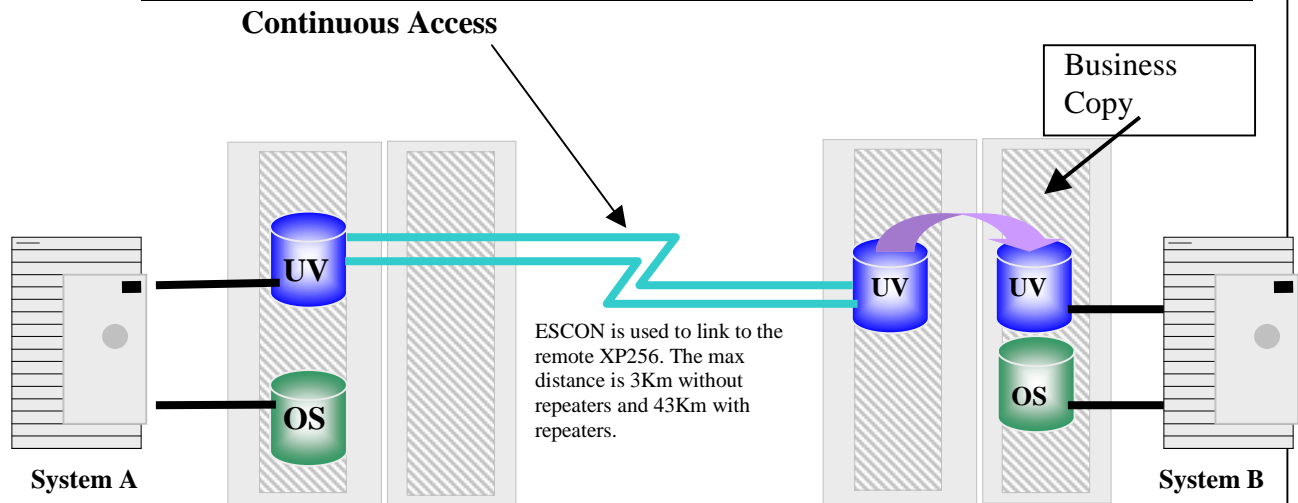
Issue the pairsplit command to the RAID instance and wait for its completion.

(After the split you may VSOPEN the PRODUCTION_DATA volume set on System A and continue operations)

On System B

After the primary volumes have been split from the Business Copy you can issue the VSOPEN PRODUCTION_DATA command on System B and start operations.

When operations are completed, VSCLOSE PRODUCTION_DATA;NOW and from the Remote Control PC re-establish the primary and Business Copy mode. There is no need to log off users from System A to re-establish Business Copy.



The diagram above describes the use of both Continuous Access and Business Copy to provide both disaster recovery and off loading of work done by the remote data center.

The goal is to always maintain a link to the primary volume set through Continuous Access and use a separate set of disks that are maintained by Business Copy for planned events, like work offloading.

The process is almost the same as that of Business Copy. First quies I/O and log users off the PRODUCTION_DATA volume set by issuing the VSCLOSE;NOW command on System A. Then when all data has been moved to the remote XP256 only then do you split the Business Copy volumes from the Continuous Access volumes on the remote XP256.

Only when the Business Copy volumes are split on the remote XP256 can System B VSOPEN its copy of the PRODUCTION_DATA volume set.

The following text is a script that was contributed that will help in splitting and resyncing volume group pairs. With a little tweeking this script should work very well on your HP e3000.

```

#!/usr/bin/ksh
# UNISRC_ID @(#)pairs_control: $Revision: 1.0 $ $Date: 99/10/07 16:38:00 $
# MWR991007
# Sample script to illustrate how to manipulate Hewlett-Packard
# Business Copy volume pairs via command line

# Alter PATH if needed
PATH=$PATH:/HORCM/usr/bin/

# Set Environment for HORCM instance
export HORCMINST=0
# Set Env to select Business Copy operations vs CA operations
export HORCC_MRCF=1

# Set timeout variables in seconds
let MASTERTIMEOUT=10000
let SHORTTIMEOUT=600

# Set return or completion codes
RC_COMPLETE=0
RC_SIMPLEX=1
RC_PAIRED=3
RC_SPLIT=4

# Minimum rev of RAID Mgr for differential restore
DIFF_RESTORE_REV="010203"
# Minimum rev of microcode for differential restore
DIFF_RESTORE_MICROCODE="524251"

#####
# Functions Declarations #
#####

check_raidmgr ()
# Check if RAIDMgr exists and set revision
{
whence raidqry > /dev/null 2>&1
if [ "$?" -ne 0 ]
then
    echo "RAID Manager Software not detected on system!"
    exit 1
fi

# Get Rev of RAIDMgr
RMGR_REV=`raidqry -? 2>&1 |grep "^Ver&Rev"|awk '{print $2}'|tr -d ". "`
if [ $RMGR_REV -ge $DIFF_RESTORE_REV ]
then
    DIFF_RESTORE_RMGR=1
else
    DIFF_RESTORE_RMGR=0
fi

# Verify HORCM instance connection
raidqry -l > /dev/null
if [ "$?" -ne 0 ]
then
    echo "Cannot make connection to horcm instance!"
    exit 1
fi

# Get array microcode version
MICROCODE_REV=`raidqry -l|tail +2|awk '{print $7}'|cut -c 1-8|tr -d "- "`
if [ $MICROCODE_REV -ge $DIFF_RESTORE_MICROCODE ]
then
    DIFF_RESTORE_MICROCODE=1
else
    DIFF_RESTORE_MICROCODE=0
fi

if [ $DIFF_RESTORE_MICROCODE -eq 1 -a $DIFF_RESTORE_RMGR -eq 1 ]
then
    CAN_DIFF_RESTORE=1
else
    CAN_DIFF_RESTORE=0
fi

#echo "DIFF_RESTORE_MICROCODE=${DIFF_RESTORE_MICROCODE} \012
#DIFF_RESTORE_RMGR=${DIFF_RESTORE_RMGR} \012

```

```
#CAN_DIFF_RESTORE=${CAN_DIFF_RESTORE}"
```

```
}
```

```
create_pairs ()
```

```
{
```

```
# Create Business Copy pairs
```

```
date
```

```
echo "Creating Business Copy pairs for group $GROUP"
```

```
# Check status of volumes to be paired. Must be in SIMPLEX state.
```

```
pairevtwait -g $GROUP $MEMBER -nowait -nomsg
```

```
RETURN="$?"
```

```
if [ $RETURN -ne $RC_SIMPLEX ]
```

```
then
```

```
    echo "Error: Pairs $GROUP are not in SIMPLEX state. Cannot initiate copy."
```

```
    pairevtwait -g $GROUP $MEMBER -nowait
```

```
    echo "Return code was: ${RETURN}."
```

```
    exit 3
```

```
fi
```

```
# Create pairs with maximum track rate (15)
```

```
paircreate -g $GROUP $MEMBER -vl -c 15
```

```
# Monitor Status
```

```
echo "Waiting for pairs to transition to PAIRED state."
```

```
let COUNTER=0
```

```
RESULT=-1
```

```
while [ "$RESULT" -ne "$RC_COMPLETE" ]
```

```
do
```

```
    pairevtwait -g $GROUP $MEMBER -s pair -nomsg -t $SHORTTIMEOUT
```

```
    RESULT=$?
```

```
    let COUNTER=$COUNTER+$SHORTTIMEOUT
```

```
    if [ "$COUNTER" -ge "$MASTERTIMEOUT" ]
```

```
    then
```

```
        echo "Error: Timeout while waiting for pair $GROUP to transition to PAIRED state"
```

```
        pairevtwait -g $GROUP $MEMBER -nowait
```

```
        echo "Return code was: ${RESULT}."
```

```
        exit 4
```

```
    fi
```

```
done
```

```
echo "Volumes in group $GROUP synced. Copy completed"
```

```
date
```

```
}
```

```
split_pairs ()
```

```
{
```

```
# Split Business Copy pairs
```

```
date
```

```
echo "Splitting paired volumes for group $GROUP. "
```

```
# Checking for required state of PAIR
```

```
pairevtwait -g $GROUP $MEMBER -nowait -nomsg
```

```
RETURN="$?"
```

```
if [ $RETURN -ne $RC_PAIRED ]
```

```
then
```

```
    echo "Error: Pairs $GROUP are not synchronized."
```

```
    pairevtwait -g $GROUP $MEMBER -nowait
```

```
    exit 5
```

```
fi
```

```
# Split pairs
```

```
pairsplit -g $GROUP $MEMBER
```

```
echo "Waiting for split to complete."
```

```
let COUNTER=0
```

```
RESULT=-1
```

```
while [ "$RESULT" -ne "$RC_COMPLETE" ]
```

```
do
```

```
    pairevtwait -g $GROUP $MEMBER -s psus -nomsg -t $SHORTTIMEOUT
```

```
    RESULT=$?
```

```
    let COUNTER=$COUNTER+$SHORTTIMEOUT
```

```
    if [ "$COUNTER" -ge "$MASTERTIMEOUT" ]
```

```
    then
```

```
        echo "Error: Timeout while waiting for pair $GROUP to transition to PAIRED state"
```

```
        pairevtwait -g $GROUP $MEMBER -nowait
```

```
        echo "Return code was: ${RESULT}."
```

```
        exit 6
```

```

        fi
done
echo
echo "Split has completed. We may now resume processing on source volumes."
echo
}

delete_pairs ()
{
# Split Business Copy pairs
date
echo "Deleting pairing of volumes for group $GROUP. "

# Checking for required state of SPLIT (guarentees syncing)
pairevtwait -g $GROUP $MEMBER -nowait -nomsg
RETURN="$?"
if [ $RETURN -ne $RC_SPLIT ]
then
    echo "Error: Pairs $GROUP are not in PSUS state."
    pairevtwait -g $GROUP $MEMBER -nowait
    exit 7
fi

# Split pairs to Simplex state (delete)
pairsplit -g $GROUP $MEMBER -S

echo "Waiting for delete to complete."
pairevtwait -nomsg -s smpl -g $GROUP $MEMBER -t $SHORTTIMEOUT
RETURN="$?"
if [ "$RETURN" -ne $RC_COMPLETE ]
then
    echo "Error: Pairs did not transition into SMPL state in $SHORTTIMEOUT seconds."
    pairevtwait -g $GROUP $MEMBER -nowait
    echo "Return code was: ${RETURN}."
    exit 8
fi

echo
echo "Delete has completed."
echo
}

resync_pairs ()
{
# Resync previously split Business Copy pairs
date
echo "Resyncing ${GROUP}."

# Check if in required Split state
pairevtwait -g $GROUP $MEMBER -nowait > /dev/null 2>&1
RETURN=$?
if [ "$RETURN" -ne "$RC_SPLIT" ]
then
    echo "Error: Volumes $GROUP are not in a PSUS state!"
    pairevtwait -g $GROUP $MEMBER -nowait
    echo "Return code was: ${RETURN}."
    exit 9
fi

# Resync Split volumes back to paired state
pairresync -g $GROUP $MEMBER -c 15
echo
echo "Waiting for differential synchronization to complete."
let COUNTER=0
RESULT=-1
while [ "$RESULT" -ne "$RC_COMPLETE" ]
do
    pairevtwait -g $GROUP $MEMBER -s pair -nomsg -t $SHORTTIMEOUT
    RESULT=$?
    let COUNTER=$COUNTER+$SHORTTIMEOUT
    if [ "$COUNTER" -ge "$MASTERTIMEOUT" ]
    then
        echo "Error: Timeout while waiting for pair $GROUP to transition to PAIRED state"
        pairevtwait -g $GROUP $MEMBER -nowait
        echo "Return code was: ${RESULT}."
    fi
done

```



```

        exit 10
    fi
done
echo
echo "Synchronization completed."
}

restore_pairs_diff ()
{
# Restore volumes by making remote volume the source or P-VOL
if [ $CAN_DIFF_RESTORE -ne 1 ]
then
    echo "Insufficient microcode or RAIDMgr version for differential restore!"
    exit 15
fi
date
echo "Creating Business Copy (restore) pairs for group $GROUP"

# Check status of volumes to be paired. Must be in SUSP/SPLIT state.
pairevtwait -g $GROUP $MEMBER -nowait -nomsg
RETURN="?"
if [ $RETURN -ne $RC_SPLIT ]
then
    echo "Error: Pairs $GROUP are not in SPLIT state. Cannot initiate differential restore."
    pairevtwait -g $GROUP $MEMBER -nowait
    echo "Return code was: ${RETURN}."
    exit 13
fi

# Restore Business Copy Volumes with differential mode
pairresync -g $GROUP $MEMBER -c 15 -restore

# Monitor Status
echo
let COUNTER=0
echo "Waiting for differential re-synchronization to complete."
RESULT=-1
while [ "$RESULT" -ne "$RC_COMPLETE" ]
do
    pairevtwait -g $GROUP $MEMBER -s pair -nomsg -t $SHORTTIMEOUT
    RESULT=?
    let COUNTER=$COUNTER+$SHORTTIMEOUT
    if [ "$COUNTER" -ge "$MASTERTIMEOUT" ]
    then
        echo "Error: Timeout while waiting for pair $GROUP to transition to PAIRED state"
        pairevtwait -g $GROUP $MEMBER -nowait
        exit 14
    fi
done
echo "Volumes in group $GROUP synced. Restore completed"
date
}

restore_pairs_full ()
{
# Restore volumes by making remote volume the source or P-VOL
date
echo "Creating Business Copy (restore) pairs for group $GROUP"

# Check status of volumes to be paired. Must be in SIMPLEX state.
pairevtwait -g $GROUP $MEMBER -nowait -nomsg
RETURN="?"
if [ $RETURN -ne $RC_SIMPLEX ]
then
    echo "Error: Pairs $GROUP are not in SIMPLEX state. Cannot initiate copy."
    pairevtwait -g $GROUP $MEMBER -nowait
    echo "Return code was: ${RETURN}."
    exit 11
fi

# Create Business Copy volumes in reverse (restore)
paircreate -g $GROUP $MEMBER -vr -c 15

# Monitor Status
echo
let COUNTER=0

```

```

echo "Waiting for pairs to transition to PAIRED state."
RESULT=-1
while [ "$RESULT" -ne "$RC_COMPLETE" ]
do
    pairevtwait -g $GROUP $MEMBER -s pair -nomsg -t $SHORTTIMEOUT
    RESULT=$?
    let COUNTER=$COUNTER+$SHORTTIMEOUT
    if [ "$COUNTER" -ge "$MASTERTIMEOUT" ]
    then
        echo "Error: Timeout while waiting for pair $GROUP to transition to PAIRED state"
        pairevtwait -g $GROUP $MEMBER -nowait
        exit 12
    fi
done
echo "Volumes in group $GROUP synced. Copy completed"
date
}

display_pairs ()
{
    # Display config and status of Business Copy pairs
    pairedisplay -g $GROUP $MEMBER -fxc
}

#####
# Main Body of Code      #
#####

# Check if required number of parms
if [ "$#" -lt 2 ]
then
    echo "Usage: $0 GROUP ACTION [MEMBER]"
    echo "Must supply the HORCM Group name and action (member optional) to be performed."
    exit 1
fi

check_raidmgr

# Get BC group name and action to perform
GROUP=$1
ACTION=$2
DEVICE=$3

if [ -z "$DEVICE" ]
then
    MEMBER=""
else
    MEMBER="-d $DEVICE"
fi

# Select and execute action
case "$ACTION"
in
    'create')      create_pairs
                   break;;
    'split')       split_pairs
                   break;;
    'delete')      delete_pairs
                   break;;
    'resync')      resync_pairs
                   break;;
    'diffrestore') restore_pairs_diff
                   break;;
    'fullrestore') restore_pairs_full
                   break;;
    'display')     display_pairs
                   break;;
    *)             echo "Error: Unknown Action $ACTION requested!"
                   exit 2;;
esac

exit 0

```