

This is a recent installation of this patch XU*8*345 in a working test environment for the purpose of validating that it would install correctly and function.

PACKAGE: XU*8.0*345

```
-----
XPDATFORM Calculated      8171205
XPDATFORM2 Calculated     74234458
XPDAMENU Calculated       38908521
XPDAMSTR Calculated       12068983
XPDAPMES Calculated       3542508
XPDAPROC Calculated       11411376
XPDAPSVR Calculated       29959996
XPDAPUTL Calculated       38046316
XPDAUTL Calculated        36668626
XPDAUTL2 Calculated       38149774
XPDAUTO Calculated        70852100
XPDAUTO2 Calculated       34638202
XPDAUTOM Calculated       41314670
XPDDCS Calculated         18646094
XPDI Calculated           42987724
XPDI1 Calculated          18355565
XPDIB Calculated          5402970
XPDIL Calculated          21320112
XPDIL1 Calculated         36313386
XPDIPM Calculated         3311751
XPDIQ Calculated          21211593
XQ00 Calculated           49167293
```

22 Routines checked, 0 failed.

- 1 Load a Distribution
- 2 Verify Checksums in Transport Global
- 3 **Print Transport Global**
- 4 Compare Transport Global to Current System
- 5 Backup a Transport Global
- 6 Install Package(s)
Restart Install of Package(s)
Unload a Distribution

Select Installation Option: 3 Print Transport Global
Select INSTALL NAME: XU*8.0*345
=> COPY OF XU*8*345

This Distribution was loaded on with header of
COPY OF XU*8*345
It consisted of the following Install(s):
XU*8.0*345

Select one of the following:

- 1 Print Summary
- 2 **Print Summary and Routines**
- 3 Print Routines

What to Print: 2 Print Summary and Routines
DEVICE: HOME// VIRTUAL TELNET

PACKAGE: XU*8.0*345 Aug 03, 2012 2:06 pm

PAGE 1

TYPE: SINGLE PACKAGE
NATIONAL PACKAGE: KERNEL

TRACK NATIONALLY: YES
ALPHA/BETA TESTING: YES

DESCRIPTION:

See patch XU*8*345 in the National Patch Module on FORUM for complete information on this patch.

```

DELETE ENV ROUTINE:
DELETE PRE-INIT ROUTINE:
DELETE POST-INIT ROUTINE: No

```

PRINT TEMPLATE:	ACTION:
XPD PATCH CHECK FILE #9.76	SEND TO SITE
XPD PATCHES FILE #9.7	SEND TO SITE

SORT TEMPLATE:		ACTION:
XPD PATCH CHECK	FILE #9.76	SEND TO SITE
XPD PATCHES	FILE #9.7	SEND TO SITE

FUNCTION:	ACTION:
XPD PATCH DATE	SEND TO SITE
XPD PATCH DESIGNATION	SEND TO SITE
XPD PATCH SEQUENCE	SEND TO SITE

MAIL GROUP:	ACTION:
XPD VISTA PATCH	SEND TO SITE

ROUTINE:	ACTION:
XPDAFTP	SEND TO SITE
XPDAFTP2	SEND TO SITE
XPDAMENU	SEND TO SITE
XPDAMSTR	SEND TO SITE
XPDAPMES	SEND TO SITE
XPDAPROC	SEND TO SITE
XPDAPSVR	SEND TO SITE
XPDAPUTL	SEND TO SITE
XPDAUTL	SEND TO SITE
XPDAUTL2	SEND TO SITE
XPDAUTO	SEND TO SITE
XPDAUTO2	SEND TO SITE
XPDAUTOM	SEND TO SITE
XPDDCS	SEND TO SITE
XPDI	SEND TO SITE
XPDI1	SEND TO SITE
XPDIB	SEND TO SITE
XPDIL	SEND TO SITE
XPDIL1	SEND TO SITE
XPDIPM	SEND TO SITE
XPDIQ	SEND TO SITE
XQ00	SEND TO SITE
OPTION:	ACTION:
XPD AUTOMATIC PATCHING MENU	SEND TO SITE
XPD DESIGNATE INFORMATIONAL	SEND TO SITE
XPD MAIN	USE AS LINK FOR MENU/ITEM/SUBS
CRIBERS	
XPD MASTER - SERVANT ESTABLISH	SEND TO SITE

XPD MASTER - SERVANT RESPOND	SEND TO SITE
XPD PATCH LISTING	SEND TO SITE
XPD PATCH REPORT	SEND TO SITE
XPD PATCH SERVER	SEND TO SITE
XPD PATCH STATUS	SEND TO SITE
XPD PATCHING MASTER	SEND TO SITE
XPD PATCHING MENU	SEND TO SITE
XPD PATCHING SERVANT	SEND TO SITE
XPD SEND PATCH TO SERVANT	SEND TO SITE
XPD UNSTICK QUEUE	SEND TO SITE
XPD VAPU ADD SERVER TO GROUP	SEND TO SITE
XPD VAPU COMPLETE PATCH	SEND TO SITE
XPD VAPU CREATE CACHE/NT FILE	SEND TO SITE
XPD VAPU DELETE PATCH	SEND TO SITE
XPD VAPU DELETE TASKS	SEND TO SITE
XPD VAPU EDIT EXE LOCATION	SEND TO SITE
XPD VAPU EDIT INSTALL MODS	SEND TO SITE
XPD VAPU EDIT PKG MODS	SEND TO SITE
XPD VAPU ENABLE/DISABLE PKG	SEND TO SITE
XPD VAPU LIST TASKS	SEND TO SITE
XPD VAPU MAIN MENU	SEND TO SITE
XPD VAPU MANUAL INSTALL	SEND TO SITE
XPD VAPU MNG BSKTS	SEND TO SITE
XPD VAPU MNG PATCH INST QUE	SEND TO SITE
XPD VAPU PATCH MONITOR MENU	SEND TO SITE
XPD VAPU PRINT PATCH REPORT	SEND TO SITE
XPD VAPU REQUEUE TASKS	SEND TO SITE
XPD VAPU SEND MODIFIERS	SEND TO SITE
XPD VAPU SITE PARAMETERS	SEND TO SITE
XPDAUTO	SEND TO SITE

ALPHA/BETA TESTING: YES

INSTALLATION MESSAGE: NO

ADDRESS:

REQUIRED BUILDS:

XU*8.0*275

ACTION:

Don't install, remove global

XPDAFTP

XPDAFTP ;CAVHCS-619/TEA - AUTOPATCH CONTROL ROUTINE ;05/04/2004 09:37

;8.0;KERNEL;**345**;Jul 10, 1995;Build 1

;THIS ROUTINE RETRIEVES FTP KIDS DISTRIBUTIONS

;GET FTP FILE DIRECTORY FROM 9.72

S XPDAFTDR=\$P(\$G(^XPD(9.72,1,0)),U,10)

;GET KIDS INSTALL FILENAME FROM 9.74

S XPDAFTNM=A7

;GET CACHE Executable Location or VMS BATCH QUEUE Name

S XPDAEXNM=\$G(^XPD(9.72,1,1))

;GET FTP DESTINATION ISC

S XPDAFTDS=\$P(\$G(^XPD(9.72,1,0)),U,9),XPDAFTDS=\$P(\$G(^XPD(9.73,XPDAFTDS,0)),U)

I ^%ZOSF("OS")["DSM" K XPDATEST G DSM

I ^%ZOSF("OS")["OpenM-NT",\$ZV["NT" G NT

I ^%ZOSF("OS")["OpenM-NT",\$ZV["VMS" G CVMS

Q

DSM ;THIS CODE RUNS ON VMS/DSM SYSTEMS

FTPR ;

;S XPDDEV=IO U XPDDEV ;SET UP LOG FILE

;create the VMS .COM procedure

S COMFILE="XPDVMTFTP.com"

O COMFILE:NEW U COMFILE

W "\$! FTP RETRIEVAL PROCEDURE",!

W "\$ FTP "_XPDAFTDS,!

W "anonymous",!

W "test@test",!

W "lcd "_XPDAFTDR,!

W "cd software",!

W "ascii",!

W "get "_XPDAFTNM,!

W "quit",!

```

C COMFILE
;
;command to pass to either $SPAWN(DSM) or $ZF(CACHE)
S X=$&ZLIB.%SUBMIT("SYS$:"_XPDAFTDR_"XPDVMFTP.COM","/que="_XPDAEXNM_"/DELETE")
;BUILD FULL FILENAME FOR KIDS
S XPDFTPF=$P($G(^XPD(9.72,1,0)),U,10)_$G(XPDAFTNM)
G:$G(XPDATEST) EXIT^XPDAFTP2
G LOAD^XPDAFTP2
Q
;
TEST S XPDAFTDS="DOMAIN.EXT",XPDAFTNM="DG_53_P506.KID",XPDAFTDR="
[DSMTST]",XPDATEST=1
S XPDAEXNM=$G(^XPD(9.72,1,1))
G DSM
CVMS ;VMS/CACHE
S XPDAEXNM=$P($G(^XPD(9.72,1,1)),U) Q:XPDAEXNM']"'
OP2 D OPEN^%ZISH("XPDGTFTP",XPDAFTDR,"XPDVMFTP.com","W",300) Q:POP OP2 S XPDDEV=
IO U XPDDEV ;SET UP LOG FILE
U IO W "$ FTP "_XPDAFTDS,!,"anonymous",!,"test@VAPU",!,"cd software",!,"lcd "_X
PDAFTDR,!,"ascii",!,"get "_XPDAFTNM,!,"quit"
D CLOSE^%ZISH("XPDGTFTP")
S X=$ZF(-1,"SUBMIT "_XPDAFTDR_"XPDVMFTP.COM /QUE="_XPDAEXNM_"/DELETE")
Q
NT ;SET UP FTP SCRIPT AND GET FILE
OPEN ;OPEN A HOST FILE
S XPDAEXNM=$P($G(^XPD(9.72,1,1)),U) Q:XPDAEXNM']"'
D OPEN^%ZISH("XPDGTFTP",XPDAEXNM,"xpdscript.com","W",300) G:POP OPEN S XPDDEV=I
O U XPDDEV ;SET UP LOG FILE
U IO
W "$! FTP RETRIEVAL PROCEDURE",! ;For Cache VMS *added REM.
W "$ FTP "_XPDAFTDS,!
W "anonymous",!,"test@cavhcs",!,"cd software",!,"lcd "_XPDAFTDR,!,"ascii",!,"get
"_XPDAFTNM,!,"quit"
D CLOSE^%ZISH("XPDGTFTP")
S X=$ZF(-1,"@xpdscript.com")
;BUILD FULL FILENAME FOR KIDS
S XPDFTPF=$P($G(^XPD(9.72,1,0)),U,10)"\"_$G(XPDAFTNM)
G LOAD^XPDAFTP2

XPDAFTP2
XPDAFTP2 ;CAVHCS-619/TEA - AUTOPATCH CONTROL ROUTINE ;05/04/2004 09:38
;8.0;KERNEL;**345**;Jul 10, 1995;Build 1
LOAD ;
;NOW QUEUE IT UP
;GET PATCH INSTALLER'S DUZ
;GET AUTOINSTALL PARAMETERS
S XPDASITE=$G(^XPD(9.72,1,0)) I '$D(XPDASITE) S XQA("G.PATCHES")="",XQAMSG="Fil
e: 9.72-VAPU SITE PARAMETERS, has not been set up" D SETUP^XQALERT G EXIT
;CHECK TO SEE IF AUTOINSTALL IS ENABLED - QUIT IF NOT
I '$P(XPDASITE,"^",5) S XQAMSG="File: 9.72-VAPU SITE PARAMETERS, has not been set up" D
SETUP^XQALERT G EXIT
;SET UP QUEUE TIME
S X=$P(XPDASITE,"^",4),X1="0000" S XPDATM=$E(X1,1,(4-$L(X)))_X ;TIME IN MILITARY
;SET UP QUEUE DATE
D NOW^%DTC S X1=X,X2=$P(XPDASITE,"^",3) D C^%DTC S X=X_"_"_XPDATM,XPADT=$$FMTH
^XLFD(X)
;S DUZ(0)="@",
S ZTRTN="QUE^XPDAFTP2",ZTSAVE("XPD*")="",ZTSAVE("F*")="",ZTDESC="HDF AUTOPATCH
_XPDAFTNM_" INSTALL: "_XPDANM,ZTDTH=XPADT,ZTREQ="@ D ^%ZTLOAD
;SEND ALERT OF QUEUED INSTALL
S XQA("G.PATCHES")="",XQAMSG="HDF "_XPDAFTNM_" for "_XPDANM_" TSK: "_$G(ZTSK)_"
Queued for INSTALL. " D SETUP^XQALERT
Q ;COMMENT OUT TO DEBUG, Also comment out the D ^%ZTLOAD ON THE 3RD LINE UP
QUE ;TASK ENTRY POINT - QUE FOR DISTRIBUTION TYPE LOADS
;K XPDSTPXU,XPDASTPT,XPDASXWB,XPDASTPL,XPDASTPM,XPDINC,XPDATEMP
;GET PATCH INSTALLER'S DUZ

```

[illegible]

```

D CLOSE ^%ZISH("AUTOPATCH")
;Now let's check and see if the INSTALL was successful
S XPDAX0=$0(^XPD(9.7,"B",XPDANM,0)) I $G(XPDAX0) S XPDATMP=$G(^XPD(9.7,XPDAX0
,0)),XPDAX0=$P(XPDATMP,"^",9) D ;SEND ALERT AND EXIT
.;SEND ALERT AND EXIT
.S XPDASTAT=$S(XPDAX0=3:"INSTALLED SUCCESSFULLY!!!",1:"SOMETHING WRONG! INSTALL
UNSUCCESSFUL!!!!"),XQA("G.PATCHES")="",XQAMSG=$G(XPDANM)_ " Has an Install Status "_XPDASTAT
K:$G(XPDAX0)=3 XQAROU D SETUP^XQALERT
.Q
;Do a final check on the INSTALL file to ensure the SEQUENCE # is placed in FILE COMMENT
I $G(XPDAX0)=3 S DA=$0(^XPD(9.7,"B",XPDANM,0)) I $G(DA) K DIE S DIE="^XPD(9.7,"
I $P($G(^XPD(9.7,DA,2)), "^")'["#"_XPDASEQ] S XPDASEQ="SEQUENCE #"_XPDASEQ,DR="6///^S X=XPDASEQ" D
^DIE K DIE
G EXIT
STOP ;STOP ALL REQUIRED PROCESSES
S:$G(XPDSTPXU) XPDSTPXU=0 I XPDSTPXU D DISABL W !,"INHIBIT LOGONS COMMAND HAS BEEN ISSUED!" H 30
S:$G(XPDASTPL) XPDASTPL=0 I XPDASTPL D CLEAR^HLCS2,LLP^HLCS2(1) W !,"STOP HL7 BACKGROUND
PROCESSES COMMAND HAS BEEN ISSUED!" H 30
S:$G(XPDASTPM) XPDASTPM=0 I XPDASTPM S $P(^XMB(1,1,0),U,16)=1 S $P(^XMB(1,1,0),U,18)=1 W !,"STOP
MAILMAN COMMAND HAS BEEN ISSUED!" H 120
S:$G(XPDASXWB) XPDASXWB=0 I XPDASXWB D STOPALL^XWBTCP W !,"STOP BROKER LISTENER COMMAND HAS BEEN
ISSUED!" H 30
S:$G(XPDASTPT) XPDASTPT=0 I XPDASTPT H 60 D GROUP^ZTMKU("SMAN(NODE)"),GROUP^ZTMKU("SSUB(NODE)") W
!,"STOP TASKMAN COMMAND HAS BEEN ISSUED!" H 30
Q
START ;RESTART ALL REQUIRED PROCESSES
I $G(XPDSTPXU) D ENABLE W !,"ENABLE LOGONS COMMAND HAS BEEN ISSUED!"
I $G(XPDASTPT) D START^ZTMB H 90 W !,"RESTART TASKMAN COMMAND HAS BEEN ISSUED (DELAY 90)!"
I $G(XPDASTPL) D CLEAR^HLCS2,STARTF^HLCS2,LLP^HLCS2(0),STRT^HLCS2 W !,"RESTART HL7 BACKGROUND JOBS
COMMAND HAS BEEN ISSUED!"
I $G(XPDASXWB) D RESTART^XWBTCP W !,"RESTART DEFINED BROKER LISTENERS COMMAND HAS BEEN ISSUED!"
I $G(XPDASTPM) D START^XMKPL S $P(^XMB(1,1,0),U,18)=" " W !,"RESTART MAILMAN COMMAND HAS BEEN
ISSUED!!"
Q
DISABL ; disable logons in the volume set file...
N NVSX,NVSXN
S NVSX=0
F S NVSX=$0(^%ZIS(14.5,NVSX)) Q:'NVSX D
.S NVSXN=$P(^%ZIS(14.5,NVSX,0),"^")
.S $P(^%ZIS(14.5,NVSX,0),"^",2)="Y"
.S ^%ZIS(14.5,"LOGON",NVSXN)=1
.K NVSXN
Q
ENABLE ; (re-)enable logons in the volume set file...
S NVSX=0
F S NVSX=$0(^%ZIS(14.5,NVSX)) Q:'NVSX D
.S $P(^%ZIS(14.5,NVSX,0),U,2)="N"
.S NVSXN=$P(^%ZIS(14.5,NVSX,0),U)
.S ^%ZIS(14.5,"LOGON",NVSXN)=0
.K NVSXN
K NVSX
Q
ASK ;VALIDATE ROUTINE OPTION AND PROTOCOLS ARE VALID.
Q
REQUE ;REQUEUE TASKS THAT DIDN'T MEET PREREQUISITES
U XPDDEV W !!!,XPDANM," HAS BEEN REQUED FOR INSTALL!",!,"PLEASE CORRECT DEFICIENCY DESCRIBED IN
THE LOG FILE FOR SUCCESS!"
D ;
.S ZTRTN="QUE^XPDFT2",ZTSAVE("XPD*")="",ZTSAVE("XM*")=""
.S ZTDESC="Msg #_"$G(XMZ)_ " Sub: "_$P($G(^XMB(3.9,XMZ,0)), "^") D NOW^%DTC S X1=X,X2=1 D C^%DTC S
ZTDTH=%H_"_"$P($H,"_",2) S ZTREQ="ZTDTH,,ZTDESC;ZTRTN"
;SEND ALERT AND EXIT
S XQA("G.PATCHES")="",XQAMSG="Install "_$G(XPDANM)_ " KID DIST. "_$G(XPDFTNM)_ " Has been
REQUEUED! TASK: "_$G(ZTSK)_ " Log file Action Required. " D SETUP^XQALERT
U XPDDEV W !!!,"TASK REQUEUED TO TASK # "_$G(ZTSK) G EXIT
Q

```

```

GETANS ;CHECK FOR VAPU INSTALL MODIFIERS (9.74)
;CHECK FOR VAPU PACKAGE INSTALL MODIFIERS
;INSTALL MODIFIERS TAKE PRECEDENCE OVER PACKAGE MODIFIERS
;OUT OF ORDER PROTOCOLS/OPTIONS HAVE EQUAL PRECEDENCE.
;Any OPTION/PROTOCOL defined in either file will be taken offline.
;
;GETTING INSTALL MODS ---> GO TO PACKAGE MODS IF NONE EXIST
;SHOULD HAVE XPDANM HERE
S DIC="^XPD(9.74,"X=$G(XPDANM),DIC(0)="M" D ^DIC I Y>0 S XPDAX1=X,Y,XPDATMP=$G(^XPD(9.74,+Y,0)) D
;
.S
AUTOINST=$P(XPDATMP,U,9),XPDSTPXU=$P(XPDATMP,U,6),XPDASTPT=$P(XPDATMP,U,2),XPDASTPL=$P(XPDATMP,
U,3),XPDASTPM=$P(XPDATMP,U,4),XPDASXWB=$P(XPDATMP,U,5),XPD00=$P(XPDATMP,U,10)
;
;NOW SEE IF THERE ARE PACKAGE MODIFIERS, IF INSTALL MODIFIERS WERE NOT FOUND
S XPDAPK=$P($G(XPDANM),"*"),XPDAPK=$O(^DIC(9.4,"C",XPDAPK,0)),XPDAX1=$O(^XPD(9.75,"B",XPDAPK,0)) I
XPDAX1,+Y<1 S XPDATMP=$G(^XPD(9.75,XPDAX1,0)) D ;
.S XPDSTPXU=$P(XPDATMP,U,6),XPDASTPT=$P(XPDATMP,U,2),XPDASTPL=$P(XPDATMP,U,3
),XPDASTPM=$P(XPDATMP,U,4),XPDASXWB=$P(XPDATMP,U,5),XPD00=$P(XPDATMP,U,10)
;BUILD THE XPDQX ARRAY COMPOSED OF OPTIONS AND PROTOCOLS FROM FILES 9.74 AND 9.75
;THIS ARRAY IS USED IN THE KERNEL ROUTINE XQ00
;XPDAX1=9.74'S IEN
;XPDAX1=9.75'S IEN
;
K ^TMP($J),^TMP("PRO1"),^TMP("OP1")
;OPTIONS GO FIRST
M:$G(XPDAX1) ^TMP($J)=^XPD(9.74,XPDAX1,3) M:$G(XPDAX1) ^TMP($J)=^XPD(9.75,XPDAX1,2)
M ^TMP("OP1")=^TMP($J,"B")
K ^TMP($J)
S J=" F I=1:1 S XPDQX(I)=$O(^TMP("OP1",J)),J=XPDQX(I) Q:J']"
;
;NOW PROTOCOLS
M:$G(XPDAX1) ^TMP($J)=^XPD(9.74,XPDAX1,2) M:$G(XPDAX1) ^TMP($J)=^XPD(9.75,XPDAX1,1)
M ^TMP("PRO1")=^TMP($J,"B")
K ^TMP($J)
S J=" F K=1:1 S XPDQX(K+I)=$O(^TMP("PRO1",J)),J=XPDQX(K+I) Q:J']"
K ^TMP($J),^TMP("PRO1"),^TMP("OP1"),I,J,K
Q
EXIT ;KILL VARIABLES AND QUIT
K %H,XPD AUTO,XPDACHK,XPDACHK2,XPDACHK3,XPDACHK4,XPDDEV,XPDADIR,XPD AERNM,XPDSTPXU,XPDASTPT
K XPDASXWB,XPDASTPL,XPDASTPM,XPDINC,XPDATMP,XPDANM,XPDADT,X,X1,X2,X3,Y,XPDINC,XPD AERXM
K XPD AFN,XPD00,XPD A000,XPDAPKNM,XPDASEQ,XPDASITE,XPDASTOP,XPDATM,XPDAXMZ,XPDQX,AUTOINST,DIC
K DIR,I,J,MODE,POP,TEXT,TYPE,XMDUZ,XMP2,XMSUB,XMTEXT,XMY,XMZ,XQA,XQAMSG,ZTDESC,ZTDTH,ZTRTN,ZTSAVE
K %,XPDAPK,XPDASXWB,XPDAX1,XPD AFTNM,XPD AIX,XPDASTAT,XPDAX0,DA,DR,XQADATA,XQAID,ZTSK
D CLOSE^%ZISH("AUTOPATCH")
Q

XPDAMENU
XPDAMENU ;REMOTE PATCHING MENU UTILITY;DAF;3/30/04
;8.0;KERNEL;**345**;Jul 10, 1995;Build 1
Q
SENDP ;SEND PATCH(ES)
I '$D(^XPD(9.77,1,1,"B")) W !,"YOU ARE NOT THE MASTER OF ANY SITES SO YOU CANNOT SEND PATCHES TO
ANYONE!" H 5 Q
I $P($G(^XPD(9.77,1,0)),"^",2) ["S" W !,"SORRY, YOU ARE A SERVANT AND CANNOT SEND PATCHES!" H 5 Q
N DA,DIC,DIE,DIR,DIRUT,DR,DTOUT,DUOUT,HDR,I,PATCH,RSITE,RSP,SSITE,STOP,X,XMY,XMZ,Y,ZI,ZN,ZNM,ZY
W $$$CNHDR^XPDAPUTL("Send Patches to Servant Site(s) Option")
S RSP="",STOP=0 K DIR
F Q:STOP D Q:$D(DIRUT)
.S DIR("?")="You must select a message that is a KIDS build. A non-patch or a Packman will not
work."
.S DIR(0)="P^3.9:EMZ",DIR("A")="Enter a patch to be loaded at a remote site",DIR("S")="I
$$P TYPE^XPDAPUTL(Y)" D ^DIR
.Q:$D(DIRUT)
.S DIC(0)="LMZ",DIC=9.76,X=+Y D ^DIC D
..I $D(DTOUT)!($D(DUOUT))!(Y<0) S STOP=1 Q

```

```

..I $$PLUG^XPDAPUTL(+Y,1)']"" W !,"Sorry, that message has been purged from the
message file. It may not be on the remote site." Q
..S ZY=Y S DA=+Y,DIE=9.76,DR=".5///^S X=$$PLUG^XPDAPUTL(+ZY,0);1///^S X=DUZ;2///NOW" D ^DIE
..S RSP=RSP_$S(RSP]"":":"",";1:"")_$$MSG^XPDAPUTL($$PLUG^XPDAPUTL(+ZY,1))
K DIR
S ZN="" F I=1:1 S ZN=$O(^XPD(9.77,1,1,"B",ZN)) Q:ZN']"" S I(I)=ZN
I '$D(I(2)) S RSP(1)=I(1) G CONT
S DIR(0)="S^", (ZN,RSP(1))="" F S ZN=$O(I(ZN)) Q:ZN']"" D
.S DIR(0)=DIR(0)_ZN_"_I(ZN)_"_";"
.S RSP(1)=RSP(1)_$S(RSP(1)]"":":"",";1:"")_I(ZN)
S DIR(0)=$E(DIR(0),1,($L(DIR(0))-1))_"_";_"_$L(DIR(0),"_");_"_":All of the above"
S DIR("A")="Which site(s) do you want to send this (these) patch(es) to",DIR("B")=$L(DIR(0),"_");"
S ZN="" F D ^DIR Q:$D(DIRUT) S ZN=ZN_$S(ZN]"":":"",";1:"")_Y Q:Y=($L(RSP(1),"_");)+1)
Q:$D(DTOUT)
CONT S RSP(2)=""
I +ZN'=(($L(RSP(1),"_");)+1) F I=1:1:$L(ZN,"_");" S RSP(2)=RSP(2)_$S(RSP(2)]"":":"",";1
:"")_P(RSP(1),"_";"_$P(ZN,"_";"I))
S:RSP(2)]"" RSP(1)=RSP(2)
Q:RSP']""!(RSP(1)']""")
W @IOF,!,"OK, Let me get this straight, you wish to have these patches:"
N PATCH S PATCH=""
F ZI=1:1:$L(RSP,"_");" W !,ZI,". ",$P($G(^XMB(3.9,$P(RSP,"_";"ZI),0)),"^",1) S PAT
CH($P(RSP,"_";"ZI))=$P($G(^XMB(3.9,$P(RSP,"_";"ZI),0)),"^",1)
W !,"installed on these servants:"
F ZI=1:1:$L(RSP(1),"_");" W !,ZI,". ",$P(RSP(1),"_";"ZI)
S X=$$DIRA^XPDAPUTL("Do you wish to continue","NO") Q:X<1
S SSITE=$$SSITE^XPDAPUTL()
F ZNM=1:1:$L(RSP(1),"_");" D
.S HDR="~~#~~ REQUEST TO HAVE PATCHES LOADED AT A REMOTE SITE"
.F ZI=1:1:$L(RSP,"_");" D
..S XMY("G.XPD VISTA PATCH@"_P(RSP(1),"_";"ZNM))="" ,XMZ=$P(RSP,"_";"ZI) D ENT1^XMD ;SEND A COPY TO
OF EACH PATCH MAKE SURE IT IS THERE
..W:'$$STDBUL^XPDAPMES(4,HDR_"_";"SSITE_"_";"P(RSP(1),"_";"ZNM)_"_";"PATCH($P(RSP,"_";"ZI))) !,"I was
not able to send a message to the "_P(RSP(1),"_";"ZNM)_" site."
Q
STATUS ;GET REMOTE SITE PATCH STATUS
I '$D(^XPD(9.77,1,1,"B")) W !,"YOU ARE NOT THE MASTER OF ANY SITES!" H 5 Q
I $P($G(^XPD(9.77,1,0)),"^",2) ["S" W !,"SORRY, YOU ARE A SERVANT AND CANNOT REQUEST PATCH STATUS
OF ANOTHER SITE!" H 5 Q
N DIR,DTOUT,DUOUT,HDR,I,RSITE,RSP,SSITE,X,Y,ZI,ZN,ZNM,ZQUIT S ZQUIT=0
W $$SCNHDR^XPDAPUTL("Request Patch Status From Servant Site(s) Option")
S ZN="" F I=1:1 S ZN=$O(^XPD(9.77,1,1,"B",ZN)) Q:ZN']"" S I(I)=ZN
I '$D(I(2)) S RSP=I(1) G CONT1
S DIR(0)="S^", (ZN,RSP)="" F S ZN=$O(I(ZN)) Q:ZN']"" D
.S DIR(0)=DIR(0)_ZN_"_I(ZN)_"_";"
S DIR(0)=$E(DIR(0),1,($L(DIR(0))-1))_"_";_"_$L(DIR(0),"_");_"_":All of the above"
S DIR("A")="Which site(s) do you want to get a status report from",DIR("B")=$L(DIR(0),"_");"
S RSP="" F D ^DIR Q:$D(DIRUT)!(ZQUIT) D:Y(0) ["All of the above" Q:ZQUIT S R
SP=RSP_$S(RSP]"":":"",";1:"")_(I(Y))
.S (RSP,ZI)="" F S ZI=$O(I(ZI)) Q:ZI']"" S RSP=RSP_$S(RSP]"":":"",";1:"")_(I(ZI))
.S ZQUIT=1
Q:$D(DTOUT)
W @IOF
CONT1 W !,"OK, Let me get this straight, you want a patch status report from these sites:"
F ZI=1:1:$L(RSP,"_");" W !,ZI,". ",$P(RSP,"_";"ZI)
S X=$$DIRA^XPDAPUTL("Do you wish to continue","NO") Q:X<1
F ZNM=1:1:$L(RSP,"_");" D
.S HDR="~~#~~ REQUEST FOR PATCH STATUS FROM A SERVANT SITE"
..W:'$$STDBUL^XPDAPMES(5,HDR_"_";"SSITE^XPDAPUTL_"_";"P(RSP,"_";"ZNM)) !,"I was not able to send a
message to the "_P(RSP,"_";"ZNM)_" site."
Q
REPORT ;REPORT OF PATCHES OVER TIME PERIOD
N BY,DIC,DIR,EDATE,FLDS,FR,L,TO,SDATE,ZNM
W $$SCNHDR^XPDAPUTL("Report of Remote Patch Status in My Database")
S DIR(0)="D",DIR("A")="Enter the Start date of the report" D ^DIR Q:$D(DIRUT)
S SDATE=Y K DIR

```



```

S DIR(0)="D",DIR("A")="Enter the Stop date of the report" D ^DIR Q:$D(DIRUT)
S EDATE=Y K DIR
S DIC=9.76,L=0,FLDS="[XPD PATCH CHECK]",BY="[XPD PATCH CHECK]",FR=SDATE,TO=EDATE
D EN1^DIP
Q
MARK(ANS) ;MARK A PATCH AS INFORMATIONAL IN THE XREF
N DIR,I,PATCH,RSP,SEQ,X,Y
I '$D(^XPD(9.77,1,1,"B"))&('ANS) W !,"YOU ARE NOT A MASTER SITE SO YOU CANNOT USE THIS OPTION AS A
MASTER",!,"GO BACK AND USE THIS OPTION UNDER THE SERVANT MENU" H 3 Q
I '$D(^XPD(9.77,1,1,"B"))&(ANS) W !,"YOU ARE A MASTER SITE AND THEREFORE YOU CANNOT USE THIS OPTION
AS A SERVANT",!,"GO BACK AND USE THIS OPTION UNDER THE MASTER MENU" H 3 Q
START S DIR(0)="F7:20",DIR("A")="Enter the informational patch to be marked as installed" D ^DIR
Q:$D(DIRUT) I Y'?2.5U1"*"1.3N1"."1.2N1"*"1.3N W !,"That is not a proper patch format" H 3 G START
S PATCH=Y
K DIR S DIR(0)="N^1:999",DIR("A")="Enter the sequence number" D ^DIR Q:$D(DIRUT) S PATCH=PATCH_
SEQ " Y
;K DIR S DIR(0)="S^S:Send to Servant;A:Apply locally",DIR("A")="Do you want to send this change to
a Servant or Apply locally"
;S DIR("B")="A" D ^DIR Q:$D(DIRUT) S ANS=Y
G:ANS FINI
S ZN="" F I=1:1 S ZN=$O(^XPD(9.77,1,1,"B",ZN)) Q:ZN']"" S I(I)=ZN
I '$D(I(2)) S RSP=I(1) G CONT2
S DIR(0)="S^", (ZN,RSP)=" F S ZN=$O(I(ZN)) Q:ZN']"" D
.S DIR(0)=DIR(0)_ZN_"_I(ZN)_"_";
S DIR(0)=$E(DIR(0),1,($L(DIR(0))-1))_"_";_$_L(DIR(0),";")_"_":All of the above"
S DIR("A")="Which site(s) do you want to send this patch designation to",DIR("B")=$_L(DIR(0),";")
S RSP="" F D ^DIR Q:$D(DIRUT) S RSP=RSP_$S(RSP]"";";",1:"")_(I(Y))
Q:$D(DTOUT)
CONT2 S SSITE=$$SSITE^XPDAPUTL()
F ZNM=1:1:$L(RSP,";") D
.S HDR="~~#~~ REQUEST TO HAVE PATCH DESIGNATION CHANGED TO INFORMATIONAL AT REMOTE SITE"
.F ZI=1:1:$L(RSP,";") D
..W:'$$STDBUL^XPDAPMES(6,HDR_"_";_SSITE_"_";_$_P(RSP,";";,ZNM)_"_";_PATCH) !,"I was not able to send a
message to the "_$_P(RSP(1),";";,ZNM)_" site."
FINI S:ANS ^XPD(9.76,"E",$_P(PATCH,"*",1,2)_"*"_Y,9999)=" "
W !,"OK. I entered a notation for this patch sequence."
Q

XPDAMSTR
XPDAMSTR ;REMOTE PATCHING UTILITY; MASTER-SERVANT ESTABLISHMENT;DAF;3/29/04
;8.0;KERNEL;**345**;Jul 10, 1995;Build 1
Q
EST ;ESTABLISH MASTERSHIP OVER A SITE
D CHK^XPDAPUTL
N DIR
W $$$SCNHDR^XPDAPUTL(" MASTER-SERVANT ESTABLISHMENT OPTION")
D ASK
EXIT Q
ASK N HDR,RSP,SSITE,X,ZI S RSP="" D
.F S X=$$DIRP^XPDAPUTL(4.2,"Please select a site to establish mastership over") Q:X<1 S
RSP=RSP_$S(RSP]"";";",1:"")_X
Q:RSP']""
W !,"You have these Domains selected to establish mastership over:"
F ZI=1:1:$L(RSP,";") W !,ZI,". ",$_P($_P(RSP,";";,ZI),"^",2)
S X=$$DIRA^XPDAPUTL("Do you wish to continue","NO") Q:X<1
F ZNM=1:1:$L(RSP,";") D
.S SSITE=$$SSITE^XPDAPUTL()
.S HDR="~~#~~ REQUEST TO ESTABLISH PATCHING MASTERSHIP OVER A REMOTE SITE"
.W:'$$STDBUL^XPDAPMES(1,HDR_"_";_SSITE_"_";_$_P($_P(RSP,";";,ZNM),"^",2)) !,"I was not able to send a
message to the "_$_P($_P(RSP,";";,ZNM),"^",2)_" domain."
Q
RESP ;RESPOND TO MASTER REQUEST
D CHK^XPDAPUTL
I '$D(^XPD(9.79,"XPD")) W !,"THERE ARE NO PENDING REQUEST MESSAGES!" H 5 Q
N DA,DATA,DIE,DR,RESP,SSITE,ZN,ZNM,ZX
S ZNM="" F S ZNM=$O(^XPD(9.79,"XPD",ZNM)) Q:ZNM']"" D

```

```

.S ZN=$0(^XPD(9.79,"XPD",ZNM,""))
.D:$P($G(^XPD(9.79,ZN,0)),",^",3)=1
..S DATA=$P($G(^XPD(9.79,ZN,0)), "~#~",2)
..S SSITE=$P(DATA,"~",2)
..I $P($G(^XPD(9.77,1,0)),",^",3)]" D Q
...S ZX=$$SEND($$SSITE^XPDAPUTL(),$$RSITE^XPDAPUTL($P(DATA,"~",2)),0,"This site already has a
patching Master", "",1)
...W:$PROCESS^XPDAPUTL(ZN) !,"Message processed..."
..I $P($G(^XPD(9.77,1,0)),",^",2)]"M" D Q
...S ZX=$$SEND($$SSITE^XPDAPUTL(),$$RSITE^XPDAPUTL($P(DATA,"~",2)),0,"This site is a Master site
and cannot be a Servant site", "",1)
...W:$PROCESS^XPDAPUTL(ZN) !,"Message processed..."
..W $$SCNHDR^XPDAPUTL(" Servant Request Response Option")
..W !,SSITE_ " wishes to be your patching Master site."
..S RESP=$$DIRA^XPDAPUTL("Do you agree to this","NO")
..Q:RESP<0
..I RESP W:$MASTER^XPDAPUTL(SSITE) !,"Mastership filed..."
..S DATA(2)=$TR($P(^XPD(9.77,1,0)),",^",5,7),",^",";")
..S RSITE=SSITE,SSITE=$$SSITE^XPDAPUTL()
..S HDR="MASTERSHIP RESPONSE FROM "_SSITE
..W:$SEND(SSITE,RSITE,RESP,$S(RESP:" ",1:"denied by client site"),$S(RESP:DATA(2),1:""),1)
!, "Response message sent to "_$S(RESP:"accepted",1:"denied")_ " Master site"
..W:$PROCESS^XPDAPUTL(ZN) !,"Message processed..."
Q
SEND(SSITE,RSITE,RESP,REASON,DATA,TYPE) ;SEND THE MESSAGE
S HDR=~#~ THE RESPONSE TO YOUR "_$S(TYPE:"MASTERSHIP REQUEST OVER "_SSITE,1:"PATCH REQUEST")_
WAS "_$S(RESP:"ACCEPTED",1:"REFUSED")
Q $$STDBUL^XPDAPMES($S(RESP:2,1:3),HDR_";" _SSITE_";" _RSITE_";" _$S(RESP:DATA,1:REASON))

XPDAPMES
XPDAPMES ;REMOTE PATCHING BULLETIN UTILITY;DAF;3/30/04
;8.0;KERNEL;**345**;Jul 10, 1995;Build 1
Q
STDBUL(TPE,VAR) ;STANDARD BULLETIN START POINT
;TYPE DETERMINES THE TYPE OF MESSAGE SENT
;VAR IS THE LIST OF VARIABLES THAT ARE NEEDED FOR THAT PARTICULAR TYPE
N %,DMN,HEADER,PATCH,SSITE,X,XMSITE,XMTEXT,XMSUB,XMY,XMZ,ZNM,ZN,ZNUM
S VAR=(TPE_";" _VAR)
Q:'$$CHECK^XPDAPUTL(TPE,VAR) 0
F ZNM=1:1:$L(VAR,";") D
.S VAR(ZNM)=$P($G(^XPD(9.78,TPE,1,ZNM,0)),",^",1)
.S @VAR(ZNM)=$P(VAR,";",ZNM)
S DMN=$S($P(VAR,";",4)]":$P(VAR,";",3),1:$P(VAR,";",4))
Q:DMN']" " "
S XMY("G.XPD VISTA PATCH@"_DMN)=" "
;S XMY("S.XPD PATCH SERVER@"_DMN)=" "
;S XMY("G.XPD VISTA PATCH")=" "
;S XMY("S.XPD PATCH SERVER")=" "
S XMSUB=$E(HEADER,"~" _SSITE,1,60) ;VARIABLES LIKE HEADER ARE DYNAMICALLY MADE AND CLEANED UP.
S XMTEXT="PATCH(",XMY(DUZ)=" "
S PATCH(1)="This message is a formulated bulletin for the VA patch package."
F ZNM=1:1:$P($G(^XPD(9.78,TPE,2,0)),",^",4) D
.S PATCH(ZNM)=$G(^XPD(9.78,TPE,2,ZNM,0)) D:PATCH(ZNM)[ "~#2~"
..S PATCH=$P(PATCH(ZNM), "~#2~",2),PATCH(ZNM)="~#2~"
..F ZN=1:1:$L(VAR,";") D
...S PATCH(ZNM)=PATCH(ZNM)_$S(ZN>1:" ",1:"")_ $P(VAR,";",ZN)
D:$D(ZTXT)
.S ZN=" F S ZN=$0(ZTXT(ZN)) Q:ZN']" S ZNM=ZNM+1 S PATCH(ZNM)=ZTXT(ZN)
D NOW^%DTC S PATCH(ZNM+1)="Date and time of transaction: "_+$E(%,4,5)_" / "_+$E(%,
6,7)_" / "_($E(%,1,3)+1700)_"@"_ $P(%,",",2)
N DO,DIX,DIY
D ^XMD
F ZNM=1:1:$L(VAR,";") K @VAR(ZNM)
Q XMZ

```

```

XPDAPROC
XPDAPROC ;REMOTE PATCHING UTILITY; PATCH PROCESSOR;DAF;3/29/04
;;8.0;KERNEL;**345**;Jul 10, 1995;Build 1
Q
DAILY ;RUN PATCH CHECK ROUTINELY
I $P($G(^XPD(9.77,1,0)),",^",2)="M" D ^XPDAPSVR
E D ^XPDAPSVR,SETUP H 200 D CHECK,^XPDAPSVR Q
SETUP ;SETUP PATCH TO RUN
N ZNAME,ZNM,ZNUM,ZPATCH,ZZNM
S ZNM="" F S ZNM=$O(^XPD(9.76,"ZZ",ZNM)) Q:ZNM']"" D
.S XMZ=+$P($G(^XPD(9.76,ZNM,0)),",^",1)
.S ZPATCH($P($$SEQ^XPDAPUTL(XMZ),",*",1,2),$P($$SEQ^XPDAPUTL(XMZ),",*",3))=XMZ_"^"_ZNM S ZNM="" F S
ZNM=$O(ZPATCH(ZNM)) Q:ZNM']"" D
.S ZNUM="" F S ZNUM=$O(ZPATCH(ZNM,ZNUM)) Q:ZNUM']"" D
..S XMZ=$P(ZPATCH(ZNM,ZNUM),",^",1),ZZNM=$P(ZPATCH(ZNM,ZNUM),",^",2)
..D ^XPD AUTO
..S ^XPD(9.76,"ZAZ",ZZNM)=" "
..K ^XPD(9.76,"ZZ",ZZNM)
..H 10
Q
CHECK ;CHECK TO SEE IF A PATCH HAS RUN
N DA,DIE,DR,HDR,IEN,NMBR,PATCH,XMZ,ZIEN,ZNM,ZTXT,ZX,ZZI,ZZZI
S ZNM="" F S ZNM=$O(^XPD(9.76,"ZAZ",ZNM)) Q:ZNM']"" D
.S PATCH=$P($G(^XPD(9.76,ZNM,0)),",^",3)
.S IEN=$O(^XPD(9.7,"B",PATCH,"")) Q:IEN']""
.Q:$P($G(^XPD(9.7,IEN,0)),",^",9)'=3
.K ZTXT S ZTXT=""
.S NMBR="" F ZZI=1:1 S NMBR=$O(^XPD(9.7,IEN,"MES",NMBR)) Q:NMBR']"" D
..S:$G(^XPD(9.7,IEN,"MES",NMBR,0))'=" " ZTXT(ZZI)=$G(^0))
.S NMBR="" F ZZI=1:1 S NMBR=$O(ZTXT(ZZI)) Q:NMBR']""
.D:ZZI>30
..S NMBR="" F ZZZI=25:1:(ZZI-5) S NMBR=$O(ZTXT(ZZZI)) K ZTXT(ZZZI)
..F ZZI=26,27,28 S ZTXT(ZZI)=" "
.S HDR="~~#~~ SERVANT SITE PATCH COMPLETION MESSAGE FROM "
.S ZX=$$STDBUL^XPDAPMES(4,HDR_" "; "_$$$SITE^XPDAPUTL_" ; "_$P($G(^DIC(4.2,+$P($G(^X
PD(9.77,1,0)),",^",4),0)),",^",1)_" ; "_PATCH)
.I ZX D
..K ^XPD(9.76,"ZAZ",ZNM)
..S ZNUM=$P($G(^XPD(9.76,ZNM,0)),",^",1) Q:ZNUM']""
..I ZNUM["DIZ(405375" S ZIEN=$P($G(^DIZ(405375,+ZNUM,0)),",^",1)
..E S ZIEN=+ZNUM
..S DIE=9.76,DA=ZNM,DR="3///^S X=$$SEQ^XPDAPUTL(ZIEN)" D ^DIE
Q
STATUS(TPE) ;SHOW STATUS OF PATCHES THAT ARE NOT INSTALLED YET
I TPE,$P($G(^XPD(9.77,1,0)),",^",2)["M" W !,"SORRY, YOU ARE A MASTER SITE AND THEREFORE CANNOT DO
PATCHES TO YOURSELF" H 5 Q
W:TPE $$$CNHDR^XPDAPUTL(" Status of Pending Patches Option")
N CNT,LCNT,ZNM,ZI I TPE N ZTXT
S (LCNT,CNT)=0
S CNT=CNT+1 S ZTXT(CNT)="Patches that have not been sent to queue yet:"
S ZNM="" F ZI=1:1 S ZNM=$O(^XPD(9.76,"ZZ",ZNM)) Q:ZNM']"" D
.S CNT=CNT+1,LCNT=1 S ZTXT(CNT)=ZI_" " _$P(^XPD(9.76,ZNM,0),",^",3)
I 'LCNT S CNT=CNT+1 S ZTXT(CNT)="None. "
S LCNT=0
S CNT=CNT+1 S ZTXT(CNT)="Patches that have been queued but are have not been installed yet:"
S ZNM="" F ZI=1:1 S ZNM=$O(^XPD(9.76,"ZAZ",ZNM)) Q:ZNM']"" D
.S CNT=CNT+1,LCNT=1 S ZTXT(CNT)=ZI_" " _$P(^XPD(9.76,ZNM,0),",^",3)
I 'LCNT S CNT=CNT+1 S ZTXT(CNT)="None. " D:TPE
.S ZNM="" F S ZNM=$O(ZTXT(ZNM)) Q:ZNM']"" W !,ZTXT(ZNM)
Q
UNSTIK ;UNSTICK THE ZAZ QUE
I $P($G(^XPD(9.77,1,0)),",^",2)["M" W !,"Sorry, this feature can only be run on a servant" H 5 Q
N ZNM
S ZNM="" F S ZNM=$O(^XPD(9.76,"ZAZ",ZNM)) Q:ZNM']"" D
.S ^XPD(9.76,"ZZ",ZNM)=" "
.K ^XPD(9.76,"ZAZ",ZNM)

```

Q

XPDAPSVR

XPDAPSVR ;REMOTE PATCHING UTILITY; MESSAGE SERVER;DAF;3/29/04

;8.0;KERNEL;**345**;Jul 10, 1995;Build 1

N CNT,DA,DATA,DIC,DIE,DR,STOP,TMP,TYPE,X,Y,ZHD,ZN,ZNM,ZSTP,ZTXT,ZX,ZZN

D CHK^XPDAPUTL

S STOP=\$\$STOP^XPDAPUTL

S ZHD="~~#~~" F S ZHD=\$0(^XMB(3.9,"B",ZHD)) Q:ZHD']"" D

.S ZNM="" F S ZNM=\$0(^XMB(3.9,"B",ZHD,ZNM)) Q:ZNM']"" D

..Q:\$D(^XPD(9.79,"B",ZNM))!(ZNM<STOP)

..S TMP(ZNM)=""

S ZNM="" F S ZNM=\$0(TMP(ZNM)) Q:ZNM']"" D

.S ZX=\$\$POPUL^XPDAPUTL(ZNM)

PROCESS ;PROCESS ENTRIES IN 9.79

S ZNM=""

F S ZNM=\$0(^XPD(9.79,"XPD",ZNM)) Q:ZNM']"" D

.S ZN=\$0(^XPD(9.79,"XPD",ZNM,""))

.I \$P(\$G(^XPD(9.79,ZN,0)), "^",3)=99,\$\$PROCESS^XPDAPUTL(ZN) Q

.S ZZN=0 F S ZZN=\$0(^XMB(3.9,ZNM,2,ZZN)) Q:ZZN']"" D

..S DATA=\$G(^XMB(3.9,ZNM,2,ZZN,0))

..D:DATA["~~#2~~" Q

...S DATA(1)=\$P(DATA,"~~#2~~",2)

...S \$P(^XPD(9.79,ZN,0),"^",3)=\$P(DATA(1),"~",1)

...S \$P(^XPD(9.79,ZN,0),"^",4)=\$P(DATA(1),"~",2,99)

.S TYPE=\$P(^XPD(9.79,ZN,0),"^",3)

.D:+TYPE>0&(TYPE<99) TEXT,@TYPE K ZTXT S ZTXT="" ;DAF;11/5/04 TRY TO FIX CARRYING OVER TEXT STRING

Q

TEXT ;ESTABLISH ZTXT ARRAY

N NMBR,START,ZZI S START=0

S NMBR="" F ZZI=1:1 S NMBR=\$0(^XMB(3.9,ZNM,2,NMBR)) Q:NMBR']"" D

.S:\$G(^XMB(3.9,ZNM,2,NMBR,0))["~~#1~~" START=1

.Q:'START

.S ZTXT(ZZI)=\$G(^XMB(3.9,ZNM,2,NMBR,0))

Q

1 ;MASTER ESTABLISHMENT

I \$P(\$P(\$P(\$G(^XPD(9.79,ZN,0)), "^",4),"~~#~~",2),"~",2)=\$\$SSITE^XPDAPUTL() S ZX=\$\$PROCESS^XPDAPUTL(ZN) Q

I \$P(\$P(\$P(\$G(^XPD(9.79,ZN,0)), "^",4),"~~#~~",2),"~",3)'=\$\$SSITE^XPDAPUTL()!(\$P(\$G(^XPD(9.77,1,0)), "^",3))"" D Q

.S ZX=\$\$PROCESS^XPDAPUTL(ZN)

Q

2 ;MASTER REQUEST ACCEPTED RESPONSE

N CNT,LST,ZX

I \$P(\$G(^XPD(9.77,1,0)), "^",2)["S" S ZX=\$\$PROCESS^XPDAPUTL(ZN) Q

I \$P(\$G(^XPD(9.77,1,0)), "^",2)']"" S \$P(^0),"^",2)="M"

S DATA(2)=\$P(DATA(1),"~~#~~",2)

I \$P(DATA(2),"~",2)]"",\$D(^XPD(9.77,1,1,"B",\$P(DATA(2),"~",2))) S ZX=\$\$PROCESS^XPDAPUTL(ZN) Q

S ZX=\$\$CHKMST^XPDAPUTL(),CNT=(\$P(ZX,"^",1)+1),LST=(\$P(ZX,"^",2)+1)

S ^XPD(9.77,1,1,LST,0)=\$P(DATA(2),"~",2)_"^"_STR(\$P(DATA(2),"~",4,6),"~", "^")

S \$P(^XPD(9.77,1,1,LST,0),"^",2)=\$P(^0),"^",2)_"^"_P(\$\$RSITE^XPDAPUTL(\$P(^0),"^",1)),"^",2)

S ^XPD(9.77,1,1,"B",\$P(DATA(2),"~",2),LST)=""

S \$P(^XPD(9.77,1,1,0),"^",3,4)=LST_"^"_CNT

3 S ZX=\$\$PROCESS^XPDAPUTL(ZN)

Q

4 ;PATCH INSTALL

N ZDATA,ZIEN,ZNAME,ZPTCH,ZSITE,ZX,ZZIEN

S ZDATA=\$P(\$P(\$G(^XPD(9.79,ZN,0)), "^",4),"~~#~~",2)

S ZSITE=\$P(ZDATA,"~",3)

S ZPTCH=\$P(ZDATA,"~",4)

S ZNAME=\$S(ZPTCH?'1.5U1""1.3N1"."1.2N1""1.4N:\$\$NAME^XPDAPUTL(ZPTCH),1:ZPTCH)

S ZIEN=\$0(^XPD(9.76,"C",ZNAME,"")) I ZIEN']"" D

.;S DIC(0)="LM",DIC=405375,X=ZPTCH D ^DIC

.S ZIEN=\$\$MSG^XPDAPUTL(ZPTCH)

.S DIC(0)="LM",DIC=9.76,X="`"_(ZIEN) D ^DIC S ZIEN=+Y

.S DIE=9.76,DA=ZIEN,DR=".5///^S X=ZNAME;1///^S X=DUZ;2///NOW" D ^DIE

```

I $P(ZDATA,"~",3)=$SSSITE^XPDAPUTL()&($P($G(^XPD(9.77,1,0)), "^",2)="S") D ;MESSAGE COMING IN FROM
MY MASTER (100)
.I $P($G(^DIC(4.2,$P($G(^XPD(9.77,1,0)), "^",4),0)), "^",1)'=$P(ZDATA,"~",2) W $
$SEND^XPDAMSTR($P(ZDATA,"~",2),$P(ZDATA,"~",3),0,"YOU ARE NOT THE MASTER SITE OF THIS DOMAIN","",0)
Q
.S: '$D(^XPD(9.76,ZIEN,100,0)) ^XPD(9.76,ZIEN,100,0)=""^9.7601^0^0^"_DT_"^^"
.S ZX=$$UPLOAD^XPDAPUTL(ZSITE,ZIEN,"100^0")
.S ZX=$$PROCESS^XPDAPUTL(ZN)
I $P(ZDATA,"~",2)=$SSSITE^XPDAPUTL()&($P($G(^XPD(9.77,1,0)), "^",2)="S") D ;MESSAGE GOING BACK TO
MY MASTER (125)
.S: '$D(^XPD(9.76,ZIEN,125,0)) ^XPD(9.76,ZIEN,125,0)=""^0^0^"_DT_"^^"
.S ZX=$$UPLOAD^XPDAPUTL(ZSITE,ZIEN,"125^0")
.S ZX=$$PROCESS^XPDAPUTL(ZN)
I $P(ZDATA,"~",2)=$SSSITE^XPDAPUTL()&($P($G(^XPD(9.77,1,0)), "^",2)="M") D ;MESSAGE GOING TO ONE
OF MY SERVANTS (150)
.S ZZIEN=$O(^XPD(9.76,ZIEN,150,"B",$DOMNUM^XPDAPUTL(ZSITE),"")) D:ZZIEN']""
.S DIC(0)="LM",DIC=""^XPD(9.76,"_ZIEN_",150,"",DA(1)=ZIEN,X=ZSITE D ^DIC S ZZIEN+=Y
.Q:ZZIEN<0
.S: '$D(^XPD(9.76,ZIEN,150,ZZIEN,1,0)) ^XPD(9.76,ZIEN,150,ZZIEN,1,0)=""^9.7621^0^0^"_DT_"^^"
.S ZX=$$UPLOAD^XPDAPUTL(ZSITE,ZIEN,"150^"_ZZIEN)
.S ZX=$$PROCESS^XPDAPUTL(ZN)
I $P(ZDATA,"~",3)=$SSSITE^XPDAPUTL()&($P($G(^XPD(9.77,1,0)), "^",2)="M") D ;MESSAGE COMING BACK
FROM ONE OF MY SERVANTS (200)
.S ZSITE=$P(ZDATA,"~",2)
.S ZZIEN=$O(^XPD(9.76,ZIEN,200,"B",$DOMNUM^XPDAPUTL(ZSITE),"")) D:ZZIEN']""
.S DIC(0)="LM",DIC=""^XPD(9.76,"_ZIEN_",200,"",DA(1)=ZIEN,X=ZSITE D ^DIC S ZZIEN+=Y
.Q:ZZIEN<0
.S: '$D(^XPD(9.76,ZIEN,200,ZZIEN,1,0)) ^XPD(9.76,ZIEN,200,ZZIEN,1,0)=""^9.7631^0^0^"_DT_"^^"
.S ZX=$$UPLOAD^XPDAPUTL(ZSITE,ZIEN,"200^"_ZZIEN)
.I '$D(^XPD(9.76,ZIEN,200,ZZIEN,0)) D REPAIR^XPDAPUTL(ZSITE,ZZIEN)
.S ZX=$$PROCESS^XPDAPUTL(ZN)
Q
5 ;PATCH STATUS MESSAGE
N HDR,ZDATA,ZIEN,ZNAME,ZSITE,ZX,ZZIEN
S ZDATA=$P($P($G(^XPD(9.79,ZN,0)), "^",4), "~#~",2)
S ZSITE=$P(ZDATA,"~",3)
I ZSITE=$SSSITE^XPDAPUTL()&($P($G(^XPD(9.77,1,0)), "^",2)="S") D ;coming to me and i am the
servant
.D STATUS^XPDAPROC(0)
.S HDR=~#~ PATCH STATUS RESPONSE FROM A SERVANT SITE"
.S ZX=$$STDBUL^XPDAPMES(5,HDR_""; "_ZSITE_" ; "_$P(ZDATA,"~",2))
.K ZTXT
.S ZX=$$PROCESS^XPDAPUTL(ZN)
I ZSITE=$SSSITE^XPDAPUTL()&($P($G(^XPD(9.77,1,0)), "^",2)="M") D ;coming to me and i am the master
.S ZX=$$PROCESS^XPDAPUTL(ZN)
I $P($G(^XPD(9.79,ZN,0)), "^",3)=5,$D(^XPD(9.79,"XPD",+$P($G(^XPD(9.79,ZN,0)), "^",1),ZN)) K
^XPD(9.79,"XPD",+$P($G(^XPD(9.79,ZN,0)), "^",1),ZN)
Q
6 ;MARK A PATCH AS INFORMATIONAL AT THE SERVANT SITE
N PATCH
S ZDATA=$P($P($G(^XPD(9.79,ZN,0)), "^",4), "~#~",2)
S PATCH=$P(ZDATA,"~",4),PATCH=$P(PATCH,"*",1,2)"*_ "$P(PATCH,"SEQ ",2)
S ^XPD(9.76,"E",PATCH,9999)=""
S ZX=$$PROCESS^XPDAPUTL(ZN)
Q
99 ;STARTER MESSAGE
S ZX=$$PROCESS^XPDAPUTL(ZN)
Q

XPDAPUTL
XPDAPUTL ;REMOTE PATCHING UTILITIES;DAF;3/29/04 ;05/04/2005 17395.007954
; ;8.0;KERNEL;**345**;Jul 10, 1995;Build 1
Q
CHECK(TTPE,TVAR) ;CHECK TO MAKE SURE THIS TYPE HAS THE PROPER VARIABLE COUNT
Q $$($L(TVAR,"")=$P($G(^XPD(9.78,TTPE,1,0)), "^",4):1,1:0)
CHK ;CHECK FOR EXISTENCE OF 9.77

```

```

Q:$D(^XPD(9.77,1))
S DIC=9.77,X=$$SSITE(),DIC(0)="LM" D ^DIC Q
CHKMST() ;GET NEXT MASTER NUMBERS
I '$D(^XPD(9.77,1,1,0)) S ^XPD(9.77,1,1,0)="9.7705P^0^0"
Q $P(^XPD(9.77,1,1,0),"^",3,4)
DIRP(FILE,ASK) ;DIR POINTER UTILITY
N DIR,DIRUT,Y
S DIR(0)="P^"_FILE_:QEM",DIR("A")=ASK D ^DIR Q $$($D(DIRUT):-1,1:Y)
DIRA(ASK,DFLT) ;DIR POINTER UTILITY
N DIR,Y
S DIR(0)="Y^",DIR("A")=ASK S:DFLT]" " DIR("B")=DFLT D ^DIR Q $$($D(DIRUT):-1,1:Y)
;
DISPKG ;Make "Disable" (NEW fld# 24) as default for every pkg in file 9.4 *p345
N DIE,DA,DR,FN,IEN,X
S FN="" F S FN=$D(^DIC(9.4,"B",FN)) Q:FN="" D
.S IEN=0 F S IEN=$D(^DIC(9.4,"B",FN,IEN)) Q:IEN="" D
..S X=$$GET1^DIQ(9.4,IEN,24) Q:X'=""
..S DIE=9.4,DA=IEN,DR="24///0" D ^DIE
Q
DOMAIN(DOMAIN) ;GO FROM DOMAIN NAME TO INSTITUTION NUMBER
Q $P($G(^DIC(4.2,+D(^DIC(4.2,"B",DOMAIN,"")),0)), "^",13)
DOMNUM(DOMAIN) ;GET THIS SITE'S IEN FOR THE DOMAIN
Q $D(^DIC(4.2,"B",DOMAIN,""))
FIXIT ;PLUG IN PATCH AND SEQUENCE NUMBER INTO FIELD 3 OF 9.76
N ZIEN,ZNM,ZNUM
S ZNM="" F S ZNM=$D(^XPD(9.76,ZNM)) Q:ZNM<1 D
.Q:$P($G(^XPD(9.76,ZNM,0)), "^",3)]" "
.S ZNUM=$P($G(^XPD(9.76,ZNM,0)), "^",1) Q:ZNUM' "]" "
.I ZNUM["DIZ(405375" D
..S ZIEN=$P($G(^DIZ(405375,+ZNUM,0)), "^",1)
.E S ZIEN+=ZNUM
.S DIE=9.76,DA=ZNM,DR="3///^S X=$$SEQ^XPDAPUTL(ZIEN)" D ^DIE
Q
FNDATE(IEN,ZIEN) ;FIND THE INSTALL DATE IN A MESSAGE
N ZNM,DATE S DATE=0
S ZNM="" F S ZNM=$D(^XPD(9.76,IEN,200,ZIEN,1,ZNM)) Q:ZNM' "]" " D
.S:$G(^XPD(9.76,IEN,200,ZIEN,1,ZNM,0))["Date and time of transaction: " DATE=$P(^0),"Date and
time of transaction: ",2)
Q $$DATE]" ":DATE,1:"NOT INSTALLED YET")
INIT ;INITIALIZE FILES
N ZX
S ZX=$$START,ZX=$$STOP
D DISPKG
Q
LST853() ;FIND LAST 9.79 ENTRY
N ZNM
S ZNM=$P($G(^XPD(9.77,1,100)), "^",1) S:'ZNM ZNM=$D(^XPD(9.79,"B","), -1)
Q $$ZNM:ZNM,1:0)
MASTER(DOMAIN) ;SET UP A MASTER FOR THIS CLIENT
N DIE,DA,DR
S DIE=9.77,DA=1,DR="1///S;2///^S X=$$DOMAIN^XPDAPUTL(DOMAIN);3///^S X=DOMAIN;4///NOW;4.5;4.6" D
^DIE Q 1
MSG(NAME) ;GET MESSAGE WITH EXACT NAME EVEN THOUGH IT IS TRUNCATED AT 30 CHARS
N RSLT,ZNM
S RSLT=0,ZNM="" F S ZNM=$D(^XMB(3.9,"B",$(NAME,1,30),ZNM)) Q:ZNM' "]" " S:NAME=
$P($G(^XMB(3.9,ZNM,0)), "^",1) RSLT=ZNM
Q RSLT
NAME(ZNM) ;CLEAN UP PATCH NAME
N NAME
S NAME=$P(ZNM,"Released ",2),NAME=$P(NAME," SEQ",1),NAME=$P(NAME," ",1)
S:$P(NAME,"*",2)'["."&(NAME'["DBA") $P(NAME,"*",2)=$P(NAME,"*",2)_" ".0"
Q NAME
POPUL(NUM) ;ADD ENTRY TO FILE 9.79
N DIC,X,Y
S DIC=9.79,DIC(0)="ML",X=NUM D ^DIC
Q Y

```

```

PROCESS(NUM)      ;PROCESS THE PROCESSES FIELD
N DIE,DA,DR
S DIE=9.79,DA=NUM,DR="1///1;1.5///NOW" D ^DIE Q 1
PLUG(IEN,TPE)      ;COME UP WITH PATCH NAME OR MESSAGE # FOR USE IN XREF OR JUST TO GET MESSAGE #
N ZNM,ZNAM,ZNUM
S ZNM=$P(^XPD(9.76,IEN,0),"^",1)
I ZNM["DIZ(405375" S IEN=$P(^DIZ(405375,$P(ZNM,";",1),0),"^",1)
E S IEN+=ZNM
S ZNAM=$P($G(^XMB(3.9,IEN,0)),"^",1) D:ZNAM']""
.S ZNUM=$O(^XMB(3.9,"B",$P(^DIZ(405375,$P(ZNM,";",1),0),"^",2),"")) S ZNAM=$P($
G(^XMB(3.9,+ZNUM,0)),"^",1) D:ZNAM']""
..S ZNUM=$O(^XMB(3.9,"B",$E("Copy of: "_$P(^DIZ(405375,$P(ZNM,";",1),0),"^",2),1,30),"")) S
ZNAM=$P($G(^XMB(3.9,+ZNUM,0)),"^",1)
Q $$ (ZNAM)"" :$(TPE:ZNAM,1:$NAME(ZNAM)),1:"")
PTYPE(IEN)      ;FIND OUT WHAT KIND OF PATCH THIS IS
N ZNAM,ZNM,ZZ S ZZ=""
S ZNM="" F S ZNM=$O(^XMB(3.9,IEN,2,ZNM)) Q:ZNM']"" D
.S ZNAM=$G(^XMB(3.9,IEN,2,ZNM,0))
.I ZNAM["X-MM-Type: X" S ZZ=0
.I ZNAM["X-MM-Type: K" S ZZ=1
.I ZNAM["X-MM-Type: K" S ZZ=1
.I ZNAM["No routines included" S ZZ=0
Q +ZZ
REPAIR(ZSITE,ZZIEN) ;REPAIR BAD 200 NODES
S ^XPD(9.76,ZIEN,200,0)="^9.7602P^1^1"
S ^XPD(9.76,ZIEN,200,ZZIEN,0)=$O(^DIC(4.2,"B",ZSITE,""))
Q
RSITE(SITE)      ;FIND THE DOMAIN FOR A SITE
Q:SITE["domain.ext" SITE
S SITE=$(SITE=xxx!(SITE=xxxx):"YOUR.SITE.DOMAIN.EXT",SITE=xxxx!(SITE=9999):"IMG-YOUR-
SITE.DOMAIN.EXT",1:SITE)
I SITE["XXXXX-XXXXX" Q SITE
E Q $P(^DIC(4.2,$P($G(^DIC(4,+SITE,6)),"^",1),0),"^",1)
SCNHDR(STMNT)    ;SCREEN HEADER
N LINE,SPACE,WORDS
;p345 - Join SET not work with DSM
;S ($P(LINE,"*",60),$P(SPACE," ",10))=""
S $P(LINE,"*",60)="", $P(SPACE," ",10)=""
S WORDS="WELCOME TO THE REMOTE PATCHING PACKAGE"
W @IOF,!,SPACE,LINE,!,SPACE,SPACE,WORDS,!,SPACE,SPACE,STMNT,!,SPACE,LINE Q ""
SEQ(IEN)          ;OUTPUT PATCH NAME IN SEQUENCE ORDER
N ZNAME
S ZNAME=$P($P(^XMB(3.9,IEN,0),"^",1),"Released ",2)
I $P(ZNAME,"*",2) ['." S $P(ZNAME,"*",2)=$P(ZNAME,"*",2)_" ".0"
Q $P(ZNAME,"*",1,2)_"*"_"$P(ZNAME,"#",2)
SSITE() ;FIND THIS SITE'S DOMAIN
Q $P(^DIC(4.2,$P(^XTV(8989.3,1,0),"^",1),0),"^",1)
START() ;SET FILES UP IF NOT DONE YET
I 'SD(^XPD(9.77,1)) S DIC=9.77,DIC(0)="LMZ",X=$P(^DIC(4.2,$P(^XTV(8989.3,1,0),"^",1),0),"^",13) D
^DIC
S TMP=$O(^XMB(3.8,"B","XPD VISTA PATCH","")) I $P($G(^XMB(3.8,TMP,1,0)),"^",4)<1&($D(DUZ)) D
.S $P(^XMB(3.8,TMP,1,1,0),"^",1)=DUZ
.S $P(^XMB(3.8,TMP,1,0),"^",1,4)="^3.81P^1^1"
.S:$P($G(^XMB(3.8,TMP,3)),"^",1)']"" $P(^3),"^",1)=DUZ
Q $$$STOP
STOP() ;STOP SET
N STOP,TMP,ZX
S STOP=$LST853 Q:STOP STOP
S ZX=$$TEST^XPDAPUTL() Q:$POPUL^XPDAPUTL(ZX)']"" "" S DIE=9.79,DA=1,DR="1///1;2///99;2///START"
D ^DIE
S STOP=($P($G(^XMB(3.9,0)),"^",3)-1000)
Q $(STOP>0:STOP,1:0)
TEST() ;SEND TEST MESSAGE TO POPULATE XPD MESSAGE FILE
N HDR,SSITE
S SSITE=$P(^DIC(4.2,$P(^XTV(8989.3,1,0),"^",1),0),"^",13)
S HDR="THIS IS A TEST OF THE VISTA PATCHING MODULE"

```

```

Q $$STDBUL^XPDAPMES(99,HDR_" ; "_SSITE)
UPLOAD(SITE,IEN,TPE)      ;UPLOAD TEXT INTO A 9.76 ENTRY ZTXT ARRAY MUST BE SET
;IF TPE IS 150 OR 200 WILL BE TWO PIECE WITH 2ND PIECE BEING SUBNODE
N CNT,LST,START,ZIEN,ZNM
I +TPE[100! (+TPE[125) D
.S ZNM="" F S ZNM=$0(ZTXT(ZNM)) Q:ZNM']"" D
..S LST=($P(^XPD(9.76,IEN,+TPE,0),"^",3)+1),CNT=($P(^ (0),"^",4)+1)
..S ^XPD(9.76,IEN,+TPE,LST,0)=ZTXT(ZNM)
..S $P(^XPD(9.76,IEN,+TPE,0),"^",3,4)=LST_"^"_CNT
.S:+TPE[100&(IEN>0) ^XPD(9.76,"ZZ",IEN)=""
.K:+TPE[125 ^XPD(9.76,"ZZ",IEN) E D
.S ZIEN=$P(TPE,"^",2),START=0
.S ZNM="" F S ZNM=$0(ZTXT(ZNM)) Q:ZNM']"" D
..S:ZTXT(ZNM)[ ""#5"" START=1 Q:'START
..Q:ZTXT(ZNM)=""
..S LST=($P(^XPD(9.76,IEN,+TPE,ZIEN,1,0),"^",3)+1),CNT=($P(^ (0),"^",4)+1)
..S ^XPD(9.76,IEN,+TPE,ZIEN,1,LST,0)=ZTXT(ZNM)
..S $P(^XPD(9.76,IEN,+TPE,ZIEN,1,0),"^",3,4)=LST_"^"_CNT
K ZTXT
Q 1

```

XPDAUTL

XPDAUTL ;CAVHCS-619/TEA - AUTOPATCH CONTROL ROUTINE ;05/04/2004 09:38

;8.0;KERNEL;**345**;Jul 10, 1995;Build 1

SITE ;EDIT VAPU SITE PARAMETERS

S DIE="^XPD(9.72," ,DA=1,DR=".01;4;6;2;3;7;8;9;10" D ^DIE

G EXIT^XPDAUTO2

EXE ;EDIT EXECUTABLE DIRECTORY

S DIE="^XPD(9.72," ,DA=1,DR="11" D ^DIE

G EXIT^XPDAUTO2

MSERV ;EDIT G.PATCHES REMOTE MEMBERS

S DIC="^XMB(3.8," ,DIC(0)="M",X="PATCHES" D ^DIC D:+Y<1 Q:'+Y S DA=+Y,DIE=DIC,DR="12" W

!,"EDITING PATCHES Mail Group - REMOTE MEMBERS: " D ^DIE Q

.W !!!!!,"You MUST HAVE a G.PATCHES Mail Group for AUTOPATCHING to work!!!!",!,"Please create a G.PATCHES Mail Group before continuing!!!!"

G EXIT^XPDAUTO2

PKMOD ;EDIT PACKAGE MODIFIERS

S (DIE,DIC)="^XPD(9.75," ,DIC(0)="AEQMZL" D ^DIC Q:+Y<1 S DA=+Y,DR=".01;1;2;3;4;5;6;7;8;11" D ^DIE

K Y,DIC,DIE,DR,DA

G EXIT^XPDAUTO2

PKENABL ;ENABLE/DISABLE PACKAGE FOR AUTOINSTALL

S (DIE,DIC)="^DIC(9.4," ,DIC(0)="AEQMZ" D ^DIC Q:+Y<1 S DA=+Y,DR="24" D ^DIE

K Y,DIC,DIE,DR,DA

G EXIT^XPDAUTO2

SENDMOD ;RETRIEVE INSTALL MODIFIERS FROM VAPU INSTALL MODIFIERS (9.74) AND CREATE MAIL MESSAGE

S U=""

S DIC="^XPD(9.74," ,DIC(0)="AEQMZ" D ^DIC Q:+Y<0 S XPDAIX=+Y,XPDATMP=\$G(^XPD(9.74,XPDAIX,0))

;ENCAPSULATE IN MAIL

S XMSUB="##MODIFY INSTALL##" _\$P(\$G(XPDATMP),U)_"##" _\$P(\$G(XPDATMP),U,11),XMTEXT="TEXT("

S TEXT(1)="##ROOT##" _XPDATMP

S TEXT(2)="OPTIONS/PROTOCOLS"

S TEXT(3)="##"

S I=4 F J=1:1 S TEXT(I+J)=\$G(^XPD(9.74,XPDAIX,3,J,0)) Q:TEXT(I+J)=""

S I=(I+J) F J=1:1 S TEXT(I+J)=\$G(^XPD(9.74,XPDAIX,2,J,0)) Q:TEXT(I+J)=""

S TEXT(I+J)="##"

D ^XMD S XPDAXMZ=XMZ

G EXIT^XPDAUTO2

UPIN ;UPDATE THE INSTALL FILE

D TEST S XPDANM=\$G(X1),XPDASEQ=\$G(X2) K X1,X2

S X=XPDANM,(DIC,DIE)="^XPD(9.74," ,DIC(0)="ML" D ^DIC S DA=+Y,DR=".02///^S X=XPDASEQ;9///1" D ^DIE

S XQAID="AND1;" _DUZ_" ;" _\$P(\$G(%)," ",2),XQADATA=\$G(XPDANM)_U_\$G(XPDASEQ)_U_\$G(X

MZ)_U_\$G(XPDFTNM)_U_"TEST",XQAROU="MAN^XPDAUTL",XQA("G.PATCHES")="",XQAMSG="Install "_\$G(XPDANM)_

MM: " _\$G(XMZ)" Must be INSTALLED MANUALLY " D SETUP^XQALERT

G EXIT^XPDAUTO2

MAN ;PROCESS MANUAL ALERTS OR INSTALL MANUALLY FROM MENU

S XPDACHK=1,XQAKILL=0


```

S DIR("A")="Would you like to MANUALLY INSTALL NOW",DIR("B")="Y",DIR(0)="Y" D ^DIR Q:Y'=1 ;G
^XPDAUTOM
;GET NECESSARY INFORMATION FROM FILE
;p345-split line because over 245 char.
I $L($G(XQAID))>1 S XPDAXQID=XQAID,XPDAXQDT=XQADATA,(XPDANM,X)=$P($G(XQADATA),U
),XMZ=$P($G(XQADATA),U,3) S DIC="^XPD(9.74," ,DIC(0)="M"
I $L($G(XQAID))>1 D ^DIC D:+Y<1 Q:+Y<1 S (XPDAA,DA)=+Y,DIE="^XPD(9.74," ,DR="
.01;.02;9;1;2;3;4;5;11;7;8;6;10//0" D:XMZ VDOC D ^DIE
.;THERE WAS NOT AN INSTALL MODIFIER - LET'S CREATE ONE
.S (DIE,DIC)="^XPD(9.74," ,DIC(0)="L" D ^DIC S (XPDAA,DA)=+Y,XPDASEQ=$P($G(XQADA
TA),U,2),XPDFTNM=$P($G(XQADATA),U,4),DR=".02///^S X=XPDASEQ;12///^S X=XMZ;6///^S X=XPDFTNM" D
^DIE S Y=XPDAA
;
I $L($G(XQAID))<1 S DIC="^XPD(9.74," ,DIC(0)="AMQE" D ^DIC Q:+Y<1 S XMZ=$P(^XPD
(9.74,+Y,0),U,12),(XPDAA,DA)=+Y D:XMZ VDOC S DIE="^XPD(9.74," ,DR=".01;.02;9;1;2;
3;4;5;11;7;8;6;10//0" D ^DIE
S XMR=$P(^XMB(3.9,XMZ,0),U)
S XPDTEMP=$G(^XPD(9.74,XPDAA,0)),(XMZ,XPDAXMZ)=$P(XPDTEMP,U,12),XPDANM=$P(XPD
ATEMP,U),XPDASEQ=$P(XPDTEMP,U,11),XPDFTNM=$P(XPDTEMP,U,12)
D HOME^%ZIS S XPDDEV=IO ;I $P($G(^XPD(9.74,XPDAA,0)),U,12)
;E S XPDFTNM=$P($G(^XPD(9.74,XPDAA,0)),U,7)
D GETANS^XPDAUTOM
G XI^XPDAUTOM
VDOC ;VIEW KIDS MailMan documentation
S XQAKILL=0
W !!!!!, " PATCH DOCUMENTATION RETRIEVAL"
S DIR(0)="Y",DIR("B")="N",DIR("A")="Would you like to PRINT the PATCH Document?" D ^DIR I Y=1 D ;
.S:$D(XQADATA) XMZ=$P(XQADATA,U,3) I XMZ D XMPRT^XPDUTL(XMZ)
Q:$G(XPDACHK)
S XQAKILL=0,DIR(0)="Y",DIR("B")="Y",DIR("A")="Would you like to KEEP this ALERT for future
Document Printing?" D ^DIR I Y=0 D ;
.S XQAKILL=1 D DELETE^XQALERT K Y,DIR
G:$G(XPDACHK) EXIT^XPDUTL2
Q
TEST ;GET THE INSTALL NAME FROM MAIL MESSAGE
S X=XMB(3.9,XMZ,0) ;X1=$P(X," ",2)
S X1=$P(X,"Released ",2),X1=$P(X1," SEQ",1) ;DAF;CHANGED TO COVER COPIEDMESSAGES
I X1["*" S X1=$P(X," ",3) I X1["*" S X1=$P(X1," ")
S XPDAREC=$P($P(^XMB(3.9,XMZ,0),U,3)," ",1,3)
S X2=$P(X,"#",2) F I=1:1:$L(X2) I $E(X2,I) Q
I X1["." S $P(X1,"*",2)=$P(X1,"*",2)_"0"
S X2=$E(X2,I,$L(X2)),X2+=X2
Q
INFORM ;CHECK FOR INFORMATIONAL PATCH UPDATE INSTALL FILE
S II=0 F S II=$O(^XMB(3.9,XMZ,2,II)) Q:+II=0 S XPDAX0=$G(^XMB(3.9,XMZ,2,II,0)
),XPDAX0=$S(XPDAX0["INFORMATIONAL":6,XPDAX0["Informational":6,XPDAX0["DBA":7,XPDAX0["Database":7,XPD
AX0["Other":8,1:0) I $G(XPDAX0) D Q
.D TEST^XPDUTL S XPDANM=$G(X1),XPDASEQ=$G(X2) K X1,X2
.S DUZ(0)="@"
.K DIC S X=XPDANM,XPDASEQ="SEQ #_"_XPDASEQ,DIC="^XPD(9.7," ,DIC(0)="LX",DIC("DR")=".01///^S
X=XPDANM;6///^S X=XPDASEQ;.02///^S X=XPDAX0" D ^DIC K DIC,Y
.;SEND AN ALERT FOR DOCUMENTATION PRINTING
.D DOCXQA
G EXIT^XPDUTL
DOCXQA ;SEND DOCUMENTATION ALERT
D ;SEND ALERT FOR DOCUMENTATION RETRIEVAL TO RECIPIENTS LISTED IN FILE 9.75
S XPDAPK=$P($G(XPDANM),"*"),XPDAPK=$O(^DIC(9.4,"C",XPDAPK,0)),XPDAX1=$O(^XPD(9.75,"B",XPDAPK,0))
Q:'XPDAX1
.S I=0 F S I=$O(^XPD(9.75,XPDAX1,3,"B",I)) Q:+I=0 S XQA(I)=" "
.K I D DT^DICRW S XQA("G.PATCHES")=" ",XQAID="XPDAX1;DUZ;"_P($G(%)," ",2),P(XQA
DATA,U,3)=$G(XMZ),XQAROU="VDOC^XPDUTL",XQAMSG="DOCS FOR: "_P($G(^XMB(3.9,$G(XMZ),0),U) D
SETUP^XQALERT
Q
ADPK ;CHECK TO SEE IF FILE 9.75 HAS AN ENTRY FOR THIS PATCH'S PACKAGE
S (A1,XPDAPKNM)=$$PKG^XPDUTL(XPDANM),XPDAPKNM=$O(^DIC(9.4,"C",XPDAPKNM,0)),X=$O
(^XPD(9.75,"B",XPDAPKNM,0)) I X="" D ;

```

```

.;
.K DIC S DIC="^XPD(9.75," ,DIC(0)="MNL",X="`" _XPDAPKNM D ^DIC Q:+Y<1
.S X="PATCH,ALPHA T",DIC="^XPD(9.75," _+Y_" ,3," ,DA(1)=+Y,DIC(0)="LBNX0",DIC("P")
="$P(^DD(9.75,12,0),U,2)" D ^DIC
Q
ADRECIP ;ADD PATCH DOCUMENTATION RECIPIENTS
K DIC S DIC="^XPD(9.75," ,DIC(0)="LMAEQ" D ^DIC Q:+Y<1
S DIE="^XPD(9.75," ,DR="12",DA=+Y D ^DIE
K DIE,DA,DR,Y
Q
XMPRT(XPDAXMZ1) ;PRINT A MAIL MESSAGE
N DUZ,XMZ,IO,XMK,XMDUZ,XMINSTR
S (XMDUZ,DUZ)=$P(^XPD(9.72,1,0),U,7),XMZ=XPDAXMZ1,XMINSTR="K"
S XMK=$$BSKT^XMXUTIL2(DUZ,XMZ)
I XMK=0 U IO W !!!!,"MESSAGE NO LONGER EXISTS IN PATCH USER'S BASKET!!!!",!!!,"
SORRY??? )-: "
D INIT^XMJMOI1(XMDUZ,XMZ,.XMSUBJ,.XMFROM,.XMORIGN8,.XMINSTR,.XMRESTR,.XMIEN,.XMPRESPSO,.XMPRESP)
S XMV("NAME")=""
D PRINT^XMJMOI(1,0)
Q

XPDATL2
XPDATL2 ;;CAVHCS-619/TEA - AUTOPATCH CONTROL ROUTINE;08/29/03 07:35 ;05/03/200
5 454142.330749
;;8.0;KERNEL;**345**;Jul 10, 1995;Build 1
QUEL ;MANAGE PATCH USER INSTALL QUEUE
N DUZ
S DUZ=$P(^XPD(9.72,1,0),U,7)
;D ^XUTMQ
;Q
;DO IT BETTER
K ^TMP("XPD")
;Add $G -*p345
F I=0:0 S I=$O(^%ZTSK(I)) Q:+I=0 I $P($G(^%ZTSK(I,0)),U,3)=DUZ,$G(^%ZTSK(I,.3,"XPDADT")) D ;
.S $E(XPDAAT,1)=^%ZTSK(I,.3,"XPDANM"),$E(XPDAAT,14)=I,$E(XPDAAT,24)=^%ZTSK(I,.3
,"XPDAXMZ"),$E(XPDAAT,32)=$E(^%ZTSK(I,.3,"XPDASUB1"),1,15) D ;PARSE DESCRIPTION
.S Y=^%ZTSK(I,.3,"XPDACOMP") X ^DD("DD") S $E(XPDAAT,50)=Y,%H=^%ZTSK(I,.3,"XPDADT") D YX^%DTC S
$E(XPDAAT,63)=Y
.S ^TMP("XPD",$J,XPDAAT)="",XPDAAT=""
.Q
W !!! S I="" F S I=$O(^TMP("XPD",$J,I)) Q:$L(I)'>7 W !,I
K ^TMP("XPD") Q
QUER ;REQUE TASKS
N DUZ
S DUZ=$P(^XPD(9.72,1,0),U,7)
D ^XUTMR
Q
QUED ;DELETE TASKS
N DUZ
S DUZ=$P(^XPD(9.72,1,0),U,7)
D ^XUTMD
Q
MBSKT ;MANAGE PATCH USER MAIL BASKETS
W !!!,"UNDER DEVELOPMENT!"
Q
MKBAT ;CREATE CACHE/NT BATCH FILE
S XPDAEXNM=$P($G(^XPD(9.72,1,1)),U) I $L(XPDAEXNM)>3 S X=$E(XPDAEXNM,$L(XPDAEXNM),$L(XPDAEXNM)) I
X'="" S XPDAEXNM=XPDAEXNM_"\"
S XPDAFTDS=$P($G(^XPD(9.72,1,0)),U,9),XPDAFTDS=$P($G(^XPD(9.73,XPDAFTDS,0)),U)
D OPEN^%ZISH("XPDAFTP",XPDAEXNM,"XPDAFTP.BAT","W",300) G:POP MKBAT S XPDDEV=IO U XPDDEV
U IO W "ftp -v -i -s:XPDSRIPT.TXT "_XPDAFTDS
D CLOSE^%ZISH("XPDAFTP")
H 3 W !!!,"Okay I'm done!!" H 2
Q
TRACK ;
;

```

```
;Parse Mail Message for:
;Patch #, Subject, Priority, Recpt Date, Compliance Date
;
S XMREC="D REC^XMS3"
F D I $G(XPDASUB1)]["$G(XPDACOMP)]["$G(XPDAPRI)]" Q ;LOOP TIL VARIABLES FOUND
.X XMREC
.I XMRG["Priority: " S XPDAPRI=$P(XMRG,"Priority: ",2)
.I XMRG["Compliance Date: " S XPDACOMP=$P(XMRG,"Compliance Date: ",2)
.I XMRG["Subject: " S XPDASUB1=$P(XMRG,"Subject: ",2)
.I XMRG["Run Date: " S RNDATE=$P($P($P(XMRG,"Run Date: ",2)," ",1)," ",2) I
RNDATE<2004 S XPDACOMP=DT K RNDATE ;DAF;STOP LOOPING FOR OLD PATCHES
Q
MON ;MONITOR
;LOOP THROUGH THE "B" X-REF OF FILE 619070 XPDA0 COMPARE TO INSTALL FILE FOR PATCHES
;THAT HAVE NOT BEEN INSTALLED. If the patch has an install status of completed in
;the INSTALL File this "AC" & "ACD" X-RefS are deleted.
N XPDAX,XPDA0,XPDAHD,XPDAHD2,I,J
S XPDAX1=4 K TEXT
S I=" F S I=$0(^XPD(9.74,"B",I)) Q:I=" I $D(^XPD(9.7,"B",I)) S J=" J=$0(^XPD(9.7,"B",I,J)) D
.I $P(^XPD(9.7,J,0),U,9)=3 K ^XPD(9.74,"AC",I),^XPD(9.74,"ACD",I)
.S K=$G(^XPD(9.7,J,1)) I $P(^XPD(9.7,J,0),U,9)>4,$P(^XPD(9.7,J,0),U,11),$P(K,U,3) K
^XPD(9.74,"AC",I),^XPD(9.74,"ACD",I)
.I $P(^XPD(9.7,J,0),U,9)<3 S ^XPD(9.74,"AC",I)="
.I $P(^XPD(9.7,J,0),U,9)>4,'$P(^XPD(9.7,J,0),U,11),'$P(K,U,3) S ^XPD(9.74,"AC",I)="
.Q
;FIND DELINQUENT PATCHES FIRST
;WILL NEED TO SET ACD X-REF
D NOW^%DTC S XPDADT=X
S I=" F S I=$0(^XPD(9.74,"AC",I)) Q:I=" D ;ADD UNINSTALLED PATCHES TO TEXT()
.S J=" F S J=$0(^XPD(9.74,"B",I,J)) Q:+J=0 D
..S XPDAX=$G(^XPD(9.74,J,4))
..I XPDADT>$P(XPDAX,U,2) S ^XPD(9.74,"ACD",I)="
..I XPDADT>$P(XPDAX,U,2) K ^XPD(9.74,"ACD",I)
..I '+$P(XPDAX,U,2) K ^XPD(9.74,"ACD",I)
S TEXT(1)="The following patches are not installed and COMPLIANCE date has passed!"
S TEXT(2)=" ",XPDAX2=0
S $E(XPDAHD,1)="Patch #",$E(XPDAHD,14)="Description",$E(XPDAHD,40)="Priority",$
E(XPDAHD,55)="Received",$E(XPDAHD,68)="Compliance"
S $E(XPDAHD2,1)="====",$E(XPDAHD2,14)="====",$E(XPDAHD2,40)="===="
", $E(XPDAHD2,55)="====",$E(XPDAHD2,68)="===="
S TEXT(3)=XPDAHD,TEXT(4)=XPDAHD2
S I=" F S I=$0(^XPD(9.74,"ACD",I)) Q:I=" D ;BUILD TEXT() FOR DELINQUENT PATCHES
.S J=" F S J=$0(^XPD(9.74,"B",I,J)) Q:+J=0 D
..S XPDAX=$G(^XPD(9.74,J,4))
..S Y=$P(XPDAX,U,2) I 'Y S XPDACOMP="NO INFO" K ^XPD(9.74,"ACD",I)
..I Y,$L(Y)=7 X ^DD("DD") S XPDACOMP=Y
..S Y=$P(XPDAX,U,4) I 'Y S XPDAREC="NO INFO"
..I Y,$L(Y)=7 X ^DD("DD") S XPDAREC=Y
..S XPDASUB1=$P(XPDAX,U,3),XPDAPRI=$P(XPDAX,U,5),XPDANM=$P(^XPD(9.74,J,0),U)
..S $E(XPDA0,1)=XPDANM,$E(XPDA0,14)=$E(XPDASUB1,1,20),$E(XPDA0,40)=XPDAPRI,$E(X
PDA0,55)=XPDAREC,$E(XPDA0,68)=XPDACOMP
..S TEXT(XPDAX1+1)=XPDA0,XPDAX1=XPDAX1+1,XPDA0="",XPDAX2=XPDAX2+1
F I=1:1:3 S TEXT(XPDAX1+I)=" "
I 'XPDAX2 S TEXT((XPDAX1+I+1))=" NO DELINQUENT PATCHES "
I XPDAX2 S TEXT((XPDAX1+I+1))=" COUNT: "(XPDAX2)
S XPDAX1=(XPDAX1+(I+2)) F I=1:1:3 S TEXT(XPDAX1+I)=" "
S XPDAX1=(XPDAX1+I),XPDA0=""
;NOW FIND UNINSTALLED PATCHES
S TEXT(XPDAX1+1)="The following patches are not installed."
S TEXT(XPDAX1+2)=" "
S TEXT(XPDAX1+3)=XPDAHD,TEXT(XPDAX1+4)=XPDAHD2,XPDAX1=XPDAX1+4,XPDAX2=0
S I=" F S I=$0(^XPD(9.74,"AC",I)) Q:I=" D ;ADD UNINSTALLED PATCHES TO TEXT()
.S J=" F S J=$0(^XPD(9.74,"B",I,J)) Q:+J=0 D
..S XPDAX=$G(^XPD(9.74,J,4))
..S Y=$P(XPDAX,U,2) I 'Y S XPDACOMP="NO INFO"
..I Y,$L(Y)=7 X ^DD("DD") S XPDACOMP=Y
```

```

..S Y=$P(XPDAX,U,4) I 'Y S XPDAREC="NO INFO"
..I Y,$L(Y)=7 X ^DD("DD") S XPDAREC=Y
..S XPDASUB1=$P(XPDAX,U,3),XPDAPRI=$P(XPDAX,U,5),XPDANM=$P(^XPD(9.74,J,0),U)
..S $E(XPDA0,1)=XPDANM,$E(XPDA0,14)=$E(XPDASUB1,1,20),$E(XPDA0,40)=XPDAPRI,$E(X
PDA0,55)=XPDAREC,$E(XPDA0,68)=XPDACOMP
..S TEXT(XPDAX1+1)=XPDA0,XPDAX1=XPDAX1+1,XPDA0="",XPDAX2=XPDAX2+1
F I=1:1:3 S TEXT(XPDAX1+I)=" "
I XPDAX2 S TEXT((XPDAX1+I+1))=" COUNT: "_(XPDAX2)
S XPDAX1=(XPDAX1+(I+2)) F I=1:1:3 S TEXT(XPDAX1+I)=" "
S XPDAX1=(XPDAX1+I),XPDA0=""
;I '$G(TEXT(XPDAX1+5)) S TEXT(XPDAX1+5)=" ",TEXT(XPDAX1+6)=" ",TEXT(XPDAX1+7)="
;I '$G(TEXT(XPDAX1+5)) S TEXT(XPDAX1)=" COUNT: " _XPDAX1-4
S XMY("G.XPD VAPU MONITOR")="",XMSUB="VAPU PATCH MONITOR REPORT.",XMTEXT="TEXT("
I '$G(XPDAINT) D ^XMD ;SEND MESSAGE OF DELINQUENT XPDA0 UNINSTALLED PATCHES
I '$G(XPDAINT) U IO S K=0 F S K=$O(TEXT(K)) Q:K="" W !,TEXT(K)
K XPDAX,XPDAX1,XPDAX2,XPDACOMP,XPDAHD,XPDAHD2,XPDAREC,XPDASUB1,I,J,TEXT,XPDAINT
Q
COMP ;COMPLETE PATCH INSTALL - Completion of patch installs is accomplished by documenting
;in the INSTALL file (9.7). Kids patches do this automatically but Informational, DBA, and other
;do not. When a patch is received, a record of receipt is stored in VAPU. This routine allows
;the patch installer to edit the required fields for patch completion.
K X S DIC(0)="ALE",DIE("NO^")="BACK",(DIC,DIE)=""^XPD(9.7,"",DR=".01;.02;6FILE COMMENT [Enter the
string SEQ #+Actual #][e.g. SEQ #30];9;17" D ^DIC Q:+Y<1 S DA=+Y D ^DIE
K X,Y,DIC,DIE,DA,DR
Q
DEL ;DELETES AN ENTRY FROM THE PATCH MONITOR. Occasionally a patch may be received that you don't
;want to track. For example you may receive a TEST patch that was sent to you
in error.
;This code will allow you to delete it out of the INSTALL MODIFIER file (9.74)so it won't
;be tracked by the patch monitor.
S (DIK,DIC)=""^XPD(9.74,"",DIC(0)="BQEA" D ^DIC Q:+Y<1 S DA=+Y W !!!!!,"You have
chosen to delete: " _$P(Y,U,2),!! S DIR(0)="Y",DIR("B")="NO",DIR("A")="Is this c
orrect? [YES/NO] " D ^DIR Q:'Y D ^DIK
K DIK,DIC,DA,Y,X
Q
INT ;INTERACTIVE PATCH DISPLAY
S XPDAINT=1
D ^%ZIS
G MON
Q

XPDAUTO
XPDAUTO ;CAVHCS-619/TEA - AUTOPATCH CONTROL ROUTINE ;05/04/2004 09:39
;;8.0;KERNEL;**345**;Jul 10, 1995;Build 1
;S XMZ=4974656 ; USED TO DEBUG
;GET PATCH INSTALLER'S DUZ
S (DUZ,XMDUZ)=$P($G(^XPD(9.72,1,0)),",^",7) Q:'DUZ S DUZ(0)="@
I $P($G(^XMB(3.9,XMZ,0)),",^")["##MODIFY INSTALL##" S XPDAXMZ=XMZ G MODIFY^XPDAUTO2
;TEST FOR FORUM PATCH MODULE - QUIT IF NOT
;I $P($G(^XMB(3.9,XMZ,0)),U,2)'("<"National Patch Module"@DOMAIN.EXT>") G
EXIT ;COMMENTING OUT;DAF;4/19/04
D TEST^XPDAUTL S XPDANM=$G(X1),XPDASEQ=$G(X2) I XPDANM["*" G EXIT
;IS THIS PACKAGE INSTALLED?
S (A1,XPDAPKNM)=$$PKG^XPDUtl(XPDANM),XPDAVER=$$VERSION^XPDUtl(A1) I XPDAVER=""
G VER
;CHECK FOR PACKAGE ENTRY IN 9.75 IF NOT ADD ONE
D ADPK^XPDAUTL
;TEST FOR KEYWORDS AND CHARACTERS THAT INDICATE THIS IS A PATCH
;p345-split line over limit.
I $P($G(^XMB(3.9,XMZ,0)),U) ["Released",$P($G(^XMB(3.9,XMZ,0)),U) ["*" D TRACK^XP
DAUTL2 S X=XPDANM,(DIC,DIE)=""^XPD(9.74,"",DIC(0)="MXL" D ^DIC Q:+Y<1 D
.S (XPDA0,DA)=+Y,DR=".02///^S X=XPDASEQ;12///^S X=XMZ;14///^S X=XPDACOMP;15///
^S X=XPDASUB1;17///^S X=XPDAPRI;16///^S X=XPDAREC" D ^DIE
;
D CHECK^XMP2A I Y<1 D G INFORM^XPDAUTL
.;SEND NOTICE OF RECEIPT AND WAITING ON INSTALL MOD

```

```

.S XQA("G.PATCHES")="",XQAMSG="MM: "_XMZ_" FOR "_XPANM_" VAPU NEEDS AN INSTALL
MODIFIER TO CONTINUE INSTALL! WAITING FOR MODIFIER." D SETUP^XQALERT
.D DOCXQA^XPDAUTL
;GET AUTOINSTALL PARAMETERS
S XPDASITE=$(^XPD(9.72,1,0)) I '$D(XPDASITE) S XQA("G.PATCHES")="",XQAMSG="File: 9.72-VAPU SITE PARAMETERS, has not been set up" D SETUP^XQALERT G EXIT
;CHECK TO SEE IF AUTOINSTALL IS ENABLED - QUIT IF NOT
I '$P(XPDASITE,"^",5) S XQAMSG="File: 9.72-VAPU SITE PARAMETERS, has not been set up" D SETUP^XQALERT G EXIT
;SET UP QUEUE TIME
S X=$P(XPDASITE,"^",4),X1="0000" S XPDATM=$E(X1,1,(4-$L(X)))_X ;TIME IN MILITARY
;SET UP QUEUE DATE
D NOW%DTC S X1=X,X2=$P(XPDASITE,"^",3) D C%DTC S X=X_"_"_XPDATM,XPDADT=$$FMTH^XLFD(X)
;S DUZ(0)="@",
S XPDAXMZ=XMZ,ZTRTN="QUE^XPDAUTO",ZTSAVE("XPD*")="",ZTSAVE("XM*")="",ZTDESC="AUTO PATCH for Message # "_XMZ_" Subject: "_$P($G(^XMB(3.9,XMZ,0)),("^)",ZTDTH=XPDA DT,ZTREQ="@",ZTIO="" D ^%ZTLOAD ;DAF;ADDED ZTIO TO STOP DEVICE ERROR
;SEND ALERT OF QUEUED INSTALL & SET TASK # IN 9.74
S:$D(^XPD(9.74,XPDA,0)) $P(^XPD(9.74,XPDA,4),U)=$G(ZTSK)
S XQAID="XPDA2",XQA("G.PATCHES")="",XQAMSG=XMZ_"_"_$P($G(^XMB(3.9,XMZ,0)),("^")_" TSK: "_$G(ZTSK)_" Has been queued for INSTALL." D SETUP^XQALERT
D DOCXQA^XPDAUTL
Q ;COMMENT OUT TO DEBUG, Also comment out the D ^%ZTLOAD ON THE 3RD LINE UP
QUE ;TASK ENTRY POINT
K XPDSTPXU,XPDASTPT,XPDASXWB,XPDASTPL,XPDASTPM,XPDINC,XPDATEMP,XPANM
D TEST^XPDAUTL S XPANM=$G(X1),XPDASEQ=$G(X2) I XPANM["*" G EXIT
S XMR=^XMB(3.9,XMZ,0),DUZ=XMDUZ,XPD00=0,XPDA DIR=$P($G(XPDASITE),U,6)
;IF COMING FROM FORUM THE MESSAGE IS TREATED IN THE NEXT LINE - NOTE SPACE BETWEEN : and K
S XPDACHK=0,II=0 F S II=$O(^XMB(3.9,XMZ,2,II)) Q:+II=0 I ^ (II,0)="X-MM-Type:
K" S XPDACHK=1 K II Q
I $G(XPDACHK) D CHECK^XMP2A I +Y<1 G UPIN^XPDAUTL
D GETANS^XPDAUTO2
;WHEN PATCH IS FORWARDED LOCALLY PATCH IS TREATED DIFFERENTLY- Note no space between : and K -SHOULD PROBABLY ADD ADDITIONAL CHECK FOR LOCAL
I '$G(XPDACHK) S XPDACHK=0,II=0 F S II=$O(^XMB(3.9,XMZ,2,II)) Q:+II=0 I ^ (II,0)="X-MM-Type:K" S XPDACHK=1 K II Q
I $G(XPDACHK) D CHECK^XMP2A I +Y<1 G UPIN^XPDAUTL
D GETANS^XPDAUTO2
S II=0 F S II=$O(^XMB(3.9,XMZ,2,II)) Q:+II=0 I ^XMB(3.9,XMZ,2,II,0)["**INSTALL NAME**" S XPANM=^XMB(3.9,XMZ,2,(II+1),0) K II Q
;##### CHECK FILE 9.74 FOR MANUAL INSTALLTION #####
S (DIE,DIC)="^XPD(9.74,"",DIC(0)="M",X=XPANM D ^DIC I +Y>0 S XPDAIX=+Y K Y D ;
.S XPDAXMZ=XMZ,DA=XPDAIX,DR="12//^S X=XPDAXMZ" D ^DIE,DT^DICRW
.I $P($G(^XPD(9.74,XPDAIX,0)),U,9) D G EXIT ;SEND ALERT AND EXIT
.S XQAID="XPDA1";"DUZ";"$P($G%),".",2),XQADATA=$G(XPDANM)_U_$G(XPDASEQ)_U_$G(XMZ)_U_$G(XPDFTNM)_U_"TEST",XQAROU="MAN^XPDAUTL",XQA("G.PATCHES")="",XQAMSG="Install "_$G(XPDANM)_" MM: "_$G(XMZ)_" Must be INSTALLED MANUALLY " D SETUP^XQALERT
I '($G(XPDANM))",)$G(XMZ) D G EXIT
.;SEND ALERT NOT INSTALLABLE
.S XQA("G.PATCHES")="",XQAMSG="MM: "_XMZ_" "_$P($G(^XMB(3.9,XMZ,0)),("^",1)_" NO T INSTALLABLE." D SETUP^XQALERT
I '$D(XPDANM) G EXIT
D:$G(XPDACHK) ;NOW CHECK THE SEQUENCE #
S XPDASEQ=+$P(^XMB(3.9,XPDAIXMZ,0),"#",2)
.S I=$P(XPDANM,"*",1,2),X=0,XPDACHK2=1 F S I=$O(^XPD(9.7,"B",I)) Q:I'[$P(XPDANM,"*",1,2) S J=$O(^ (I,0)) I +$P($G(^XPD(9.7,J,2)),("#",2)=(XPDASEQ-1) S XPDACHK2=0 Q
I $G(XPDACHK2) S:$O(^XPD(9.76,"E",$P(XPDANM,"*",1,2)_"**_(XPDASEQ-1),""))!(XPDASEQ=1) XPDACHK2=0
;DAF if patch DID NOT ENTER SEQ # IN FILE COMMENT CHECK 9.76
OPEN ;
D:$D(XPDANM) I POP G OPEN ;SET UP LOG FILE
.S XPDADIR=$P($G(^XPD(9.72,1,0)),U,6) S:$G(XPDADIR) XPDADIR=0 S
XPDAFN=$P(XPDANM,"*")_"_"_$P($P(XPDANM,"*",2),".")_"_"_$P($P(XPDANM,"*",2),".",2)_"_"_$P(XPDANM,"*",3)_"_.TXT",MODE=1
.D OPEN^%ZISH("AUTOPATCH",XPDADIR,XPDAFN,"A",300) I 'POP S XPDDEV=IO U XPDDEV

```

```
;CHECK TO SEE IF THIS PACKAGE IS ENABLED
S (A1,XPDAPKNM)=$$PKG^XPDUTL(XPDANM),XPDAPKNM=$0(^DIC(9.4,"C",XPDAPKNM,0)),XPDACHK3=$G(^DIC(9.4,XPDAPKNM,24)) I 'XPDACHK3 D G EXIT
.D DT^DICRW S XQAID="XPD1;"_DUZ_" ;"$P($G(%),"",2),XQADATA=$G(XPDANM)_U_$G(XPDASEQ)_U_$G(XMZ)_U_$G(XPDFTNM)_U_"TEST",XQAROU="MAN^XPDUTL",XQA("G.PATCHES")=""
.S XQAMSG="PACKAGE IEN:"_XPDAPKNM_" PKNM: "_A1_" is disabled in Package file.
INSTALL MANUALLY!!!!"
.D SETUP^XQALERT
I $G(XPDACHK2) D G REQUE ;SEND MESSAGE THAT PREVIOUS SEQUENCE CAN'T BE FOUND
.U XPDDEV W !!!,"ACTION REQUIRED: Sequence # "_XPDASEQ-1_" could not be found.",!,"An Install file entry with File Comment",!,"containing #Seq-number must exist before I can continue!!!"
.Q
I '$G(XPDACHK) D G EXIT
.S TEXT(1)="1-Message received by 'G.PATCH' that wasn't from the Patch "
.S TEXT(2)="module or wasn't installable. #"_XPDAXMZ,XMY("G.IRM")=""
.S XMSUB="NON Kids message received. No processing done.",XMTEXT="TEXT("
.D ^XMD S XMZ=XPDAXMZ
I $P($G(^XMB(3.9,XMZ,0)),U,2)'("<"National Patch Module"@DOMAIN.EXT>")&($P(^XMB(3.9,XMZ,0),U,2)'["@DOMAIN.EXT>")) D G EXIT ;DAF;ALLOWED ANY MESSAGE FROM FORUM;
AM SELECTIVELY HANDLING MESSAGES ANYWAY
.S TEXT(1)="2-Message received by 'G.PATCH' that wasn't from the Patch ",TEXT(2)="module, or wasn't installable. #"_XPDAXMZ,XMY("G.IRM")="" ,XMSUB="NON Kids message received. No processing done.",XMTEXT="TEXT(" D ^XMD S XMZ=XPDAXMZ
.Q
;
XI ;CHECK INSTALL FILE STATUS FOR PREVIOUS INSTALLATIONS OR ERRORS
S (XPDAXDA,XPDAX0)=$0(^XPD(9.7,"B",XPDANM,0)) I $G(XPDAX0) S XPDTEMP=$G(^XPD(9.7,XPDAX0,0)),XPDAX0=$P(XPDTEMP,"",9) I XPDAX0>0 D I XPDAX0'=6 G EXIT ;SEND ALERT AND EXIT
.;SEND ALERT AND EXIT
.;ALLOW INFORMATIONAL OVERRIDE
.I XPDAX0=6 S DIK="^XPD(9.7,"DA=XPDAXDA D ^DIK K DIK,DA Q
.S XQA("G.PATCHES")="" ,XQAMSG=$G(XPDANM)_ " Has a Install Status > 0!! No INSTALL PERFORMED!! " D SETUP^XQALERT
.U XPDDEV W !!!,$G(XPDANM)_ " Has a Install Status > 0!! No INSTALL PERFORMED!! "
.Q
S XPDASTOP=0,XPDAUTO=1,TYPE="N",XMP2="I" D XI^XMP2 S XMZ=XPDAXMZ W !!! ;LOAD KIDS
S XPDASTOP=0,XPDAUTO=1,TYPE="N",XMP2="I" D EN1^XPDACS S XMZ=XPDAXMZ I $G(XPDCHKSM)>0 W
!!!,"CHECKSUM FAILED! ABORTING!!",!!!! G EXIT ;VERIFY CHECKSUM
S XPDASTOP=0,XPDAUTO=1,TYPE="N",XMP2="I" D EN^XPDIB S XMZ=XPDAXMZ W !!!,"BACKING UP ROUTINES",!!!
;BACKUP
S XPDASTOP=0,XPDAUTO=1,TYPE="N",XMP2="I" D STOP^XPD AUTO2,EN^XPD I,START^XPD AUTO2
S XMZ=XPDAXMZ D ^%ZISC ;INSTALL
D CLOSE^%ZISH("AUTOPATCH")
;Now let's check and see if the INSTALL was successful
S XPDAX0=$0(^XPD(9.7,"B",XPDANM,0)) I $G(XPDAX0) S XPDTEMP=$G(^XPD(9.7,XPDAX0,0)),XPDAX0=$P(XPDTEMP,"",9) D ;SEND ALERT
.;SEND ALERT AND EXIT
.S XPDASTAT=$S(XPDAX0=3:"INSTALLED SUCCESSFULLY!!!",1:"SOMETHING WRONG! INSTALL UNSUCCESSFUL!!!!"),XQA("G.PATCHES")="" ,XQAMSG=$G(XPDANM)_ " Has an Install Status "_XPDASTAT
K:$G(XPDAX0)=3 XQAROU D SETUP^XQALERT
.Q
;Do a final check on the INSTALL file to ensure the SEQUENCE # is placed in FILE COMMENT
I $G(XPDAX0)=3 S DA=$0(^XPD(9.7,"B",XPDANM,0)) I $G(DA) K DIE S DIE="^XPD(9.7,"
I $P($G(^XPD(9.7,DA,2)),U,2)'("<"_XPDASEQ) S XPDASEQ="SEQ #"_XPDASEQ,DR="6///^
S X=XPDASEQ D ^DIE K DIE,DR,DA
G EXIT
REQUE ;REQUEUE TASKS THAT DIDN'T MEET PREREQUISITES
U XPDDEV W !!!,XPDANM," HAS BEEN REQUED FOR INSTALL!",!,"PLEASE CORRECT DEFICIENCY DESCRIBED IN THE LOG FILE FOR SUCCESS!"
D ;
.S ZTRTN="QUE^XPD AUTO",ZTSAVE("XPD*")="" ,ZTSAVE("XM*")=""
.S ZTDESC="Msg #"_XG(XMZ)_ " Sub: " $P($G(^XMB(3.9,XMZ,0)),U,2) D NOW^%DTC S X1=X,X2=1 D C^%DTC S ZTDTH=%H_" ,"$P($H,"",2) S ZTREQ="ZTDTH,ZTDESC;ZTRTN"
;SEND ALERT AND EXIT
```

```

S XQA("G.PATCHES")="",XQAMSG="Install "_$G(XPDANM)_" MM: "_$G(XMZ)_" Has been REQUEUED! TASK:
_"$G(ZTSK)_" Log file Action Required. " D SETUP^XQALERT
U XPDDEV W !!!,"TASK REQUEUED TO TASK # "_$G(ZTSK) G EXIT
Q
VER ;PACKAGE DOESN'T APPEAR TO BE INSTALLED
;SEND ALERT AND EXIT
S XQA("G.PATCHES")="",XQAMSG="Install "_$G(XPDANM)_" MM: "_$G(XMZ)_" This PACKAGE doesn't appear
to be installed. ABORTING!!!" D SETUP^XQALERT
G EXIT
EXIT ;KILL VARIABLES AND QUIT
K %H,XPDAUTO,XPDACHK,XPDACHK2,XPDACHK3,XPDACHK4,XPDDEV,XPDADIR,XPDAERNM,XPDSTPXU,XPDASTPT
K XPDASXWB,XPDASTPL,XPDASTPM,XPDINC,XPDATMP,XPDANM,XPDADT,X,X1,X2,X3,Y,XPDINC,XPDAERXM
K XPDAFN,XPD00,XPDA000,XPDAPKNM,XPDASEQ,XPDASITE,XPDASTOP,XPDATM,XPDAXMZ,XPDQX,AUTOINST,DIC
K DIR,I,J,MODE,POP,TEXT,TYPE,XMDUZ,XMP2,XMR,XMSUB,XMTEXT,XMY,XMZ,XQA,XQAMSG,ZTD
ESC,ZTDTN,ZTRTN,ZTSAVE
K %,A1,XPDCHKSM,XPDAPK,XPDAX1,XPDFTNM,XPDAIX,XPDASTAT,XPDAX0,XQADATA,XQAID,ZTSK,XMR
;K DUZ(0)
D CLOSE^%ZISH("AUTOPATCH")
Q

XPDAUTO2
XPDAUTO2 ;CAVHCS-619/TEA - AUTOPATCH CONTROL ROUTINE ;05/04/2004 09:39
;;8.0;KERNEL;**345**;Jul 10, 1995;Build 1
STOP ;STOP ALL REQUIRED PROCESSES
S:'$G(XPDSTPXU) XPDSTPXU=0 I XPDSTPXU D DISABL W !,"INHIBIT LOGONS COMMAND HAS BEEN ISSUED!" H 30
S:'$G(XPDASTPL) XPDASTPL=0 I XPDASTPL D CLEAR^HLCS2,LLP^HLCS2(1) W !,"STOP HL7 BACKGROUND
PROCESSES COMMAND HAS BEEN ISSUED!" H 30
S:'$G(XPDASTPM) XPDASTPM=0 I XPDASTPM S $P(^XMB(1,1,0),U,16)=1 S $P(^XMB(1,1,0),U,18)=1 W !,"STOP
MAILMAN COMMAND HAS BEEN ISSUED!" H 120
S:'$G(XPDASXWB) XPDASXWB=0 I XPDASXWB D STOPALL^XWBTCP W !,"STOP BROKER LISTENER COMMAND HAS BEEN
ISSUED!" H 30
S:'$G(XPDASTPT) XPDASTPT=0 I XPDASTPT H 60 D GROUP^ZTMKU("SMAN(NODE)"),GROUP^ZTMKU("SSUB(NODE)") W
!,"STOP TASKMAN COMMAND HAS BEEN ISSUED!" H 30
Q
START ;RESTART ALL REQUIRED PROCESSES
I $G(XPDSTPXU) D ENABLE W !,"ENABLE LOGONS COMMAND HAS BEEN ISSUED!"
I $G(XPDASTPT) D START^ZTMB W !,"RESTART TASKMAN COMMAND HAS BEEN ISSUED (DELAY 90)!" H 90
I $G(XPDASTPL) D CLEAR^HLCS2,STARTF^HLCS2,LLP^HLCS2(0),STRT^HLCS2 W !,"RESTART HL7 BACKGROUND JOBS
COMMAND HAS BEEN ISSUED!"
I $G(XPDASXWB) D RESTART^XWBTCP W !,"RESTART DEFINED BROKER LISTENERS COMMAND HAS BEEN ISSUED!"
I $G(XPDASTPM) D START^XMKPL S $P(^XMB(1,1,0),U,18)="" W !,"RESTART MAILMAN COMMAND HAS BEEN
ISSUED!!"
Q
ASK ;VALIDATE ROUTINE OPTION AND PROTOCOLS ARE VALID.
Q
QUES ;Ask Developer questions pertinent to AUTOPATCHING and store information in file 9.74.
;Ask if any questions are asked in POST or PRE Install that require User interaction
;and would pre-clude the KIDs patch from being automaticall installed.
;Ask Developer if Logons should be inhibited.
;Ask Developer if Taskman should be stopped.
;Ask Developer if HL7 should be stopped.
;Ask Developer if Mailman should be stopped.
;Ask Developer if Broker should be stopped.
;Ask Developer to list Options to be placed out of service.
;Ask Developer to list Protocols to be placed out of service.
;INSTALL MODIFIERS ARE STORED IN FILE 9.74 for retrieval.
S DIC="^XPD(9.74," ,DIC(0)="AEQMZL" D ^DIC S (XPDA,DA)=+Y,DIE="^XPD(9.74," ,DR="
.01;.02;9;1;2;3;4;5;11;7;8;6;10//0" D ^DIE
Q
GETANS ;CHECK FOR VAPU INSTALL MODIFIERS (9.74)
;CHECK FOR VAPU INSTALL MODIFIERS
;INSTALL MODIFIERS TAKE PRECEDENCE OVER PACKAGE MODIFIERS
;OUT OF ORDER PROTOCOLS/OPTIONS HAVE EQUAL PRESCEDENCE.
;Any OPTION/PROTOCOL defined in either file will be taken offline.
;
;GETTING INSTALL MODS ---> GO TO PACKAGE MODS IF NONE EXIST

```

```

S II=0 F S II=$0(^XMB(3.9,XMZ,2,II)) Q:+II=0 I ^XMB(3.9,XMZ,2,II,0)[ "***INSTALL NAME**" S
XPDANM=^XMB(3.9,XMZ,2,(II+1),0) K II Q
S DIC=" ^XPD(9.74," ,X=$G(XPDANM),DIC(0)="M" D ^DIC I Y>0 S XPDAIX=+Y,XPDATMP=$G
(^XPD(9.74,XPDAIX,0)) D ;
.S AUTOINST=$P(XPDATMP,U,9),XPDSTPXU=$P(XPDATMP,U,6),XPDASTPT=$P(XPDATMP,U,2
),XPDASTPL=$P(XPDATMP,U,3),XPDASTPM=$P(XPDATMP,U,4),XPDASXWB=$P(XPDATMP,U,5),XPD00=$P(XPDATMP,U
,10)
;
;NOW SEE IF THERE ARE PACKAGE MODIFIERS, IF INSTALL MODIFIERS WERE NOT FOUND
S XPDAPK=$P($G(XPDANM),"*"),XPDAPK=$0(^DIC(9.4,"C",XPDAPK,0)),XPDAX1=$0(^XPD(9.75,"B",XPDAPK,0)) I
XPDAX1,$P($G(^XPD(9.74,XPDAIX,0)),U,9)=" " S XPDATMP=$G(^XPD(9.75,XPDAX1,0)) D ;
.S XPDSTPXU=$P(XPDATMP,U,6),XPDASTPT=$P(XPDATMP,U,2),XPDASTPL=$P(XPDATMP,U,3
),XPDASTPM=$P(XPDATMP,U,4),XPDASXWB=$P(XPDATMP,U,5),XPD00=$P(XPDATMP,U,10)
;BUILD THE XPDXQ ARRAY COMPOSED OF OPTIONS AND PROTOCOLS FROM FILES 9.74 AND 9.75
;THIS ARRAY IS USED IN THE KERNEL ROUTINE XQ00
;XPDAIX=9.74'S IEN
;XPDAX1=9.75'S IEN
;
K ^TMP($J),^TMP("PRO1"),^TMP("OP1")
;OPTIONS GO FIRST
M:$G(XPDAIX) ^TMP($J)=^XPD(9.74,XPDAIX,3) M:$G(XPDAX1) ^TMP($J)=^XPD(9.75,XPDAX1,2)
M ^TMP("OP1")=^TMP($J,"B")
K ^TMP($J)
S J=" " F I=1:1 S XPDXQ(I)=$0(^TMP("OP1",J)),J=XPDXQ(I) Q:J'] " "
;
;NOW PROTOCOLS
M:$G(XPDAIX) ^TMP($J)=^XPD(9.74,XPDAIX,2) M:$G(XPDAX1) ^TMP($J)=^XPD(9.75,XPDAX1,1)
M ^TMP("PRO1")=^TMP($J,"B")
K ^TMP($J)
S J=" " F K=1:1 S XPDXQ(K+I)=$0(^TMP("PRO1",J)),J=XPDXQ(K+I) Q:J'] " "
K ^TMP($J),^TMP("PRO1"),^TMP("OP1"),I,J,K
Q
MODIFY ;PROCESS INSTALL MODIFIER MESSAGE
;XPDAXMR SHOULD LOOK LIKE "##MODIFY INSTALL##EN*7.0*80##71" WHERE 2ND ##PIECE IS INSTALL NAME AND
3RD PIECE IS SEQUENCE #.
;XPDAXMZ=MESSAGE IEN
S XPDAXMR=$G(^XMB(3.9,XPDAXMZ,0))
S XPDANM=$P($G(XPDAXMR),"##",3),XPDASEQ=$P($P($G(XPDAXMR),"##",4),U)
;SEE IF ENTRIES IN 9.4 OR 9.74 AND ADD IF NOT AND MODIFY EXISTING ENTRIES.
;IF ALREADY INSTALLED SEND NOTIFICATION
S DIC=" ^XPD(9.74," ,DIC(0)="M",X=XPDANM D ^DIC D:+Y<1 S XPDAIX=+Y
.S X=XPDANM,DIC(0)="ML" D ^DIC
.Q
;NOW GET DATA FROM MESSAGE AND STORE IN 9.74
S U=" ^"
F I=1:1 S XPDA0=$G(^XMB(3.9,XMZ,2,I,0)) Q:XPDA0=" " S:XPDA0["##ROOT##" XPDATMP=$P(XPDA0,"##",3)
D Q
.;NOW STORE ROOT IN 9.74
.S
A0=XPDATMP,A1=$P(A0,U),A2=$P(A0,U,2),A3=$P(A0,U,3),A4=$P(A0,U,4),A5=$P(A0,U,5),A6=$P(A0,U,6),A7=$P
(A0,U,7),A8=$P(A0,U,8),A9=$P(A0,U,9),A10=$P(A0,U,10),A11=$P(A0,U,11)
.S DIE=" ^XPD(9.74," ,DA=XPDAIX,DR=".02///^S X=A11;1///^S X=A2;2///^S X=A3;3///^S
X=A4;4///^S X=A5;6///^S X=A7;9///^S X=A9;10///^S X=A8;11///^S X=A10"
.D ^DIE
;NOW GET ANY OPTIONS AND PROTOCOLS
;OPTIONS ROOT ==>> ^XPD(9.74,6,3,
;PROTOCOLS ROOT==>> ^XPD(9.74,6,2,
S J=I+2,INC=1 I $G(^XMB(3.9,XPDAXMZ,2,J,0))="##" D ;
.;ENTER OPTIONS & PROTOCOLS IN 9.74
.S DA(1)=XPDAIX,DA=INC,DIE=" ^XPD(9.74," _XPDAIX_",3," ,DR="8",DR(2,9.7408)=".01///^S X=XPDA1"
.S J=J+1 F K=J:1 S XPDAX1=$G(^XMB(3.9,XPDAXMZ,2,K,0)) Q:XPDA1="##" D ;
..D ^DIE S INC=INC+1,DA=INC D:XPDA1=" " Q:XPDA1="##"
...S INC=1,DA=INC,DIE=" ^XPD(9.74," _XPDAIX_",2," ,DR="7",DR(2,9.7407)=".01///^S X=XPDA1"
;CHECK FOR PATCH ENTERED IN ERROR
I $G(A8) G ERROR
;check for FTP files

```



```

I $P(A7,".",2)="KID" G ^XPDAFTP ;MAYBE THIS SHOULD BE A GO
G EXIT
Q
DISABL ; disable logons in the volume set file...
N NVSX,NVSXN
S NVSX=0
F S NVSX=$0(^%ZIS(14.5,NVSX)) Q:'NVSX D
.S NVSXN=$P(^%ZIS(14.5,NVSX,0),"^")
.S $P(^%ZIS(14.5,NVSX,0),"^",2)="Y"
.S ^%ZIS(14.5,"LOGON",NVSXN)=1
.K NVSXN
Q
ENABLE ; (re-)enable logons in the volume set file...
S NVSX=0
F S NVSX=$0(^%ZIS(14.5,NVSX)) Q:'NVSX D
.S $P(^%ZIS(14.5,NVSX,0),U,2)="N"
.S NVSXN=$P(^%ZIS(14.5,NVSX,0),U)
.S ^%ZIS(14.5,"LOGON",NVSXN)=0
.K NVSXN
K NVSX
Q
EXIT ;KILL VARIABLES AND QUIT
K %H,XPDAUTO,XPDACHK,XPDACHK2,XPDACHK3,XPDACHK4,XPDDEV,XPDA DIR,XPDAERNM,XPDSTPXU,XPDASTPT
K XPDASXWB,XPDASTPL,XPDASTPM,XPDINC,XPDATEMP,XP DANM,XP DADT,X,X1,X2,X3,Y,XP DINC,XPDAERXM
K XPDAFN,XPDOO,XPDAOOO,XPDA PKNM,XPDASEQ,XP DASITE,XP DASTOP,XP DATM,XP DAXMZ,XPDXQ,AUTOINST,DIC
K DIR,I,J,MODE,POP,TEXT,TYPE,XMDUZ,XMP2,XMSUB,XMTEXT,XMY,XMZ,XQA,XQAMSG,ZTDESC,ZTDTH,ZTRTN,ZTSAVE
K A0,A1,A2,A3,A4,A5,A6,A7,A8,A9,A10,A11,XPDA0,XPDA1,XPDA A,XPDA PK,XPDA X1,XPDAIX,
XPDA XMR,DA,DIE,DR,INC
D CLOSE^%ZISH("AUTOPATCH")
Q

```

ERROR ;UPDATE INSTALL FILE WITH ERROR ENTRY

```
;
S DUZ(0)="@"
K DIC S X=XPANM,XPDASEQ="SEQUENCE #" _XPDASEQ,DIC="^XPD(9.7," ,DIC(0)="L",DIC("DR")=".01///^S
X=XPANM;6///^S X=XPDASEQ;.02///5" D ^DIC K DIC,Y
S XQA("G.PATCHES")="",XQAMSG="Install "_$G(XPDANM)" "_XPDASEQ_" Has been FLAGGED as ENTERED IN
ERROR. WILL NOT INSTALL!!!" D SETUP^XQALERT
G EXIT
```

XPDAUTOM

XPDAUTOM ;CAVHCS-619/TEA - AUTOPATCH CONTROL ROUTINE ;05/04/2004 09:39

;8.0;KERNEL;**345**;Jul 10, 1995;Build 1

SETUP ;GET NECESSARY INFORMATION FROM FILE

D QUES

```
S XPDATMP=$G(^XPD(9.74,XPDA,0)),(XMZ,XPDAXMZ)=$P(XPDATMP,U,12),XPANM=$P($G(
^XPD(9.74,XPDA,0)),U),XPDASEQ=$P($G(^XPD(9.74,XPDA,0)),U,11) D HOME^%ZIS S XPD
DEV=IO I $P($G(^XPD(9.74,XPDA,0)),U,12)
```

E S XPDAFTNM=\$P(\$G(^XPD(9.74,XPDA,0)),U,7)

D GETANS

S XPDAUTO=0

XI ;CHECK INSTALL FILE STATUS FOR PREVIOUS INSTALLATIONS OR ERRORS

```
S XPDAX0=$O(^XPD(9.7,"B",XPANM,0)) I $G(XPDAX0) S XPDATMP=$G(^XPD(9.7,XPDA,0
,0)),XPDAX0=$P(XPDATMP,"",9) I XPDAX0>0 D G:$G(Y)=0!$D(DUOUT)!$D(DTOUT) EXIT
```

;SEND ALERT AND EXIT

.;SEND NOTICE TO SCREEN

.W !!!,"THIS INSTALL HAS A STATUS OF "_XPDAX0_" "_\$P(\$P(^DD(9.7,.02,0),XPDA,2),":",2),";")

.S DIR("A")="Do you want to continue? (Y/N) ",DIR(0)="Y",DIR("B")="N" D ^DIR Q:

Y=0!\$D(DUOUT)!\$D(DTOUT)

I \$G(XMZ) S XMR=\$G(^XMB(3.9,XMZ,0))

```
I $G(XPDAFTNM) [" .KID" S XPDAUTO=1 I $G(XMZ) D CHECK^XMP2A G:+Y<1 NOIN D:$G(XMZ
) XI^XMP2 S:$G(XPDAXMZ) XMZ=XPDAXMZ S XPDAUTO=0 W !!! ;LOAD KIDS
```

```
I $G(XPDAFTNM) [" .KID" S XPDAUTO=1 D EN1^XPDIL S XPDAUTO=0 K XPDAFTNM W !!!,"NOW LET'S LOOK AT
THOSE KIDS CHECKSUMS"
```

```
D EN1^XPDDCS G:$G(Y)=0!$D(DUOUT)!$D(DTOUT) EXIT S:$G(XPDAXMZ) XMZ=XPDAXMZ I $G(
XPDCHKSM)>0 W !!!,"CHECKSUM FAILED! ABORTING!!",!!!! G EXIT ;VERIFY CHECKSUM
```

```
W !!! S DIR("A")="WERE CHECKSUMS OK? ",DIR(0)="Y",DIR("B")="Y" D ^DIR G:$G(Y)=0
!$D(DUOUT)!$D(DTOUT) EXIT
```

W !!!,"Okay then let's back up the ROUTINES!! "

D EN^XPDIB S:\$G(XPDAXMZ) XMZ=XPDAXMZ W !!!,"BACKING UP ROUTINES",!!! ;BACKUP

W !!!,"OKAY!! It's time to INSTALL!! This is your last chance to ABORT!!!"

```
S DIR("A")="Is everything OK and shall I continue? ",DIR(0)="Y",DIR("B")="Y" D ^DIR
G:Y=0!$D(DUOUT)!$D(DTOUT) EXIT
```

```
S XPDASTOP=0,TYPE="N",XMP2="I" D STOP^XPDAUTO2 S XPDAUTO=0 D EN^XPDI D START^XPDAUTO2 S
XMZ=XPDAXMZ D ^%ZISC ;INSTALL
```

;Now let's check and see if the INSTALL was successful

```

S XPDAX0=$0(^XPD(9.7,"B",XPDANM,0)) I $G(XPDAX0) S XPDTEMP=$G(^XPD(9.7,XPDAX0
,0)),XPDAX0=$P(XPDTEMP,"",9) D ;SEND ALERT AND EXIT
.;SEND ALERT AND EXIT
.I XPDAX0=3,$D(XQAID) S XQAID=XPDAXQID,XQAKILL=0 D DELETE^XQALERT
.S XPDASTAT=$S(XPDAX0=3:"INSTALLED SUCCESSFULLY!!!",1:"SOMETHING WRONG! INSTALL
UNSUCCESSFUL!!!!"),XQA("G.PATCHES")="",XQAMSG=$G(XPDANM)_" Has an Install Status "_XPDASTAT
K:$G(XPDAX0)=3 XQAROU D SETUP^XQALERT
.Q
;Do a final check on the INSTALL file to ensure the SEQUENCE # is placed in FILE COMMENT
I $G(XPDAX0)=3 S DA=$0(^XPD(9.7,"B",XPDANM,0)) I $G(DA) K DIE S DIE="^XPD(9.7,"
I $P($G(^XPD(9.7,DA,2)),")'["#"_XPDASEQ) S XPDASEQ="SEQUENCE #"_XPDASEQ,DR="6///^S
X=XPDASEQ",DIE("NO^")="" D ^DIE K DIE
G:$G(XPDACHK) EXIT
Q
EXIT ;KILL VARIABLES AND QUIT
K %H,XPDAUTO,XPDACHK,XPDACHK2,XPDACHK3,XPDACHK4,XPDDEV,XPDADIR,XPDAERNM,XPDSTPXU,XPDASTPT
K XPDASXWB,XPDASTPL,XPDASTPM,XPDINC,XPDTEMP,XPDANM,XPDADT,X,X1,X2,X3,Y,XPDINC,XPDAERXM,XMR
K XPDAPN,XPD00,XPDA000,XPDAPKNM,XPDASEQ,XPDASITE,XPDASTOP,XPDATM,XPDAXMZ,XPDXQ,AUTOINST,DIC
K DIR,I,J,MODE,POP,TEXT,TYPE,XMDUZ,XMP2,XMSUB,XMTEXT,XMY,XMZ,XQA,XQAMSG,ZTDESC,ZTDTH,ZTRTN,ZTSAVE
K DR,XPDAA,XPDAPK,XPDAX1,XPDAIX,XPDAX0,XPDCHKSM,XPDASTPW,XPDASTAT,XPDAXQID,DA,D
TOUT,DUOUT,XQAID,XQAKILL
Q
QUES ;Ask Developer questions pertinent to AUTOPATCHING and store information in file 9.74.
;Ask if any questions are asked in POST or PRE Install that require User interaction
;and would pre-clude the KIDs patch from being automaticall installed.
;Ask Developer if Logons should be inhibited.
;Ask Developer if Taskman should be stopped.
;Ask Developer if HL7 should be stopped.
;Ask Developer if Mailman should be stopped.
;Ask Developer if Broker should be stopped.
;Ask Developer to list Options to be placed out of service.
;Ask Developer to list Protocols to be placed out of service.
;INSTALL MODIFIERS ARE STORED IN FILE 9.74 for retrieval.
S DIC="^XPD(9.74,"DIC(0)="AEQMZL" D ^DIC S (XPDAAX,DA)=+Y,DIE="^XPD(9.74,"DR="
.01;.02;9;1;2;3;4;5;11;7;8;6;10//0" D ^DIE
.Q
GETANS ;CHECK FOR VAPU INSTALL MODIFIERS (9.74)
;CHECK FOR VAPU INSTALL MODIFIERS
;INSTALL MODIFIERS TAKE PRECEDENCE OVER PACKAGE MODIFIERS
;OUT OF ORDER PROTOCOLS/OPTIONS HAVE EQUAL PRECEDENCE.
;Any OPTION/PROTOCOL defined in either file will be taken offline.
;
;GETTING INSTALL MODS ---> GO TO PACKAGE MODS IF NONE EXIST
S DIC="^XPD(9.74,"X=$G(XPDANM),DIC(0)="M" D ^DIC I Y>0 S XPDAIX=+Y,XPDTEMP=$G(^XPD(9.74,+Y,0)) D
;
.S AUTOINST=$P(XPDTEMP,U,9),XPDSTPXU=$P(XPDTEMP,U,6),XPDASTPT=$P(XPDTEMP,U,2
),XPDASTPL=$P(XPDTEMP,U,3),XPDASTPM=$P(XPDTEMP,U,4),XPDASXWB=$P(XPDTEMP,U,5),XPD00=$P(XPDTEMP,U
,10)
;
;NOW SEE IF THERE ARE PACKAGE MODIFIERS, IF INSTALL MODIFIERS WERE NOT FOUND
S XPDAPK=$P($G(XPDANM),"*"),XPDAPK=$0(^DIC(9.4,"C",XPDAPK,0)),XPDAX1=$0(^XPD(9.
75,"B",XPDAPK,0)) I XPDAX1,+Y<1 S XPDTEMP=$G(^XPD(9.75,XPDAX1,0)) D ;
.S XPDSTPXU=$P(XPDTEMP,U,6),XPDASTPT=$P(XPDTEMP,U,2),XPDASTPL=$P(XPDTEMP,U,3
),XPDASTPM=$P(XPDTEMP,U,4),XPDASXWB=$P(XPDTEMP,U,5),XPD00=$P(XPDTEMP,U,10)
;BUILD THE XPDXQ ARRAY COMPOSED OF OPTIONS AND PROTOCOLS FROM FILES 9.74 AND 9.75
;THIS ARRAY IS USED IN THE KERNEL ROUTINE XQOO
;XPDAIX=9.74'S IEN
;XPDAX1=9.75'S IEN
;
K ^TMP($J),^TMP("PRO1"),^TMP("OP1")
;OPTIONS GO FIRST
M:$G(XPDAIX) ^TMP($J)=^XPD(9.74,XPDAIX,3) M:$G(XPDAX1) ^TMP($J)=^XPD(9.75,XPDAX1,2)
M ^TMP("OP1")=^TMP($J,"B")
K ^TMP($J)
S J="" F I=1:1 S XPDXQ(I)=$0(^TMP("OP1",J)),J=XPDXQ(I) Q:J']""
;

```

```

;NOW PROTOCOLS
M:$G(XPDAIX) ^TMP($J)=^XPD(9.74,XPDAIX,2) M:$G(XPDAX1) ^TMP($J)=^XPD(9.75,XPDAIX,1)
M ^TMP("PRO1")=^TMP($J,"B")
K ^TMP($J)
S J=" F K=1:1 S XPDQ(K+I)=$0(^TMP("PRO1",J)),J=XPDQ(K+I) Q:J']"
K ^TMP($J),^TMP("PRO1"),^TMP("OP1"),I,J,K
Q
STOP ;STOP ALL REQUIRED PROCESSES
S:$G(XPDSTPXU) XPDSTPXU=0 I XPDSTPXU D DISABL W !,"INHIBIT LOGONS COMMAND HAS BEEN ISSUED!" H 30
S:$G(XPDASTPL) XPDASTPL=0 I XPDASTPL D CLEAR^HLCS2,LLP^HLCS2(1) W !,"STOP HL7
BACKGROUND PROCESSES COMMAND HAS BEEN ISSUED!" H 30
S:$G(XPDASTPM) XPDASTPM=0 I XPDASTPM S $P(^XMB(1,1,0),U,16)=1 S $P(^XMB(1,1,0),U,18)=1 W !,"STOP
MAILMAN COMMAND HAS BEEN ISSUED!" H 30
S:$G(XPDASXWB) XPDASXWB=0 I XPDASXWB D STOPALL^XWBTCP W !,"STOP BROKER LISTENER COMMAND HAS BEEN
ISSUED!" H 30
S:$G(XPDASTPT) XPDASTPT=0 I XPDASTPT H 60 D GROUP^ZTMKU("SMAN(NODE)"),GROUP^ZTMKU("SSUB(NODE)") W
!,"STOP TASKMAN COMMAND HAS BEEN ISSUED!" H 30
Q
START ;RESTART ALL REQUIRED PROCESSES
I $G(XPDSTPXU) D ENABLE W !,"ENABLE LOGONS COMMAND HAS BEEN ISSUED!"
I $G(XPDASTPT) D START^ZTMB H 90 W !,"RESTART TASKMAN COMMAND HAS BEEN ISSUED (DELAY 90)!"
I $G(XPDASTPL) D CLEAR^HLCS2,STARTF^HLCS2,LLP^HLCS2(0),STRT^HLCS2 W !,"RESTART HL7 BACKGROUND JOBS
COMMAND HAS BEEN ISSUED!"
I $G(XPDASXWB) D RESTART^XWBTCP W !,"RESTART DEFINED BROKER LISTENERS COMMAND HAS BEEN ISSUED!"
I $G(XPDASTPM) D START^XMKPL S $P(^XMB(1,1,0),U,18)=" W !,"RESTART MAILMAN COMMAND HAS BEEN
ISSUED!!"
Q
DISABL ; disable logons in the volume set file...
N NVSX,NVSXN
S NVSX=0
F S NVSX=$0(^%ZIS(14.5,NVSX)) Q:'NVSX D
.S NVSXN=$P(^%ZIS(14.5,NVSX,0),"^")
.S $P(^%ZIS(14.5,NVSX,0),"^",2)="Y"
.S ^%ZIS(14.5,"LOGON",NVSXN)=1
.K NVSXN
Q
ENABLE ; (re-)enable logons in the volume set file...
S NVSX=0
F S NVSX=$0(^%ZIS(14.5,NVSX)) Q:'NVSX D
.S $P(^%ZIS(14.5,NVSX,0),U,2)="N"
.S NVSXN=$P(^%ZIS(14.5,NVSX,0),U)
.S ^%ZIS(14.5,"LOGON",NVSXN)=0
.K NVSXN
K NVSX
Q
NOIN ;PATCH ISN'T INSTALLABLE
W !!!!!,"THIS ISN'T AN INSTALLABLE MESSAGE AND I DON'T KNOW A KIDS FILENAME.",!,
"I CAN'T CONTINUE. YOU'RE ON YOUR OWN.",!,"ARE YOU SURE THIS ISN'T AN INFORMATIONAL PATCH????"
G EXIT

XPDDCS
XPDDCS ;SFISC/RSD - Display Checksum for a package ;05/05/2008
;8.0;KERNEL;**2,44,108,202,393,511**;Jul 10, 1995;Build 1
;Per VHA Directive 2004-038, this routine should not be modified.
Q
EN1 ;Verify checksums in Transport Global
N D0,DIC,X,XPDI,XPDS,XPST,XPDT,Y,Z
;S DIC="^XPD(9.7," ,DIC(0)="AEQMZ",DIC("S")="I $D(^XTMP("XPDI",Y))"
;D ^DIC Q:Y<0
S XPDS="I $D(^XTMP("XPDI",Y))"
S XPST=$$LOOK^XPDI1(XPDS) Q:XPST>0
S XPDSHW=$$ASK Q:$D(DIRUT)
S XPD("XPDT")="",XPD("XPST")="",XPD("XPDSHW")="",X="XUTMDEVQ"
;during Virgin install, XUTMDEVQ might not exists
X ^%ZOSF("TEST") E D Q
.S IOSL=99999,IOM=80,IOF="#",IOST="",Y=0 D LST1(9.7)

```

```

S Y="LST1^XPDDCS(9.7)",Z="Checksum Print"
;p345-rename AND* to XPD* - Patch was Cancelled keep code for future.
I '$G(XPDAUTO) D EN^XUTMDEVQ(Y,Z,.XPD)
I '$G(XPDAUTO) S IO=XPDDEV U XPDDEV D LST1^XPDDCS(9.7)
Q
;
ASK() ;Ask if want each routine listed
N DIR
I '$D(XPDAUTO) Q 1
S DIR(0)="YAO",DIR("A")="Want each Routine Listed with Checksums: ",DIR("A",1)="",DIR("B")="Yes"
D ^DIR
Q Y
;
EN2 ;print from build (system)
N D0,DIC,XPD,XPD,XPDT,XPDST,Y,Z
;S DIC="^XPD(9.6,",DIC(0)="AEQMZ"
;D ^DIC Q:Y<0
S XPDST=$$LOOK^XPDB1() Q:XPDST'>0
S XPDST=$$ASK Q:$D(DIRUT)
S XPD("XPDT(")="",XPD("XPDSHW")="",Y="LST1^XPDDCS(9.6)",Z="Checksum Print"
;p345-rename AND* to XPD* - Patch was Cancelled keep code for future.
I '$G(XPDAUTO) D EN^XUTMDEVQ(Y,Z,.XPD)
I '$G(XPDAUTO) S:'$D(XPDDEV) XPDDEV=0 U XPDDEV D LST1^XPDDCS(9.6)
Q
;
LST1(FILE) ;Print group
N XPDI S XPDI=0
F S XPDI=$0(XPDT(XPDI)) Q:XPDI'>0 S D0+=XPDT(XPDI) D PNT(FILE)
Q
;
PNT(XPDFIL) ;print
N XPD0,XPDC,XPDDT,XPDE,XPDI,XPDJ,XPDJG,XPDQ,XPDUL,X
Q:'$D(^XPD(XPDFIL,D0,0)) S XPD0=^(0),XPDJG=1,$P(XPDUL,"-",IOM)="",XPDDT=$$HTE^XLFD($H,"1PM")
W:$E(IOST,1,2)="C-" @IOF D HDR
W !
S XPDI="", (XPDQ,XPDE)=0
;XPDFIL=9.7 use transport global exists
I XPDFIL=9.7 D
.I '$D(^XTMP("XPDI",D0)) W !," ** Transport Global doesn't exist **" S XPDQ=1 Q
.;check for missing nodes in transport global
.I '$D(^XTMP("XPDI",D0,"BLD"))="" W !," **Transport Global corrupted, please reload **" S XPDQ=1
Q
.F XPDC=0:1 S XPDI=$0(^XTMP("XPDI",D0,"RTN",XPDI)) Q:XPDI="" S XPDJ=$G(^XPDI)) D Q:XPDQ
..I XPDJ="" W !," **Transport Global corrupted, please reload **" S XPDQ=1 Q
.;if deleting at site, there is no checksum
..I +XPDJ=1 S XPDC=XPDC-1 Q
..D SUM(XPDI,$NA(^XTMP("XPDI",D0,"RTN",XPDI)),$P(XPDJ,U,3),$P(XPDJ,U,4))
..S XPDQ=$$CHK(4)
;check build file
E D
.F XPDC=0:1 S XPDI=$0(^XPD(9.6,D0,"KRN",9.8,"NM","B",XPDI)) Q:XPDI="" S XPDJ=$0(^XPDI,0)) D
Q:XPDQ
..Q:'$D(^XPD(9.6,D0,"KRN",9.8,"NM",+XPDJ,0)) S XPDJ=$P(^XPDI,U,4)
.;quit if no checksum, routine wasn't loaded
..I XPDJ="" S XPDC=XPDC-1 Q
..N DIF,XCNP,%N
..S X=XPDI,DIF="^TMP($J,"RTN",XPDI,"",XCNP=0
..X ^%ZOSF("TEST") E W !,XPDI,?10,"Doesn't Exist" Q
..X ^%ZOSF("LOAD")
..D SUM(XPDI,$NA(^TMP($J,"RTN",XPDI)),XPDJ,"")
..S XPDQ=$$CHK(4)
Q:XPDQ
W !!?3,XPDC," Routine" $$XPDC>1:"s",1:"")_ " checked, ",XPDE," failed.",!
;p345-rename AND* to XPD*-Patch was Cancelled keep code for future.
I '$G(XPDAUTO) S XPDCHKSM=XPDE
Q

```

```

;
;XPDR=routine name, Z=global root, XPD=check sum, XPDBS=before Checksum from FORUM
SUM(XPDR,Z,XPD,XPDBS) ;check checksum
N Y
;See if we have a before checksum and compare.
I $L(XPDBS) D BEFORE(XPDR,XPDBS)
;first char. is the sum tag used in XPDRSUM
I XPD'?1U1.N W !,XPDR,?10,"ERROR in Checksum" S XPDE=XPDE+1 Q
S @("Y=$$SUM" _$E(XPD)_ ^XPDRSUM(Z)),XPD=$E(XPD,2,255)
I Y=XPD,XPDSHW W !,XPDR,?10,"Calculated "_$J(XPD,10)
I Y'=XPD W !,XPDR,?10,"Calculated "_$C(7)_$J(Y,10)_, expected value "_XPD S XPDE=XPDE+1
Q
;
BEFORE(RN,SUM) ;Check a before Checksum
N DIF,XCNP,%N,X
I SUM'?1U1.N Q
K ^TMP($J,"XPDDCS",RN) ;patch 511
S X=RN,DIF=^TMP($J,"XPDDCS",RN),XCNP=0
X ^%ZOSF("TEST") E W !,RN,?10,"Not on current system." Q
X ^%ZOSF("LOAD")
S DIF=$NA(^TMP($J,"XPDDCS",RN))
S @("Y=$$SUM" _$E(SUM)_ ^XPDRSUM(DIF)),SUM=$E(SUM,2,255)
I Y'=SUM W !,RN,?10,"Before Checksum Calculated "_Y_ expected value "_SUM
Q
;
CHK(Y) ;Y=excess lines, return 1 to exit
Q:$Y<(IOSL-Y) 0
I $E(IOST,1,2)="C-" D Q:'Y 1
.N DIR,I,J,K,X
.S DIR(0)="E" D ^DIR
S XPDPG=XPDPG+1
W @IOF D HDR
Q 0
;
HDR W !,"PACKAGE: ",$P(XPD0,U)," ",XPDDT,?70,"PAGE ",XPDPG,! ,XPDUL,!
Q

XPDI
XPDI ;SFISC/RSD - Install Process ;05/04/2004 09:41
; ;8.0;KERNEL;**10,21,39,41,44,58,68,108,145,184,229,345**;Jul 10, 1995;Build 1
EN ;install
N DIR,DIRUT,POP,XPD,XPDA,XPDD,XPDIJ,XPDDIQ,XPDIT,XPDIABT,XPDNM,XPDNOQUE,XPDPKG,
XPDREQAB,XPDST,XPDSET,XPDSET1,XPDT,XPDQUIT,XPDQUES,Y,ZTSK,%
S %="I '$P(^ (0),U,9),$D(^XPD(9.7,"ASP",Y,1,Y)),$D(^XTMP("XPDI",Y))",XPDST=$$LOOK^XPDI1(%)
Q:'XPDST!$D(XPDQUIT)
S XPDIT=0,(XPDSET,XPDSET1)=$P(^XPD(9.7,XPDST,0),U) K ^TMP($J)
;Check each part of XPDT array
F S XPDIT=$O(XPDT(XPDIT)) Q:XPDIT D Q:'$D(XPDT)!$D(XPDQUIT)
.S XPDA=XPDT(XPDIT),XPDNM=$P(XPDT(XPDIT),U,2),XPDPKG=+$P($G(^XPD(9.7,+XPDT(XPD
IT),0),U,2),%=$P(^ (0),U,5)
.W !,"Checking Install for Package ",XPDNM
.;check that Install file was created correctly
.I '$D(^XPD(9.7,XPDA,"INI"))!$D(^("INIT")) W !,"**INSTALL FILE IS CORRUPTED**",!,*7 S XPDQUIT=1 Q
.;run enviroment check routine
.;XPDREQAB req. build missing, =2 global killed
.I $$ENV^XPDI1(1) S:$G(XPDREQAB)=2 XPDQUIT=1 Q
.;save variables that are setup in environ. chck. routine
.I $D(XPDNOQUE)!$D(XPDDIQ) D
..S:$D(XPDNOQUE) ^XTMP("XPDI",XPDA,"ENVVAR","XPDNOQUE")=XPDNOQUE
..I $D(XPDDIQ) M ^XTMP("XPDI",XPDA,"ENVVAR","XPDDIQ")=XPDDIQ
.D QUES^XPDI1(XPDA) Q:'$D(XPDT(+XPDIT))!$D(XPDQUIT)
.;XPDIJ=XPDA if XPDIJ routine is part of Build
.S:$D(^XTMP("XPDI",XPDA,"RTN","XPDIJ")) XPDIJ=XPDA
.D XQSET^XPDI1(XPDA)
;NONE = no Build to install
G NONE:'$O(XPDT(" "))!$D(XPDQUIT)!($G(XPDREQAB))

```

```

;check that we have all Builds to install
S XPD=XPDEST,XPDNM=XPDEST,Y=0
F S Y=$0(^XPD(9.7,"ASP",XPD,Y)) Q:'Y S %+= $0(^ (Y,0)) I '$D(XPD("DA",%)) G NONE
W !
;See if a Master Build
S %=$0(^XTMP("XPI",XPD,"BLD",0)),%=$P(^ (%,0),U,3) S:%=1 XPD("MASTER")=XPD
;Inhibit Logon Question
D DIR^XPDQ("XPI") I $D(DIRUT) D ABRTALL(2) Q
;disable options question
D DIR^XPDQ("XPZ") I $D(DIRUT) D ABRTALL(2) Q
;XPDEST=set name,(also build name), of options that will be disabled
;XPDEST1=setname or null if they don't want to disable
D I XPDEST1="" D ABRTALL(2) Q
.;if they say no, set XPDEST1=""
.S:'$G(XPDQ("XPZ1")) XPDEST1="",Y=0
.S ^XTMP("XQ00",XPDEST,0)=XPDEST_ " is being installed by KIDS^"_DT_U_DUZ
.I XPDEST1]"" D Q:XPDEST1=""!(XPDEST1="")
.;merge the options/protocols that were put in ^TMP($J,"XQ00",build name)
..M ^XTMP("XQ00",XPDEST)=^TMP($J,"XQ00",XPDEST)
..D INIT^XQ00(.XPDEST1) Q:"^[XPDEST1
..N DIR S DIR(0)="N^0:60:0",DIR("B")=0
..S DIR("A")="Delay Install (Minutes)",DIR("?")="Enter the number of minutes to delay the
installing of Routines after the Disable of Options"
.;p345-rename AND* to XPD*
..W ! D:'$G(XPD AUTO) ^DIR S:$G(XPD AUTO) Y=0 I $D(DIRUT) S XPDEST1=""
.;Y is set in the call to DIR in previous .DO
.;save setname into first Build and the Delay in minutes, Y
.K XPD S XPD(9.7,XPDEST_,"",7)=(XPDEST1]"" )_XPDEST,XPD(9.7,XPDEST_,"",8)=Y
.D FILE^DIE("", "XPD")
;check if they want to update other CPUs
I $G(XPDQ("XPZ2")) D I $D(DIRUT) D ABRTALL(2) Q
.N DA,DIE,DIR,DR,I,XPD,X,Y,Z
.;if they haven't already added Volume Sets, populate the multiple
.I '$0(^XPD(9.7,XPD,"VOL",0)) D I $D(XPD) D UPDATE^DIE("", "XPD")
..X ^ZOSF("UCI") S Y=$P(Y,"",2),(I,Z)=0
..F S I=$0(^ZIS(14.5,I)) Q:'I S X=$G(^ (I,0)) S:$P(X,U)]""&$P(X,U,11)&($P(X,U
)'=Y) Z=Z+1,XPD(9.703,"+_Z_","_XPD_",",.01)=$P(X,U)
.W !,"I will Update the following VOLUME SETS:",!
.S I=0 F S I=$0(^XPD(9.7,XPD,"VOL",I)) Q:'I W ?3,$P(^ (I,0),U),!
.W ! S DIR(0)="Y",DIR("A")="Want to edit this list",DIR("B")="NO"
.D ^DIR Q:$D(DIRUT) D:Y
..S DA=XPD,DIE=""^XPD(9.7,"",DR=30,DR(2,9.703)=".01"
..D ^DIE
.I '$0(^XPD(9.7,XPD,"VOL",0)) W !,"No VOLUME SETS selected!!" Q
.Q:$TM^%ZTLOAD ;quit if Taskman is running
.W !,"TASKMAN is not running. If you install now, you must run the routine XPD
CPU",!,"in the production UCI for each of the VOLUME SETS you have listed once"
.W !,"the installation starts!!",!,"If you Queue the install, the VOLUME SETS will be updated
automatically.",*7,*7,!
DEV S POP=0 S:'$D(^DD(3.5,0)) POP=1
;check if home device is defined
I 'POP S IOP="",%ZIS=0 D ^%ZIS
;Kernel Virgin Install
I POP S XPD=XPDEST D:$G(XPDIJ) XPDIJ^XPDI1 G EN^XPDIJ
;set XPD=starting Build, ask for device for messages
;XPDNOQUE is defined means don't let them queue output
W !,"Enter the Device you want to print the Install messages."
W:'$D(XPDNOQUE) !,"You can queue the install by enter a 'Q' at the device prompt."
W !,"Enter a '^' to abort the install.",!
S XPD=XPDEST,%ZIS=$P("Q",U,'$D(XPDNOQUE))
;p345-rename AND* to XPD*
I '$G(XPD AUTO) D ^%ZIS G:POP ASKABRT
I $G(XPD AUTO) U XPDDEV
;reset expiration date to T+7 on transport global

```

```

S XPDD=$$FMADD^XLFD(T,7),^XTMP("XPD",0)=XPDD_U_DT
I $D(IO("Q")) D G ASKABRT:$D(ZTSK)[0 D XPDIJ^XPD1:$G(XPDIJ),QUIT^XPD1(XPDST)
Q
. N DIR,NOW S NOW=$$HTFM^XLFD($$HADD^XLFD($H,,2)) ;Must be in future
. S DIR(0)="DA^"_NOW_"^"_XPDD_"^AEFRSX"
. S DIR("A")="Request Start Time: "
. S DIR("B")=$$FMTE^XLFD(NOW)
. S DIR("?",1)="Enter a Date including Time"
. S DIR("?",2)="The time must be in the future and not to exceed 7 days in the future."
. S DIR(" ?")="Current date/time: "_DIR("B")
. D ^DIR
.Q:$D(DIRUT)
.S ZTDTH=Y,ZTRTN="EN^XPDIJ",ZTDESC=XPDNM_" KIDS install",ZTSAVE("XPDA")=""
.D ^%ZTLOAD,HOME^%ZIS K IO("Q")
.Q:$D(ZTSK)[0
.W !,"Install Queued!",!!
.;save task into first Build
.K XPD S XPD(9.7,XPDST_"",",5)=ZTSK,XPDIT=0
.F S XPDIT=$O(XPD(XPDIT)) Q:'XPDIT S XPD(9.7,+XPD(XPDIT)_"",",.02)=1 D FILE^DIE("", "XPD") K XPD
;run install
;p345-rename AND* to XPD*
I '$G(XPD AUTO) U IO D XPDIJ^XPD1:$G(XPDIJ),QUIT^XPD1(XPDST) G EN^XPDIJ
I '$G(XPD AUTO) U XPDDEV D XPDIJ^XPD1:$G(XPDIJ),QUIT^XPD1(XPDST) G EN^XPDIJ
Q
;
;XPDA=ien to del, XPK=1 kill global, XPDALL=1 deleting all
;XPDST=starting package.
ABORT(XPDA,XPK,XPDALL) ;abort install of Build XPDA
N %,DA,DIK,XPDJ,XPDNM,Y
Q:'$D(^XPD(9.7,XPDA,0)) S XPDNM=$P(^ (0),U)
D BMES^XPDUTL(XPDNM_" Build will not be installed"_$S(XPK=1:" Transport Global
deleted!",1:"")),MES^XPDUTL(" " _$HTE^XLFD($H))
S DIK="^XPD(9.7,"XPDJ=XPD("NM",XPDNM),DA=XPDA
;kill XPD array, but don't kill global if XPK=2
K XPD("NM",XPDNM),XPD("DA",XPDA),XPD(XPDJ),XPD("GP") Q:XPK=2
K ^XTMP("XPD",XPDA)
;if we are not deleting all packages and we are deleting the starting package
;set the next package to the starting package. It must always be 1.
I '$G(XPDALL),XPDA=XPDST S Y=$O(XPD(0)) D:Y
.;unlock starting install
.L ^XPD(9.7,XPDST)
.S XPDST=+XPD(Y),XPD(1)=XPD(Y),XPD("DA",XPDST)=1,XPD("NM",$P(XPD(Y),U,2))=1,XPDIT=0
.K XPD(Y) N XPD
.S %="XPD(9.7,"""_XPDST_"",""),%@(3)=XPDST,%@(4)=1
.;loop thru the rest of the packages and reset the starting package field
.F S Y=$O(XPD(Y)) Q:'Y D
..S XPD(9.7,+XPD(Y)_"",",3)=XPDST
.D FILE^DIE("", "XPD")
D ^DIK
Q
ASKABRT ;ask if want to unload distribution
N DIR,DIRUT,X,Y
S XPDQUIT=1,DIR(0)="Y",DIR("A")="Install ABORTED, Want to remove the Transport
globals",DIR("B")="YES"
W ! D ^DIR I Y D ABRTALL(1) Q
L ^XPD(9.7,XPDST)
Q
ABRTALL(XPK) ;abort all Builds
N XPDA
S XPDIT=0
F S XPDIT=$O(XPD(XPDIT)) Q:'XPDIT S XPDA=+XPD(XPDIT) D ABORT(XPDA,XPK,1)
;unlock starting install
L ^XPD(9.7,XPDST)
Q
NONE W !,"**NOTHING INSTALLED**",!
Q

```


XPDI1

```
XPDI1 ;SFISC/RSD - Cont of Install Process ;05/04/2004 09:42
; ;8.0;KERNEL;**58,61,95,108,229,275,345**;Jul 10, 1995;Build 1
;lookup into file 9.7, XPDS=DIC("S") for lookup
;return 0-fail or ien, XPDT=array of linked builds
LOOK(XPDS,XPDL) ;lookup Install
N DIC,Y,XPDL,XPDI,%
;p345-rename AND* to XPD*
I '$G(XPD AUTO) S DIC(0)="QEAMZ",DIC="^XPD(9.7,"
I '$G(XPD AUTO) S X=XP DANM,DIC(0)="QEMZ",DIC="^XPD(9.7," S:$L($G(XPDS)) DIC("S")=XPDS
D ^DIC Q:Y<0 0
I '$G(XPDL) L +^XPD(9.7,+Y,0):0 E W !,"Being accessed by another user" Q 0
S XPD+=Y,XPDI=0
W !!,"This Distribution was loaded on ",,$FMTE^XLFD($P(Y(0),U,3))," with header of
",!?3,$G(^XPD(9.7,XPDL,2)),!?3,"It consisted of the following Install(s):",!
;build XPDT array
I '$D(^XPD(9.7,"ASP",XPDL)) D XPDT(1,XPDL) Q XPD
F S XPDI=$O(^XPD(9.7,"ASP",XPDL,XPDI)) Q:'XPDI S Y=+$O(^XPDI,0)) D XPDT(XPDI,Y)
I '$O(XPDT(0)) S XPDIQUIT=1 D QUIT(XPD)
Q XPD
;
QUIT(Y) ;unlock ien Y
L -^XPD(9.7,+Y) Q
;
XPDT(P1,P2) ;Build XPDT array
N % S %=$P($G(^XPD(9.7,P2,0)),U)
I %="" W:$X ! W "***ERROR in Install, You need to remove the Distribution and reload it**",! S
XPDIQUIT=1 Q
S XPDT(P1)=P2_U_%,(XPDT("DA",P2),XPDT("NM",%))=P1 W:$X>64 ! W $J(%,15)
Q
;
QUES(XPDA) ;install questions; XPDA=ien in file 9.7
N XPDANS,XPDFIL,XPDFILN,XPDFILO,XPDFLG,XPDNM,XPDQUES,X,Y
S XPDNM=$P(^XPD(9.7,XPDA,0),U) W !!,"Install Questions for ",XPDNM,!
;pre-init questions
D DIR^XPDIQ("PRE") I $D(XPDIQUIT) D ASKABRT^XPDI Q
;file install questions
S (XPDFIL,XPDFLG)=0
F S XPDFIL=$O(^XTMP("XPDI",XPDA,"FIA",XPDFIL)) Q:'XPDFIL S X=^(XPDFIL),X(0)=^
(XPDFIL,0),X(1)=^(XPDFIL),XPDFILO=^(0,1) D Q:$D(XPDIQUIT)
.;check for DD screening logic
.I $G(^X(10))]" N XPDSR S XPDSR=^(10) ;^(10) is ref to ^XTMP("XPDI",XPDA,"FIA
",XPDFIL,0,10) from prev line
.;XPDFILN=file name^global ref^partial DD
.;XPDANS=new file^DD screen failed^Data exists^update file name^user
.;doesn't want to update data 1=yes,0=no
.S XPDFILN=X_X(0)_U_X(1),XPDANS='($D(^DIC(XPDFIL,0))#2)_'^^'_ '$O(@X(0)_0))
.I 'XPDFLG W !,"Incoming Files:" S XPDFLG=1
.W ! D DIR^XPDIQ("XPF",XPDFIL_#") Q:$D(XPDIQUIT)
.S:$G(XPDQUES("XPF",XPDFIL_#2))=0 $P(XPDANS,U,5)=1
.S ^XTMP("XPDI",XPDA,"FIA",XPDFIL,0,2)=XPDANS
.;kill the answers so we can re-ask for next file
.F I=1:1:2 K XPDQUES("XPF",XPDFIL_#_I)
;XPDIQUIT is by file questions in previous do loop, set in XPDIQ
I $D(XPDIQUIT) D ASKABRT^XPDI Q
;ask for coordinators to incoming mail groups
S (XPDFIL,XPDFLG)=0
F S XPDFIL=$O(^XTMP("XPDI",XPDA,"KRN",3.8,XPDFIL)) Q:'XPDFIL S X=^(XPDFIL,0),Y=$G(^(-1)) D
Q:$D(XPDIQUIT)
.;XPDANS=Mail Group name
.Q:$P(Y,U)=1 ;Don't ask if deleting
.S XPDANS=$P(X,U)
.I 'XPDFLG W !!,"Incoming Mail Groups:" S XPDFLG=1
.W ! D DIR^XPDIQ("XPM",XPDFIL_#") Q:$D(XPDIQUIT)
```

```

.;kill the answers so we can re-ask for next MG
.K XPDQUES("XPM" _XPDFIL_ "#1")
.Q
I $D(XPDQUIT) D ASKABRT^XPDI Q
;ask to rebuild menus if Option is added
S (XPDFIL,XPDFLG)=0
S XPDFIL=$O(^XTMP("XPDI",XPDA,"KRN",19,XPDFIL)) D:XPDFIL
.S X=^XTMP("XPDI",XPDA,"KRN",19,XPDFIL,0)
.;XPDANS=Menu Rebuild Answer
.S XPDANS=$P(X,U)
.W ! D DIR^XPDIQ("XPO") Q:$D(XPDQUIT)
I $D(XPDQUIT) D ASKABRT^XPDI Q
;post-init questions
W ! D DIR^XPDIQ("POS") I $D(DIRUT)!$D(XPDQUIT) D ASKABRT^XPDI Q
Q
;
XQSET(XPDA) ;get options & protocols to disable
;put in ^TMP($J,"XQ00",starting build name)
N A,I,X,Y
S I=0 F S I=$O(^XTMP("XPDI",XPDA,"KRN",19,I)) Q:'I S X=^(I,0),A=^(-1) D
.S Y=$O(^DIC(19,"B",$P(X,U),0))
.;check that option exist and 0=send,1=delete,3=merge or 5=disable
.I Y,$D(^DIC(19,Y,0)),$(A:1,1:A#2) S ^TMP($J,"XQ00",XPDSET,19,Y)=$P(^0,U,1,2)
S I=0 F S I=$O(^XTMP("XPDI",XPDA,"KRN",101,I)) Q:'I S X=^(I,0),A=^(-1) D
.S Y=$O(^ORD(101,"B",$P(X,U),0))
.I Y,$D(^ORD(101,Y,0)),$(A:3:1,A:5:1,1:'A) S ^TMP($J,"XQ00",XPDSET,101,Y)=$P(^0,U,1,2)
Q
;XPDIJ need to install XPDIJ now & set routine flag to skip
XPDIJ N DIE,XPDA,XCM,XCN,XCS,X
S XPDA=XPDIJ,DIE=^XTMP("XPDI",XPDIJ,"RTN","",XPDIJ,"",XCN=0,X="XPDIJ"
X ^%ZOSF("SAVE") D RTNLOG^XPDUTL("XPDIJ") ;Save and update ROUTINE file
S XCN=$$RTNUP^XPDUTL("XPDIJ",2)
Q

XPDI B
XPDI B ;SFISC/RSD - Backup installed Package ;05/04/2004 09:42
; ;8.0;KERNEL; **10,58,108,178,345** ;Jul 10, 1995;Build 1
EN ;
N XCNP,DIF,DIR,DIRUT,XMSUB,XMDUZ,XMDISPI,XMZ,XPD,XPDA,XPDNM,XPDQUIT,XPDST,XPDT,X,Y,%
S %="I '$P(^0,U,9),$D(^XPD(9.7,"ASP",Y,1,Y)),$D(^XTMP("XPDI",Y))",XPDST=$$LOOK^XPDI1(%)
Q:'XPDST!$D(XPDQUIT)
S DIR(0)="F^3:65"
S DIR("A")="Subject"
S DIR("?")="characters and must not contain embedded uparrow."
S DIR("?",1)="Enter the subject for this Packman Backup Message"
S DIR("?",2)="This response must have at least 3 characters and no more than 63"
S DIR("B")=$E(("Backup of "_$P(XPDT(1),U,2)" install on "_$FMTE^XLFD(T),1,63)
;p345-rename AND* to XPD*
I '$G(XPD AUTO) D ^DIR I $D(DIRUT) D QUIT^XPDI1(XPDST) Q
I '$G(XPD AUTO) S Y=DIR("B")
S XMSUB=Y,XMDUZ=+DUZ
D XMZ^XMA2 I XMZ<1 D QUIT^XPDI1(XPDST) Q
S Y=$$NOW^XLFD(T,%=$$DOW^XLFD(T,Y),Y=$$FMTE^XLFD(T,Y,2)
S X="PACKMAN BACKUP Created on "_%_" "_$P(Y,"@")_" at "_$P(Y,"@",2)
I $D(^VA(200,DUZ,0)) S X=X_" by "_$P(^0,U)_"
S:$D(^XMB("NAME")) X=X_"at "_$P(^("NAME"),U)_"
S ^XMB(3.9,XMZ,2,0)="^3.92A^_DT,^(1,0)=$TXT "_X,XCNP=1
S XPDT=0
F S XPDT=$O(XPDT(XPDT)) Q:'XPDT D
.S XPDA=+XPDT(XPDT),XPDNM=$P(XPDT(XPDT),U,2),XPD=""
.I '$D(^XTMP("XPDI",XPDA,"RTN")) W !,"No routines for ",XPDNM,! Q
.W !,"Loading Routines for ",XPDNM
.F S XPD=$O(^XTMP("XPDI",XPDA,"RTN",XPD)) Q:XPD="" D ROU(XPD) W ".
S:$G(XPD AUTO) XMY(XMDUZ)=" D EN3^XMD,QUIT^XPDI1(XPDST)
Q
;

```

```

ROU(X) N %N,DIF
X ^%ZOSF("TEST") E W !,"Routine ",X," is not on the disk." Q
S XCNP=XCNP+1,^XMB(3.9,XMZ,2,XCNP,0)="$ROU "_X_" (PACKMAN_BACKUP)",DIF="^XMB(3.9,XMZ,2,"
X ^%ZOSF("LOAD")
S $P(^XMB(3.9,XMZ,2,0),U,3,4)=XCNP_U_XCNP,^(XCNP,0)="$END ROU "_X_" (PACKMAN-BACKUP)"
Q

XPDIL
XPDIL ;SFISC/RSD - load Distribution Global ;05/05/2008
; ;8.0;KERNEL;**15,44,58,68,108,422,525**;Jul 10, 1995;Build 1
;
EN1 N POP,XPDA,XPDST,XPDIT,XPDT,XPDGP,XPDQUIT,XPDREQAB,XPDSKPE
S:'$D(DT) DT=$$DT^XLFD S:'$D(U) U="^"
S XPDST=0
D ST I $G(XPDQUIT) D ABRTALL^XPDIT(1) G NONE
;XPDST= starting Build
;XPDT("DA",ien)=seq # to install
;XPDT("NM",build name)=seq #
;XPDT(seq #)=ien^Build name
;XPDT("GP",global)= 1-replace, 0-overwrite^ien
;XPDGP=globals from a Global Package
;XPDSKPE=1 don't run Environment Check^has question been asked
S XPDIT=0,XPDSKPE="0^0"
F S XPDIT=$O(XPDT(XPDIT)) Q:'XPDIT S XPDA+=XPDT(XPDIT) D I '$D(XPDT) Q
.;check if this Build has an Envir. Check
.I $G(^XTMP("XPDIT",XPDA,"PRE"))]]"" D I $G(XPDQUIT) D ABRTALL^XPDIT(1) Q
.;quit if we already asked this question
..Q:$P(XPDSKPE,U,2)
..S $P(XPDSKPE,U,2)=1
..N DIR,DIRUT
..S DIR(0)="Y",DIR("A")="Want to RUN the Environment Check Routine",DIR("B")="YES"
..S DIR("A",1)="Build "_$P(XPDT(XPDIT),U,2)"_ has an Environmental Check Routine"
..D ^DIR I $D(DIRUT) S XPDQUIT=1 Q
..S:'Y XPDSKPE="1^1"
.D PKG^XPDIL1(XPDA)
;Global Package
G:$D(XPDGP) ^XPDIGP
I $D(XPDT),$D(^XPD(9.7,+XPDST,0)) W !,"Use INSTALL NAME: ",$P(^0,U)," to install this
Distribution.",!
Q
ST ;global input
N DIR,DIRUT,GR,IOP,X,Y,Z,%ZIS
G:'$D(^DD(3.5,0)) OPEN
I '$D(^%ZIS(1,"B","HFS")) W !,"You must have a device called 'HFS' in order to load a
distribution!",*7 S XPDQUIT=1 Q
D HOME^%ZIS
S DIR(0)="F^3:75",DIR("A")="Enter a Host File",DIR("?")="Enter a filename and/or path to input
Distribution."
D ^DIR I $D(DIRUT) S XPDQUIT=1 Q
S %ZIS="",%ZIS("HFSNAME")=Y,%ZIS("HFSMODE")="R",IOP="HFS"
D ^%ZIS I POP W !,"Couldn't open file or HFS device!!",*7 S XPDQUIT=1 Q
;don't close device if we have a global package, we need to bring in the globals now
D GI,^%ZISC:$D(XPDGP)!$G(XPDQUIT)
Q
;
;if no device file, Virgin Install
OPEN ;use open command
N IO,IOPAR,DIR,DIRUT,DTOUT,DUOUT
S DIR(0)="F^1:79",DIR("A")="Device Name"
S DIR("?",1)="Device Name is either the name of the HFS file or the name of the HFS
Device.",DIR("?",2)="i.e. for MSM enter 51",DIR("?",3)=" for DSM enter
DISK$USER:[ANONYMOUS]:KRN8.KID"
D ^DIR I $D(DIRUT) S POP=1 Q
S IO=Y,DIR(0)="FO^1:79",DIR("A")="Device Parameters"
S DIR("?",1)="Device Parameter is the Open parameter this M operating system needs
to",DIR("?",2)="open the Device Name.",DIR("?",3)="i.e. for MSM enter ("B:

```

```

\KRN8.KID": "R")",DIR("?")="      for DSM enter  READONLY"
D ^DIR I $D(DTOUT)!$D(DUOUT) S POP=1 Q
S IOPAR=Y
X "O IO: "_IOPAR_" :10" E  U $P W !,"Couldn't open ",IO S POP=1 Q
S IO(0)=$P
D GI D ^%ZISC
Q
;
GI N X,XPDSQ,Y,Z
U IO R X:10,Y:10 ;rwf was :0
U IO(0) W !!,X!, "Comment: ",Y
S XPDST("H")=Y,XPDS("H1")=Y_ ;Created on "_$P(X,"KIDS Distribution saved on ",2)
;Z is the string of Builds in this file
U IO F X=1:1 R Z:1 S Z=$P(Z,"**KIDS**",2,99) Q:Z="" S X(X)=Z
U IO(0) I $G(X(1))="" W !!,"This is not a Distribution HFS File!" S XPDQUIT=1 Q
;global package, set XPDGP=flag;global^flag;global^... flag=1 replace
I $P(X(1),":")="GLOBALS" S XPDGP=$P(X(1),U,2,99),X(1)=$P(X(1),U)
S XPDIT=0,X(1)=$P(X(1),":",2,99)
W !!,"This Distribution contains Transport Globals for the following Package(s):"
F X=1:1:X-1 F Z=1:1 S Y=$P(X(X),U,Z) Q:Y="" D Q:$G(XPDQUIT)
. ;can't install if global exist, that means Build never finish install
. ;INST will show name
. S XPDIT=XPDIT+1 I '$$INST^XPDIL1(Y) S XPDQUIT=1 Q
Q:$G(XPDQUIT)
W !,"Distribution OK!",!
D:$D(XPDGP) DISP^XPDIGP
S DIR(0)="Y",DIR("A")="Want to Continue with Load",DIR("B")="YES"
D ^DIR I $D(DIRUT)!Y S XPDQUIT=1 Q
W !,"Loading Distribution...",!
;reset expiration date to T+7 on transport global
S ^XTMP("XPDI",0)=$$FMADD^XLFTD(DT,7)_U_DT
;start reading the HFS again
U IO R X:10,Y:10 ;rwf was :0
;the next read must be the INSTALL NAME
I X'="**INSTALL NAME**"!$D(XPDT("NM",Y)) U IO(0) W !!,"ERROR in HFS file format!" S XPDQUIT=1 Q
;XPDSQ is the disk sequence number
S %=$P(XPDT("NM",Y),GR="^XTMP("XPDI",Y),Y,XPDSQ=1
;X=global ref, Y=global value. DIRUT is when user aborts
;rwf next line was :0
F R X:10,Y:10 Q:X="**END**" D I $D(DIRUT) S XPDQUIT=1 Q
.I X="**INSTALL NAME**" D Q
..S %=$G(XPDT("NM",Y)) I '% S DIRUT=1 Q
..S GR="^XTMP("XPDI",Y),Y,XPDSQ=1
.S @(GR_X)=Y
U IO(0)
Q
;
NONE W !!,"**NOTHING LOADED**",!
Q

XPDI1
XPDI1 ;SFISC/RSD - cont. of load Distribution Global ;05/05/2008
;8.0;KERNEL;**15,17,39,41,44,66,68,76,85,100,108,229,525**;Jul 10, 1995;Build 1
; Per VHA Directive 2004-038, this routine should not be modified.
;
PKG(XPDA) ;check Package file
N XPD,XPDCP,XPDNM,XPDNOQUE,XPDPKG,X,Y,%
S XPDNM=$P(XPDT(XPDIT),U,2) W !?3,XPDNM
;check KIDS version against sites version, skip if package is Kernel
I $$PKG^XPDUTL(XPDNM) ["KERNEL" D I $D(XPDQUIT) D ABORT^XPDI(XPDA,1) Q
.;this is part of a Kernel multi package
.Q:$O(XPDT("NM","KERNEL")) ["KERNEL"
.S Y=$G(^XTMP("XPDI",XPDA,"VER"))
.I $$VERSION^XPDUTL("XU")<Y W !!,"But I need Version ",+Y," of KERNEL!" S XPDQUIT=1
.I $$VERSION^XPDUTL("VA FILEMAN")<$P(Y,U,2) W !,"But I Need Version ",+$P(Y,U,2)," of VA FILEMAN!"
S XPDQUIT=1

```

```

;get national package name
S %=$0(^XTMP("XPDI",XPDA,"PKG",0)),XPDPKG(0)=$G(^(+%,0)),XPDPKG=%
;XPDPKG=new ien^old ien
I XPDPKG D S XPDPKG=+Y_U_XPDPKG
.N D,DIC
.S DIC="^DIC(9.4,"",DIC(0)="X",X=$P(XPDPKG(0),U)
.D ^DIC Q:Y>0
.;if lookup fails try C & C2 x-ref
.S X=$P(XPDPKG(0),U,2),DIC(0)="MX",D="C^C2" D MIX^DIC1
;add package to Install file
I XPDPKG>0 S XPD(9.7,XPDA_"",1)=+XPDPKG D FILE^DIE("", "XPD")
;XPDSKPE= does site want to run Environ. Check
I '$G(XPDSKPE) Q:$ENV(0)=1
;global package can't have pre or post inits
Q:$D(XPDGP)
;create pre-init checkpoint
S XPDCP="INI" I '$$NEWCP^XPDUTL("XPD PREINSTALL COMPLETED") D ABORT^XPDI(XPDA,1) Q
S %=$$INRTN("INI")
;check for routine, use as call back
I $L(%),'$$NEWCP^XPDUTL("XPD PREINSTALL STARTED",%) D ABORT^XPDI(XPDA,1) Q
;create post-init checkpoint
S XPDCP="INIT" I '$$NEWCP^XPDUTL("XPD POSTINSTALL COMPLETED") D ABORT^XPDI(XPDA,1) Q
S %=$$INRTN("INIT")
I $L(%),'$$NEWCP^XPDUTL("XPD POSTINSTALL STARTED",%) D ABORT^XPDI(XPDA,1) Q
;create fileman and components check points and file rest of data
D XPCK^XPDIK("FIA"),XPCK^XPDIK("KRN")
Q
INST(XPDNM) ;add to Install file
N %X,DIC,DIR,DIRUT,DLAYGO,X,XPDI,XPDA,XPDIE,XPDDIQ,Y,SH
;check if Build was already installed
;XPD=0 abort install, else XPD=ien in Install file
I $D(^XPD(9.7,"B",XPDNM)) S (SH,Y)=0 D Q:$D(XPD) XPD
. W !,"Build ",XPDNM," has been loaded before, here is when: "
. F S Y=$0(^XPD(9.7,"B",XPDNM,Y)) Q:'Y D
.. Q:'$D(^XPD(9.7,Y,0)) S %="^(0)
.. W !?6,$P(%U)," "
.. I $P(%U,9)<3,$D(^XTMP("XPDI",Y)) W "***Transport Global already exists***",*7 S XPD=0 Q
.. S %X=$X W $$EXTERNAL^DILFD(9.7,.02,"", $P(%U,9)),!?%X,"was loaded on ",$$FMT
E^XLFD($P($G(^XPD(9.7,Y,1)),U))
. ;quit if transport global exist
. Q:$D(XPD)
. S DIR(0)="Y",DIR("A")="OK to continue with Load",DIR("B")="NO"
. D ^DIR W ! I $D(DIRUT)!'Y S XPD=0 Q
S DIC="^XPD(9.7,"",DIC(0)="XL",DLAYGO=9.7,X="""_XPDNM_""""
;add to Install file, must be new
D ^DIC
I Y<0 S SH=0 W !,"Can't add Build ",XPDNM," to Install File" Q 0
;set starting package to Y, if it is not already defined
S:'XPDST XPDST=+Y
;XPD array keeps track of all packages in this distribution
S XPDA=+Y,XPD(XPDIT)=XPDA_U_XPDNM,(XPD("DA",XPDA),XPD("NM",XPDNM))=XPDIT
S %="XPDIE(9.7,"""_XPDA_","""",@%@(0.02)=0,@%@(2)=$$NOW^XLFD,@%@(3)=XPDST,@%@(4)
)=XPDIT,@%@(5)="",@%@(6)=XPDST("H1")
D FILE^DIE("", "XPDIE")
I '$D(SH) W !?3,XPDNM ;SH is set when some other part of INST shows the name
Q XPDA
;
;XPDQUIT quit current package install, 1=kill global, 2=leave global
;XPDQUIT(package) quit package install, 1=kill, 2=leave
;XPDABORT quit the entire distribution, 1=kill, 2=leave
;XPDENV 0=loading distribution, 1=installing
ENV(XPDENV) ;environment check & version check
;returns 0=ok, 1=rejected kill global, 2=rejected leave global
N %,DIR,XPDI,XPDQUIT,XPDABORT,XPDONE,XPDGREF,XPDMREQ
M X=DUZ N DUZ M DUZ=X S DUZ(0)="@" ;See that ENV check has full FM priv.
S XPDGREF="^XTMP("XPDI","_XPDA_", "TEMP")"

```

```

S XPDMBREQ=$G(^XTMP("XPDI",XPDA,"MBREQ"))
S $P(^XPD(9.7,XPDA,0),U,7)=XPDMBREQ
;check version number
I XDPKPG>0 D I $G(XPDQUIT) D ABORT^XPDI(XPDA,1) Q 1
.N DIR,DIRUT,X,Y
.S %+=$$VER^XPDUTL(XPDNM),Y+= $G(^DIC(9.4,+XDPKPG,"VERSION")),X=XPDNM["*"]
.;if patch, version must be the same
.I X,%'=Y W !,"This Patch is for Version ",%," , you are running Version ",Y,! S
XPDQUIT=1
.;if package, version must be greater or equal
.I 'X,%<Y W !,"You have a Version greater than mine!",! S XPDQUIT=1
.Q: $G(XPDQUIT)
.I $G(XPDMBREQ) D S XPDQUIT=0,XPDDONE=1 Q
. . D MES^XPDUTL("***ABORT** Required Build "_XPDNM_", did not pass internal KID
S checks!"),ABRTALL^XPDI(1),NONE^XPDI
. . Q
.S DIR(0)="Y",DIR("A")="Want to continue installing this build",DIR("B")="NO"
.D ^DIR I Y K XPDQUIT
.Q
Q:$G(XPDDONE) 1
S %=$$REQB I % S (XPDABORT,XPDREQAB)=% G ABORT
S %=$G(^XTMP("XPDI",XPDA,"PRE")) D:%]""
.W !,"Will first run the Environment Check Routine, ",%,!
.D SAVE^XPDIJ(%),@(" "_%)
ABORT I $G(XPDABORT) D Q XPDABORT
.;if during load & leave global quit
.I 'XPDENV,XPDABORT=2 Q
.D ABRTALL^XPDI(XPDABORT)
Q:$D(XPDQUIT) 0
I $G(XPDQUIT) D ABORT^XPDI(XPDA,XPDQUIT)
S XPDI=""
;don't do if loading & leave global, need to keep XPDT(array)
F S XPDI=$O(XPDQUIT(XPDI)) Q:XPDI="" D: '(XPDQUIT(XPDI)=2&'XPDENV)
.S %=$G(XPDT("NM",XPDI)) D:% ABORT^XPDI(+XPDT(%),XPDQUIT(XPDI))
S XPDQUIT=$S($G(XPDQUIT):XPDQUIT,'$O(XPDT(0))!'$D(^XTMP("XPDI",XPDA)):1,1:0)
Q XPDQUIT
;
REQB() ;check for Required Builds
;returns 0=ok, 1=failed kill global, 2=failed leave global
N XPDACT,XPDBLD,XPDI,XPDQ,XPDQUIT,XPDY,XPDY0,X,Y
S XPDBLD=$O(^XTMP("XPDI",XPDA,"BLD",0)),XPDQUIT=0,XPDI=0
Q:$D(^XTMP("XPDI",XPDA,"BLD",XPDBLD,"REQB")) 0
F S XPDI=$O(^XTMP("XPDI",XPDA,"BLD",XPDBLD,"REQB",XPDI)) Q:'XPDI S XPDY0=^(XPDI,0) D
.S XPDQ=0,XPDY=$P(XPDY0,U),XPDACT=$P(XPDY0,U,2),X=$$PKG^XPDUTL(XPDY),Y=$$VER^XP
DUTL(XPDY),Z=$$VERSION^XPDUTL(X)
.;quit if current version is greater than what we are checking for
.Q:Z>Y
.I XPDY["*" S:Z<Y XPDQ=2
.E S:'$$PATCH^XPDUTL(XPDY) XPDQ=1
.;quit if patch is already on system
.Q:'XPDQ
.;quit if patch is sequenced prior within this build
.I $D(XPDT("NM",XPDY)),(XPDT("NM",XPDY)<XPDT("NM",XPDNM)) S XPDQ=0 Q
.S XPDQUIT=$S(XPDACT>XPDQUIT:XPDACT,1:XPDQUIT)
.;XPDACT=0 warning, =1 abort & kill global, =2 abort
.W !,$S(XPDACT:"**INSTALL ABORTED**",1:"**WARNING**")_$$S(XPDQ=1:" Patch ",1:"
Package ")_XPDY_" is Required "_$$S(XPDACT:"to install",1:"for")_" this package!!",!
Q:'XPDQUIT 0
;don't do if leave global and loading
D: '(XPDQUIT=2&'XPDENV) ABORT^XPDI(XPDA,XPDQUIT)
Q XPDQUIT
;return a routine that can be run
INRTN(X) N Y
S Y=$G(^XTMP("XPDI",XPDA,X)) Q:Y="" ""
S Y=$$S(Y["^":Y,1:"^"_Y)
Q Y

```

XPDIPM

```

XPDIPM ;SFISC/RSD - Load a Packman Message ;05/04/2004 09:43
;;8.0;KERNEL;**21,28,68,108,345**;Jul 05, 1995;Build 1
Q: '$D(^XMB(3.9,+$(XMB(3.9,XMZ,2,XPD))) Q: +XPD'=XPD S X=^(XPD,0) I $(X,1,11)="$TXT $KIDS " Q
S Y=$P(X,"$KIDS ",2)
EN I 'XPD!'$L(Y) W !,"Couldn't find a KIDS package!!",*7 Q
N DIR,DIRUT,GR,XPDA,XPDST,XPDIT,XPDT,XPDNM,XPDQUIT,XPDREQAB
S XPDST("H1")=$P(^XMB(3.9,XMZ,0),U),XPDST=0,XPDIT=1
S XPDA=$$INST^XPDIL1(Y) G: 'XPDA NONE^XPDIL
W !
S DIR(0)="Y",DIR("A")="Want to Continue with Load",DIR("B")="YES"
;p345-rename AND* to XPD*
I '$G(XPDAUTO) D ^DIR S: '$G(XPDAUTO) Y=1 I 'Y!$D(DIRUT) D ABRTALL^XPDIL1 G NONE^XPDIL
W !,"Loading Distribution...",!
S ^XTMP("XPDIL",0)=$$FMADD^XLFD(T,7)_U_DT
D GI I '$G(XPDQUIT) D ABRTALL^XPDIL1 G NONE^XPDIL
D PKG^XPDIL1(XPDA)
Q
GI D NXT Q: '$G(XPDQUIT)
I X'="**INSTALL NAME**"! '$D(XPDT("NM",Y)) S XPDQUIT=1 Q
S GR="^XTMP("XPDIL",0)_"_XPDA_", "
F D NXT Q: X="!" '$D(XPDQUIT) D
.S @ (GR_X)=Y
Q
NXT S (X,Y)="",XPD=$0(^XMB(3.9,XMZ,2,XPD)) G: +XPD'=XPD ERR S X=^(XPD,0)
I $(X,1,5)="$END " S X="" Q
S XPD=$0(^XMB(3.9,XMZ,2,XPD)) G: +XPD'=XPD ERR
S Y=^XMB(3.9,XMZ,2,XPD,0)
Q
XMP2 ;called from XMP2
N X,XPD,Y
S XPD=XCN,X=$G(^XMB(3.9,XMZ,2,XPD,0)),Y=$P(X,"$KID ",2)
D EN
S XMOUT=1
Q
ERR W !,"Error in Packman Message, ABORTING load!!"
S (X,Y)="",XPDQUIT=1
Q

```

XPDIIQ

```

XPDIIQ ;SFISC/RSD - Install Questions ;03/21/2008
;;8.0;KERNEL;**21,28,58,61,95,108,399**;Jul 10, 1995;Build 1
Q
DIR(XPFR,XPFP) ;XPFR=prefix, XPFP=file no._# or Mail Group ien
;XPFP is for XPF or XPM questions
N DIR,DR,XPDI,XPDJ,X,Y,Z
S XPFP=$G(XPFP),XPDI=$S(XPFP:XPFR_XPFP,1:XPFR)
D QUES(XPDI)
;ask questions
S X=XPFR
F S X=$0(^XTMP("XPDIL",XPDA,"QUES",X)),Z="" Q: X="!" ($P(X,XPFR)) D I $D(DIRUT) S XPDQUIT=1 Q
.S XPDJ=$S('XPFP:X,1:XPDI_$P(X,XPFR,2))
.F S Z=$0(^XTMP("XPDIL",XPDA,"QUES",X,Z)) Q: Z="" M DIR(Z)=^(Z)
; if there was a previous answer, reset DIR("B") to external or internal answer
.S: $L($G(XPDQUES(XPDJ))) DIR("B")=$G(XPDQUES(XPDJ,"B"),XPDQUES(XPDJ)) D Q: '$D(Y)
..N FLAG,X,Z K Y
..;this is the M CODE node that was set to DIR("M") in prev for loop
..;FLAG is used by KIDS questions
..I $D(DIR("M")) S %=DIR("M"),FLAG="" K DIR("M") X %
..Q: '$D(DIR)
..;'|' is used to mark variable in prompt, reset prompt with value of variable
..S: '$G(DIR("A"))["|" DIR("A")=$P(DIR("A"),"|")_@$P(DIR("A"),"|",2)_$P(DIR("A"),"|",3)
..K: '$G(DIR("B"))="" DIR("B")

```

```

..D ^DIR
.S %=$P(DIR(0),U)
.;read was optional and didn't timeout and user didn't enter anything
.I %["0",'$(DTOUT),$S(%["P":Y=-1,1:Y="") K DIRUT Q
.;quit if the user up-arrowed out
.Q:$(DIRUT)
.;if pointer, reset Y & Y(0)
.I %["P" S Y(0)=$S(%["Z":$P(Y(0),U),1:$P(Y,U,2)),Y=+Y
.;if Y(0) is not defined, but Y is
.S:$(Y)#2&'$(Y(0))#2) Y(0)=Y
.S XPDQUES(XPDJ)=Y,XPDQUES(XPDJ,"A")=$G(DIR("A")),XPDQUES(XPDJ,"B")=$G(Y(0))
.K DIR
K XPDJ S XPDI=XPFR
;code to save XPDQUES to INSTALL ANSWERS in file 9.7, loop thru the answers starting with the from
value, XPFR
F Y=1:1 S XPDI=$O(XPDQUES(XPDI)) Q:XPDI="!"($P(XPDI,XPFR))" D
.S X="XPDJ(9.701,""?+"_Y_", "_XPD_", "")",@X@(.01)=XPDI,@X@(1)=$G(XPDQUES(XPDI,"
A")),@X@(2)=$G(XPDQUES(XPDI,"B")),@X@(3)=XPDQUES(XPDI)
K XPDI D:$(XPDJ)>9 UPDATE^DIE("","XPDJ","XPDI")
Q
;
QUES(X) ;build XPDQUES array, X="INI","INIT","XPF","XPM"
;move INSTALL ANSWERS from file 9.7 to XPDQUES
;XPDQUES(X)=internal answer, XPDQUES(X,"A")=prompt, XPDQUES(X,"B")=external answer.
N Y,Z K XPDQUES S Z=X
F S Z=$O(^XPD(9.7,XPDA,"QUES","B",Z)) Q:Z="!"($P(Z,X))" S Y=$O(^Z,0) D
.Q:'$D(^XPD(9.7,XPDA,"QUES",Y,0))
.S XPDQUES(Z)=$G(^Z,1),XPDQUES(Z,"A")=$G(^("A")),XPDQUES(Z,"B")=$G(^("B")) ; ^Z(1) refer to prev
line ^XPD(9.7,XPDA,"QUES","B",Z)
Q
;
ANSWER(QUES) ;E.F. Return answer to question
N IEN I '$D(XPDA)!($G(QUES)="") Q ""
S IEN=$O(^XPD(9.7,XPDA,"QUES","B",QUES,0)) I IEN>0 Q ""
Q $G(^XPD(9.7,XPDA,"QUES",IEN,1))
;codes for install process questions
;XPDFIL=file #, XPDFILN=file name^global ref^partial DD
;XPDFILO=update DD^security codes^^^resolve pt^list template^data with
file^add,merge,overwrite,replace^user override data update
;XPDSCR=screen to determine DD update
;XPDANS is define in QUES^XPDI
XPF1 ;write over existing file
N XPDI
W !!?3,XPDFIL,?13,$P(XPDFILN,U),$P(" (Partial Definition)",U,$P(XPDFILN,U,3)),$P(" (including
data)",U,$P(XPDFILO,U,7))="y")
;file doesn't exists
I XPDANS K DIR Q
I $L($G(XPDSCR)) S XPDI=1 D Q:'XPDI
.X XPDSCR S XPDI=$T Q:XPDI
.W !,"Data Dictionary FAILED the screening logic, file will NOT be installed!"
.S $P(XPDANS,U,2)="1" K DIR
S FLAG=$P($G(^DIC(XPDFIL,0)),U)
;file exist and has the same name
I $P(FLAG,$P(XPDFILN,U))="" W !,"Note: You already have the '", $P(XPDFILN,U),"' File." K DIR Q
W *7,!,"*BUT YOU ALREADY HAVE '",FLAG,"' AS FILE #",XPDFIL,"!"
S $P(XPDANS,U,4)=1
Q
XPF2 ;data
;if they don't want to overwrite a file with a different name then set the DIRUT flag and ABORT,
this will stop the rest of the questions and abort the install
I $G(XPDQUES("XPF"_XFP1))=0 S DIRUT=1 K DIR Q
;if Data doesn't exists or DD failed screen or data wasn't sent, don't ask question
I '$P(XPDANS,U,3)!$P(XPDANS,U,2)!($P(XPDFILO,U,7))="y") K DIR Q
S %=$F("amor",$P(XPDFILO,U,8))-1
;if this is add and file is not new
I %=1 W !,"Data will NOT be added." K DIR Q

```



```

;check if dev. doesn't want to ask user
I $P(XPDFILO,U,9)!='y" W !,"I will ",$P("^MERGE^OVERWRITE^REPLACE",U,%)," your data with mine." K
DIR Q
S FLAG=$P("^merged with^to overwrite^to replace",U,%)
Q
;XPDDIQ(name)=internal value, (name,"A")=prompt, (name,"B")=external XPQ(NM) ;Build XPDDIQ
Q: '$D(XPDDIQ(NM))
I $D(XPDDIQ(NM))#2 S XPDQUES(NM)=XPDDIQ(NM) K DIR Q
S:$D(XPDDIQ(NM,"A")) DIR("A")=XPDDIQ(NM,"A")
S:$D(XPDDIQ(NM,"B")) DIR("B")=XPDDIQ(NM,"B")
Q
XPI1 ;Inhibit Logons
D XPQ("XPI1")
Q
XPM1 ;mail groups
S FLAG=XPDANS
;DIR("B") is null if first time here
I DIR("B")="" D
.;check if mail group already exist
.S X=$$FIND1^DIC(3.8,"","XQf",XPDANS,"","")
.;get the current coordinator
.Q: 'X S X=$P($G(^XMB(3.8,X,0)),U,7)
.;set the default to current coordinator
.I X,$D(^VA(200,X,0))#10 S DIR("B")=$P(^ (0),U)
D XPQ("XPM1")
Q
XP01 ;rebuild menu trees
D XPQ("XP01")
Q
XPZ1 ;disable options
D XPQ("XPZ1")
Q
XPZ2 ;move routines
N Y
;if they are not in production UCI don't ask
X ^%ZOSF("UCI") I Y'^%ZOSF("PROD") K DIR Q
;if they are not running MSM don't ask
I ^%ZOSF("OS")'["MSM" K DIR Q
Q: '$D(XPDDIQ("XPZ2"))
I $D(XPDDIQ("XPZ2"))#2 S XPDQUES("XPZ2")=XPDDIQ("XPZ2") K DIR Q
S:$D(XPDDIQ("XPZ2","A")) DIR("A")=XPDDIQ("XPZ2","A")
S:$D(XPDDIQ("XPZ2","B")) DIR("B")=XPDDIQ("XPZ2","B")
Q

XQ00
XQ00 ;SEATTLE/LUKE - Out Of Order, Man ;9/13/96 09:21
;8.0;KERNEL;**10,21,47,520**;Jul 03, 1995;Build 1
;Per VHA Directive 2004-038, this routine should not be modified.
INIT(XQSET) ;Call for Out-of-order set creation, called by KIDS
;
;The variable XQSET should be null if this is the first pass
;or if KIDS thinks the user wants a new set of options
;
S XQINI="",XQK=0 ;XQINI used as a flag to see if it's KIDS calling
I XQSET]"" S: '$D(^XTMP("XQ00",XQSET,0)) XQSET="^"
I XQSET="^" G OUT
I XQSET]"" S XQMESS=$P(^XTMP("XQ00",XQSET,0),U) G ASK1
;
EN ;Entry point for Define Out Of Order Options Set option
S XQK=0,U="^",XQSET=U
;
NAME ;Get name for this option set
W !!,"Enter a short name for this set of options and or protocols: " R XQSET:DTIME S: '$T XQSET=U
G: XQSET=U OUT
I XQSET="?" W !!,"Enter a name of 20 characters or less for this set, '^' to quit, or '??' for
help" G NAME

```

```

I XQSET["???" S XQH="XQ00-NAME" D EN^XQH G NAME
I XQSET=""!($L(XQSET)>20) W !!, "Out-of-order sets must be named with 20 or less characters. Enter
'^' to quit." G NAME
I $D(^XTMP("XQ00",XQSET,0)) D G:$D(DIRUT) OUT G:Y=0 NAME G ASK1
.S XQMESS=$P(^XTMP("XQ00",XQSET,0),U)
.W !!, "WARNING: The Out-of-order set '"',XQSET,'" already exists.",!
.S DIR("A")="Do you want to modify it? (Y/N) " S DIR(0)="Y",DIR("B")="YES" D ^DIR K DIR
.Q
;
MESS ;Get the Out Of Order Message
R !!, "What should the Out Of Order message text be? :",XQMESS:DTIME S:'$T XQMES
S=U G:XQMESS=U OUT
I XQMESS="" W !!, "This is the message that will be shown with the options/protocols",!,"that are
made out of order. For instance, ""Laboratory install in progress"" G MESS
I XQMESS["???" S XQH="XQ00-MESS" D EN^XQH G MESS
;
ASK1 S XQFIL=19
ASK ;Get options to mark
S (XQ,XQN)=""
W !!, "Enter "_$S(XQFIL=19:"options",1:"protocols")_" you wish to mark as 'Out Of Order': "
R XQ:DTIME S:'$T XQ=U G:XQ=U OUT G:XQ="" SET
I XQ=""?" D G ASK
.W !?5, "Enter "_$S(XQFIL=19:"an option",1:"a protocol")_" name,"
.W !?5, "a name preceded by a minus sign to remove "_$S(XQFIL=19:"an option",1:"a protocol"),
.W !?5,$$S(XQFIL=19:"'^PR'",1:"'^OP'")_" to switch to "_$S(XQFIL=19:"protocols",1:"options"),
.W !?5, "an uparrow (that is '^') to quit,"
.W !?5, "or '??' for more help."
.Q
I XQ["???" S XQH="XQ00" D:XQ=""?" EN^XQH D:XQ=""?" LIST D:XQ=""?" LSTFIL S XQH="XQ00-MAIN"
D:XQ=""?"?"?"?" EN^XQH G ASK
I $E(XQ,1,3)=""^OP"!($E(XQ,1,3)=""^op") S XQFIL=19,XQSWTCH="" G ASK
I $E(XQ,1,3)=""^PR"!($E(XQ,1,3)=""^pr") S XQFIL=101,XQSWTCH="" G ASK
S XQDEL=0 I $E(XQ,1)="-" S XQDEL=1,XQ=$E(XQ,2,99)
I XQ=""",XQDEL K ^XTMP("XQ00",XQSET,XQFIL) W !!, "All "_$S(XQFIL=19:"options",1:
"protocols")_" removed. Start again or '^' to quit. " G ASK
I XQ="" S XQSTART=1,XQEND="ZZZZ" D FIND G ASK
I XQ?.E1"" S XQSTART=$E(XQ,1,$L(XQ)-1),XQEND=XQSTART_$C(127) D FIND G ASK
;Get a range of options allowing for name with hyphens in them
I XQ?1E.E1"-1E.E S XQRNG=0 D G:'XQRNG ASK
.;Name has hyphen, echo back the name and quit
.S X=XQ,DIC=XQFIL,DIC(0)="EZ" D ^DIC I Y>0 S XQ=$P(Y,U,2),XQRNG=1 Q
.;It is a range, build prompt to verify range
.W ! K DIR S DIR("A")="Do mean the "_$S(XQFIL=19:"options",1:"protocols")_" from "_$P(XQ,"-")_" to
"$P(XQ,"-",2)_"? (Y/N)",DIR(0)="YA" D ^DIR K DIR I Y S (XQN,XQSTART)=$P(XQ,"-",1),XQEND=$P(XQ,"-
",2) D FIND
.Q
;
I XQ?'1E.E1"-1E.E S X=XQ,DIC=XQFIL,DIC(0)="MEZ" D ^DIC S:Y'<0 XQ=$P(Y,U,2) I Y<0 W " ??",*7 G ASK
I XQDEL K ^XTMP("XQ00",XQSET,XQFIL,+Y) G ASK
S:$E(Y(0),1,4)=""XQ00" ^XTMP("XQ00",XQSET,XQFIL,+Y)=$P(Y(0),U)_U_$P(Y(0),U,2) G ASK
;
FIND ;Find first option in wildcard list
S XQN="" S:$L(XQSTART)>2 XQN=$E(XQSTART,1,$L(XQSTART)-1)
I XQFIL=19 F XQI=0:0 S XQN=$O(^DIC(XQFIL,"B",XQN)) Q:XQN=""!($E(XQN,1,$L(XQSTART))=XQSTART)
E F XQI=0:0 S XQN=$O(^ORD(101,"B",XQN)) Q:XQN=""!($E(XQN,1,$L(XQSTART))=XQSTART)
I XQN="" W !!, " No such ",$$S(XQFIL=19:"option(s)",1:"protocol(s).") Q
S XQSTART=XQN
;
FINDR I XQFIL=19 S XQON=$O(^DIC(XQFIL,"B",XQN,0)),XQON0=^DIC(XQFIL,+XQON,0)
E S XQON=$O(^ORD(XQFIL,"B",XQN,0)),XQON0=^ORD(XQFIL,+XQON,0)
I XQDEL D DELET Q
;
GET ;Get the first option selected and put it in ^XTMP
S XQN=XQSTART I $E(XQON,1,4)=""XQ00" S ^XTMP("XQ00",XQSET,XQFIL,+XQON)=$P(XQON0
,U)_U_$P(XQON0,U,2),XQK=XQK+1
S DIC=XQFIL,DIC(0)="MZ"

```

```

NEXT ;Find the rest of the options in this range and do likewise
  I XQFIL=19 F Q:XQN=XQEND S XQN=$O(^DIC(XQFIL,"B",XQN)) Q:XQN="!"(XQN]XQEND)
S XQON=$O(^DIC(XQFIL,"B",XQN,0)),XQON0=^DIC(XQFIL,+XQON,0) I $E(XQON,1,4)'="XQ00" S
^XTMP("XQ00",XQSET,XQFIL,+XQON)=$P(XQON0,U)_U_$P(XQON0,U,2),XQK=XQK+1
  E F Q:XQN=XQEND S XQN=$O(^ORD(XQFIL,"B",XQN)) Q:XQN="!"(XQN]XQEND) S XQON=$
O(^ORD(XQFIL,"B",XQN,0)),XQON0=^ORD(XQFIL,+XQON,0) I $E(XQON,1,4)'="XQ00" S ^XTM
P("XQ00",XQSET,XQFIL,+XQON)=$P(XQON0,U)_U_$P(XQON0,U,2),XQK=XQK+1
  Q
;
DELET ;Delete option(s) from the list in ^XTMP
;W !,XQON," ",XQSTART," ",XQDEL
S XQN=XQSTART,XQDEL=0
  I XQFIL=19 F K ^XTMP("XQ00",XQSET,XQFIL,+XQON) S XQN=$O(^DIC(XQFIL,"B",XQN)),XQX=XQK-1
Q:XQN="!"(XQN]XQEND) S XQON=$O(^DIC(XQFIL,"B",XQN,0))
  E F K ^XTMP("XQ00",XQSET,XQFIL,+XQON) S XQN=$O(^ORD(XQFIL,"B",XQN)),XQX=XQK-1
Q:XQN="!"(XQN]XQEND) S XQON=$O(^ORD(XQFIL,"B",XQN,0))
  Q
;
REMOV R !,"Remove all options previously selected? ",XQ:DTIME S:'$T XQ=U G:XQ[U
OUT I XQ["N!"(XQ["n") W !,"OK, you may continue." G ASK
K ^XTMP("XQ00",XQSET)
  Q
;
LSTFIL ;Show Option File
N XQE,XQR,XQS
D RANGE^XQ002(.XQS,.XQE,.XQR) I XQR D BXREF^XQ002(XQS,XQE)
  Q
;
LIST ;List users and options selected so far.
W @IOF S (XQT,XQM)=0
F XQFIL0=19,101 D
.S XQT=0,XQN=0,XQN=$O(^XTMP("XQ00",XQSET,XQFIL0,XQN)) I XQN="" W !,"No "_$S(XQFIL0=19:"menu
options",1:"protocols")_" selected yet" Q
.W !,"You will place Out Of Order the following "_$S(XQFIL0=19:"options:",1:"protocols:"),! F
XQI=0:0 D:$Y+3>IOSL WAIT Q:XQ=U W !,$P(^XTMP("XQ00",XQSET,XQFIL0,XQN),U,2)" ["_$P(^XQN),U)_"
(IEN = "_XQN_" ) S XQN=$O(^XQN) Q:XQN=""
.Q
  Q
;
WAIT ;Skip to the head of the next page
I 1 S XQ="" R:IOST["C-" !,"Press RETURN to continue, or '^' to quit.",XQ:DTIME
S:'$T XQ=U W @IOF
  Q
;
SET ;Set 0th node in ^XTMP global
I XQFIL=19,$D(XQSWTCH) S XQFIL=101 G ASK
D ^XQDATE
I $D(^XTMP("XQ00",XQSET,0)) S XQMESS=$P(^XQSET,0,U)
S ^XTMP("XQ00",XQSET,0)=XQMESS_U_%Y_U_$P(^VA(200,DUZ,0),",",")
S ^XTMP("XQ00",0)=DT_U_DT+7
;
OUT ;Clean up
I '$D(XPDNM),$D(^XTMP("XQ00",XQSET,0)),$D(^XTMP("XQ00",XQSET)) D
.;Temporary Fix: ^ at protocol prompt leaves partial set (no 0th node)
.S DIR(0)="Y",DIR("B")="YES"
.S DIR("A")="Delete this set of options? (Y/N) "
.D ^DIR
.I Y K ^XTMP("XQ00",XQSET)
.E D ^XQDATE S ^XTMP("XQ00",XQSET,0)=XQMESS_U_%Y_U_$P(^VA(200,DUZ,0),",","),^XTM
P("XQ00",0)=DT_U_DT+7
.Q
;
I '$D(XPDNM),$D(^XTMP("XQ00",XQSET,0)) D
.S DIR(0)="Y",DIR("B")="NO"
.S DIR("A")="Should I mark these options/protocols out-of-order now? (Y/N) "
.D ^DIR I Y D OFF^XQ001(XQSET)
.Q

```

```
;
K %, %Y, DIRUT, XQ, XQDEL, XQEND, XQFIL, XQFIL0, XQH, XQI, XQINI, XQK, XQM, XQMESS, XQN, XQON,
XQON0, XQRNG, XQSTART, XQSWTCH, XQT, XQX, X, Y
Q
```