*File created:* 8-Jan-92 10:57:28 **{piglet/n}<piglet>vanmelle>lispusers>NSMAINTAIN.;32**

*changes to:* (FNS \NSMT.DESCRIBE.OBJECT)
(VARS NSMAINTAINCOMS)

*previous date:* 17-Sep-91 14:31:41 {piglet/n}<piglet>vanmelle>lispusers>NSMAINTAIN.;30

*Read Table:* INTERLISP

*Package:* INTERLISP

*Format:* XCCS

;;


```
(RPAQQ NSMAINTAINCOMS
       [(COMS                                                    ; Main entry and utility fns
              (FNS NSMAINTAIN \NSMT.INITIAL.LOGIN \NSMT.HELP \NSMT.READFNAME \NSMT.LOOKUP \NSMT.LOOKUP1
                   \NSMT.CHECK.DOMAIN \NSMT.DOMAIN.MAY.EXIST \NSMT.FOREIGN.DOMAINP \NSMT.COLLECT.NAMES
                   \NSMT.GET.REMARK \NSMT.GET.PASSWORD \NSMT.LOGIN \NSMT.GETAUTHENTICATOR \NSMT.CHANGE.DOMAIN
                   \NSMT.PRINT.LIST \NSMT.PRINT.OBJECTS \NSMT.PROCESS.LIST \NSMT.READ.COMMA.LIST
                   \NSMT.SHOW.RESULT \NSMT.CHOOSE \NSMT.COURIER.OPEN \NSMT.CLEAR.CACHE EQUAL.NSADDRESS))
        (COMS                                                    ; Ordinary user commands
              (FNS \NSMT.CHANGE.PASSWORD \NSMT.DESCRIBE.ACL \NSMT.DESCRIBE.OBJECT \NSMT.DESCRIPTIVE.PROPS
                   \NSMT.DESCRIBE.PROPERTY \NSMT.PRETTY.PROPERTY \NSMT.LIST.OBJECTS \NSMT.LIST.CLEARINGHOUSES
                   \NSMT.LIST.SERVERS \NSMT.SHOW.DETAILS \NSMT.GROUP.FILTER \NSMT.LIST.ADMINISTRATORS
                   \NSMT.FETCH.ADMINISTRATORS \NSMT.FETCH.ADMINISTRATORS1 \NSMT.LIST.DOMAINS \NSMT.TYPE.ENTRY
                   \NSMT.TYPE.MEMBERS \NSMT.UNCACHE \NSMT.CLEAR.NAME.CACHE))
        (COMS                                                    ; Administrator commands
              (FNS \NSMT.ADD.ALIAS \NSMT.ADD.GROUP \NSMT.SET.INITIAL.ACL \NSMT.ADD.USER \NSMT.ADD.OBJECT
                   \NSMT.CREATE.OBJECT \NSMT.ADD.OBJECT.GENERIC \NSMT.CHANGE.ADDRESS \NSMT.CHANGE.ADMINISTRATORS
                   \NSMT.CHANGE.FORWARDING \NSMT.CHANGE.GROUP.COMPONENT \NSMT.CHANGE.REMARK
                   \NSMT.GET.OBJECT.TYPE \NSMT.REMOVE.ALIAS \NSMT.REMOVE.OBJECT \NSMT.REMOVE.USER))
        (FILES (SYSLOAD)
               DES AUTHENTICATION)
        (COMS                                                    ; Patch to clearinghouse
              (FNS CH.FINDSERVER))
        (VARS *NSMAINTAIN-COMMANDS* *NSMAINTAIN-ABORT-ITEM*)
        (ADDVARS (CH.PROPERTIES (ALIAS 1)
                        (BOOT.SERVICE 10026))
               (*NSMAINTAIN-DESCRIPTIVE-PROPERTIES* 10000 10001 10002 10003 10004 10005 10006 10007 10008 10009
                      10010 10011 10012 10013 10014 10015 10016 10017 10018 10019 10020 10021 10022 10023 10024
                      10026)
               (*NSMAINTAIN-IGNORE-PROPERTIES* 6 7 10027 20003 20002 20101)
               (*NSMAINTAIN-PROPERTY-FORMATS* (4 CLEARINGHOUSE . NETWORK.ADDRESS.LIST)
                      (8 RECORD (SIMPLE BOOLEAN)
                         (STRONG BOOLEAN))
                      (30 . NSNAME)
                      (31 CLEARINGHOUSE . MAILBOX.VALUES)
                      (10000 . STRING)
                      (10001 . STRING)
                      (10002 . STRING)
                      (10003 . STRING)
                      (10004 . STRING)
                      (10005 . STRING)
                      (10006 . STRING)
                      (10007 . STRING)
                      (10008 . STRING)
                      (10009 . STRING)
                      (10010 . STRING)
                      (10011 . STRING)
                      (10012 . STRING)
                      (10013 . STRING)
                      (10014 . STRING)
                      (10015 . STRING)
                      (10016 . STRING)
                      (10017 . STRING)
                      (10018 . STRING)
                      (10019 . STRING)
                      (10020 . STRING)
                      (10021 . STRING)
                      (10022 . STRING)
                      (10023 . STRING)
                      (10024 . STRING)
                      (10026 . STRING)
                      (10029 . STRING)
                      (10030 . STRING)
                      (10032 . STRING)
                      (10034 . STRING)
                      (10035 . STRING)
                      (15002 . STRING)
                      (20000 CLEARINGHOUSE . USERDATA.VALUE)
                      (20001 GAP . RS232CData)
                      (20006 SEQUENCE NSNAME)
                      (20007 . NSNAME)
```

```
                    (20102 GAP . RS232CBack)
                    (29965 . STRING)
                    (30005 . NSNAME))
              (*NSMAINTAIN-MEMBER-PROPERTIES* 3 20006))
       (INITVARS (*NSMAINTAIN-MEMBER-THRESHOLD* 3)
              (*NSMAINTAIN-SHOW-GROUP-ACCESS*))
       (DECLARE%: EVAL@COMPILE [P (CL:PROCLAIM '(CL:SPECIAL *NSMAINTAIN-MEMBER-THRESHOLD*
                                                  *NSMAINTAIN-SHOW-GROUP-ACCESS*))
                                  (CL:PROCLAIM '(GLOBAL *NSMAINTAIN-MEMBER-PROPERTIES*
                                                  *NSMAINTAIN-PROPERTY-FORMATS*
                                                  *NSMAINTAIN-IGNORE-PROPERTIES*
                                                  *NSMAINTAIN-DESCRIPTIVE-PROPERTIES*]
              DONTCOPY
              (FUNCTIONS WITH-CHS)
              (FILES (LOADCOMP)
                     CLEARINGHOUSE)
                                                                ; Get optimizer for CH.PROPERTY
              (CONSTANTS \CH.BROADCAST.SOCKET)
              (GLOBALVARS *NSMAINTAIN-COMMANDS* *NSMAINTAIN-ABORT-ITEM* CH.PROPERTIES)
              [P (CL:PROCLAIM '(CL:SPECIAL *USER* *LASTDOMAIN* *LASTNAME* *LASTGROUP* *LASTLIST* *LASTSTRING*
                                 *LAST-MEMBERSHIP* *SERVERTYPES* *ALLTYPES* *OBJECTTYPES* *DEFAULTDOMAIN*
                                 *REAL-NAME-CACHE* *DOMAIN*]
              (LOCALVARS . T)
              ;; For masterscope
              (VARS (*NSMT-MENU-FNS* (CL:REMOVE-DUPLICATES (FOR ENTRY IN *NSMAINTAIN-COMMANDS* WHEN
                                                              [LISTP (SETQ ENTRY (CADR (MEMB 'RETURN ENTRY]
                                                             COLLECT
                                                             (IF (EQ (CAR ENTRY)
                                                                     'FUNCTION)
                                                                 THEN
                                                                 (CADR ENTRY)
                                                                 ELSEIF
                                                                 (EQ (CAR ENTRY)
                                                                     'QUOTE)
                                                                 THEN
                                                                 (CAR (LISTP (CADR ENTRY])

;; Main entry and utility fns

(DEFINEQ

(NSMAINTAIN
  [LAMBDA NIL                                                   ; Edited 21-Nov-90 12:38 by bvm
    (PROG ([*STANDARD-OUTPUT* (PROGN                            ; Make sure T for FORMAT and PRINTOUT are the same
                                                                ; (yecch).
                                    (\GETSTREAM T 'OUTPUT]
          (*REAL-NAME-CACHE* (HASHARRAY 10 NIL [FUNCTION (LAMBDA (OBJECT)
                                                                ; Use first part of name to produce hash bits
                                                           (STRING-EQUAL-HASHBITS (fetch NSOBJECT of OBJECT]
                                    (FUNCTION EQUAL.CH.NAMES)))
          *USER* *LASTNAME* *DEFAULTDOMAIN* *LASTDOMAIN* *LASTGROUP* *LASTSTRING* *LASTLIST* *LAST-MEMBERSHIP*
          *SERVERTYPES* *ALLTYPES* *OBJECTTYPES* ORIG-USER PASS)
      ;; *REAL-NAME-CACHE* entries are of several possible forms:
      ;; 1) Ordinary ns name.  Value is distinguished name, or :NONE if no such object.
      ;; 2) org:*:*.  Value :OK => org is legal.  :NONE => no such org
      ;; 3) domain:org:*.  Same as 2, plus value :FOREIGN => domain:org is a known gatewayed domain.
      ;; 4) *:domain:org.  Value is list of domain administrators.
          (\NSMT.INITIAL.LOGIN)
          (SETQ ORIG-USER *USER*)
          [do (TERPRI T) repeatwhile (NULL (ERSETQ (bind CMD while [SETQ CMD (ASKUSER NIL NIL "CH: "
                                                                                *NSMAINTAIN-COMMANDS* T NIL
                                                                                '(AUTOCOMPLETEFLG T]
                                                                   do (COND
                                                                         ((LISTP CMD)
                                                                          (APPLY (CAR CMD)
                                                                                 (CDR CMD)))
                                                                         (T (CL:FUNCALL CMD)))
                                                                  (TERPRI T]
          (if (AND (NOT (EQUAL.CH.NAMES *USER* ORIG-USER))
                   (CL:Y-OR-N-P "Note: ~A is currently logged in.  Restore login to ~A? " *USER* ORIG-USER))
              then (SETPASSWORD '|NS::| (NSNAME.TO.STRING ORIG-USER T)
                                (PROMPTFORWORD "Password: " NIL NIL T '*])

(\NSMT.INITIAL.LOGIN
  [LAMBDA NIL                                                   ; Edited 14-Nov-90 17:12 by bvm
    ;; Get user to log in if necessary, and set *USER*, *LASTNAME*, *LASTDOMAIN*, *DEFAULTDOMAIN* appropriately
    (LET* [(CREDS (\INTERNAL/GETPASSWORD '|NS::|))
           [FULLNAME (\NSMT.LOOKUP1 (SETQ *USER* (PARSE.NSNAME (CAR CREDS]
           (BADP (CASE FULLNAME
                       (:NONE
                          (SETQ FULLNAME NIL)
```

```
                    "not a valid name")
                  ((NIL) "no verification from Clearinghouse")
                  (T NIL))]
        (CL:FORMAT T "[Default login:  ~A~@[ (~A)~];~%% Default domain: ~A]~%%" (NSNAME.TO.STRING (OR FULLNAME
                                                                                                    *USER*)
                                                                                                T)
              BADP
              (NSNAME.TO.STRING (SETQ *LASTDOMAIN* (SETQ *DEFAULTDOMAIN*
                                                    (create NSNAME
                                                            NSDOMAIN _ CH.DEFAULT.DOMAIN
                                                            NSORGANIZATION _ CH.DEFAULT.ORGANIZATION)))
                    T))
        (if (AND (SETQ *LASTNAME* FULLNAME)
                 (NOT (EQUAL.CH.NAMES *USER* FULLNAME)))
            then                                                ; Canonical name different from current login, so be helpful and
                                                                ; canonize
             (RPLACA CREDS (NSNAME.TO.STRING (SETQ *LASTNAME* (SETQ *USER* FULLNAME))
                                             T])
```

## (\NSMT.HELP

```
[LAMBDA NIL                                                     ; Edited 21-Aug-89 18:14 by bvm
```

;; Give more compact help than ASKUSER's default

```
(PRINTOUT T T T "  You need type only the initial letters of most command words.
         Use Control-E to abort a command." T T)
(LET ((LINELEN (LINELENGTH NIL T))
      *LASTSTRING* LASTN EXPLAINSTRING UNPRINTED CMD LEN TAB)
     [for ITEM in *NSMAINTAIN-COMMANDS* unless (EQ (CHCON1 (SETQ CMD (CAR ITEM)))
                                                   (CHARCODE ?))
       do                                                       ; Handle all commands but ?
          (if (AND [NOT (SETQ EXPLAINSTRING (LISTGET ITEM 'EXPLAINSTRING]
                   *LASTSTRING*
                   (> (SETQ LEN (NCHARS CMD))
                      LASTN)
                   (STRING-EQUAL *LASTSTRING* CMD :END1 LASTN :END2 LASTN))
              then                                              ; This command has same prefix as previous one
                   (if UNPRINTED
                       then (PRINTOUT T (SUBSTRING *LASTSTRING* 1 LASTN)
                                      "{"
                                      (SUBSTRING *LASTSTRING* (ADD1 LASTN)))
                            (SETQ UNPRINTED NIL)
                            (SETQ TAB (ADD1 (POSITION T)))      ; An aesthetically pleasing tab stop puts command directly under
                                                                ; next command
                           )
                   (PRIN1 "," T)
                   (if (> (+ (POSITION T)
                             (- LEN LASTN)
                             3)
                          LINELEN)
                       then                                     ; No room left on this line, so tab to reasonable place.
                            (TERPRI T)
                            (TAB TAB NIL T))
                   (PRIN1 (SUBSTRING CMD LASTN)
                          T)
              else                                              ; New prefix.
                   (if *LASTSTRING*
                       then                                     ; Clean up previous command
                            (PRINTOUT T (if UNPRINTED
                                            then *LASTSTRING*
                                            else "}")
                                      T))
                   (if EXPLAINSTRING
                       then                                     ; Explicit thing here for ?
                            (PRINTOUT T EXPLAINSTRING T)
                            (SETQ *LASTSTRING* NIL)
                     elseif (SETQ LASTN (STRPOS " " CMD))
                       then (SETQ *LASTSTRING* CMD)
                            (SETQ UNPRINTED T)
                       else (PRINTOUT T CMD T)
                            (SETQ *LASTSTRING* NIL)]
     (if *LASTSTRING*
         then                                                   ; Take care of the last line
              (PRINTOUT T (if UNPRINTED
                              then *LASTSTRING*
                              else "}")
                        T])
```

## (\NSMT.READFNAME

```
[LAMBDA (PROMPT DEFAULT DOMAINFLG ...FLG CHECK *OK)             ; Edited 14-Nov-90 17:09 by bvm
```

;; Prompt for a name with PROMPT, offering DEFAULT. If DOMAINFLG is true, we expect a domain (2-part name), else a 3-part name. If ...FLG ;; is true, print ... after successfully reading name.

;; CHECK controls whether we verify the name: NIL=don't; :OK=do, but happily accept anything; :CONFIRM=require confirmation if bad name; ;; :FOREIGN=accept names in foreign domains, otherwise like :CONFIRM; T=must be valid name.

;; *OK controls whether * is ok in any component: NIL=no; T=ok in first component only; :ANY=yes.

```
    (PROG ((COLON ":")
           NAME COLPOS FULLNAME REALNAME)
      RETRY
          (if [NULL (SETQ NAME (PROMPTFORWORD PROMPT (COND
                                                     ((AND DEFAULT (TYPENAMEP DEFAULT 'NSNAME))
                                                                                ; Make it fully qualified
                                                      (NSNAME.TO.STRING DEFAULT T))
                                                     (T DEFAULT))
                                     NIL T NIL NIL (CHARCODE (EOL]
              then (printout T " xxx" T)                                        ; aborted
                   (RETURN NIL))
          [SETQ FULLNAME (if (AND (SETQ COLPOS (STRPOS COLON NAME))
                                  (NEQ COLPOS (NCHARS NAME)))
                             then (SETQ COLPOS (STRPOS COLON NAME (ADD1 COLPOS)))
                                                                                ; Find second colon
                                  (if DOMAINFLG
                                      then                                      ; Wants domain name--a 2-part name
                                           (if COLPOS
                                               then                             ; too many colons
                                                    (PRINTOUT T " Invalid domain" T)
                                                    (RETURN NIL)
                                             else (PARSE.NSNAME NAME 2 *DEFAULTDOMAIN*))
                                    else (if (NOT COLPOS)
                                             then                               ; Org defaulted
                                                  (printout T COLON (fetch NSORGANIZATION of *DEFAULTDOMAIN*))
                                           elseif (EQ COLPOS (NCHARS NAME))
                                             then                               ; Trailing colon after domain
                                                  (printout T (fetch NSORGANIZATION of *DEFAULTDOMAIN*)))
                                         (PARSE.NSNAME NAME 3 *DEFAULTDOMAIN*))
                           else                                                 ; Completely unqualified (or only a trailing colon)
                                (if COLPOS
                                    then                                        ; User typed, e.g., "Fred:"
                                         (SETQ NAME (SUBSTRING NAME 1 -2))
                                  else (PRIN1 COLON T))
                                (if DOMAINFLG
                                    then (printout T (fetch NSORGANIZATION of *DEFAULTDOMAIN*))
                                         (create NSNAME using *DEFAULTDOMAIN* NSDOMAIN _ NAME)
                                  else (printout T (fetch NSDOMAIN of *DEFAULTDOMAIN*)
                                                   COLON
                                                   (fetch NSORGANIZATION of *DEFAULTDOMAIN*))
                                       (create NSNAME using *DEFAULTDOMAIN* NSOBJECT _ NAME]
          (if (STRPOS "*" NAME)
              then (if (CASE *OK
                             (:ANY                                              ; Any old * is ok
                                NIL)
                             ((NIL)                                             ; No * is ok
                                T)
                             (T                                                 ; * permitted in first part only
                                [OR (STRPOS "*" (fetch NSORGANIZATION of FULLNAME))
                                    (AND (NOT DOMAINFLG)
                                         (STRPOS "*" (fetch NSDOMAIN of FULLNAME]))
                       then (PRINTOUT T " ... Invalid use of *" T)
                            (SETQ DEFAULT FULLNAME)
                            (GO RETRY))
            elseif CHECK
              then                                                              ; Canonicalize the name
                   (SETQ REALNAME (\NSMT.LOOKUP FULLNAME (EQ CHECK :FOREIGN)))
                   (if (NULL REALNAME)
                       then (if (NOT (CASE CHECK
                                           (:OK                                 ; Accept it regardless
                                              T)
                                           ((:FOREIGN :CONFIRM)                 ; Accept with confirmation
                                              (CL:Y-OR-N-P " Use it anyway? "))
                                           (T                                   ; Must be valid name
                                              (TERPRI T)
                                              NIL)))
                                 then (SETQ DEFAULT FULLNAME)
                                      (GO RETRY))
                         else (SETQ FULLNAME REALNAME)))
          (COND
             (...FLG (PRIN1 " ... " T)))
          (RETURN FULLNAME])
```

## (\NSMT.LOOKUP

```
  [LAMBDA (NAME FOREIGNOK)                                                      ; Edited 14-Nov-90 17:20 by bvm
```

;; Like CH.LOOKUP.OBJECT but caches results (well, at least the positive ones). Also prints out message if it couldn't find name or name was an ;; alias

```
    (OR (TYPEP NAME 'NSNAME)
        (SETQ NAME (PARSE.NSNAME NAME)))
    (PROG ((CACHE (GETHASH NAME *REAL-NAME-CACHE*))
           FULLNAME)
          [if CACHE
              then (SETQ FULLNAME CACHE)
            else (CASE (\NSMT.CHECK.DOMAIN NAME)
```

```
                    ((:OK NIL) (if (SETQ FULLNAME (\NSMT.LOOKUP1 NAME))
                                   then (PUTHASH NAME FULLNAME *REAL-NAME-CACHE*)))
                    (:NONE (RETURN NIL))
                    (:FOREIGN (RETURN (AND FOREIGNOK NAME)))))]
          (if (NULL FULLNAME)
              then (PRINTOUT T " (couldn't verify name)")
            elseif (EQ FULLNAME :NONE)
              then (PRINTOUT T " (non-existent name)")
                   (SETQ FULLNAME NIL)
            elseif (NOT (EQUAL.CH.NAMES FULLNAME NAME))
              then (printout T " = " (NSNAME.TO.STRING FULLNAME T)))
          (RETURN FULLNAME])
```

## (\NSMT.LOOKUP1

```
[LAMBDA (NAME)                                                    ; Edited 14-Nov-90 16:34 by bvm
```

;;; Returns the canonical name of the specified object, :none if it doesn't exist, nil if we couldn't figure it out (because of chs problem)

```
    (LET ((ADDRESS (CH.FINDSERVER NAME T))
          RESULT)
         (if (NOT ADDRESS)
             then NIL
           elseif [NLISTP (SETQ RESULT (COURIER.EXPEDITED.CALL ADDRESS \CH.BROADCAST.SOCKET 'CLEARINGHOUSE
                                               'LOOKUP.OBJECT NAME (CH.GETAUTHENTICATOR)
                                               'RETURNERRORS]
             then RESULT
           elseif (EQ (CADDR (LISTP RESULT))
                      'NoSuchObject)
             then :NONE
           else NIL])
```

## (\NSMT.CHECK.DOMAIN

```
[LAMBDA (NAME)                                                    ; Edited 14-Nov-90 17:44 by bvm
```

;; See whether name is in a valid domain. Returns :ok, :none, :foreign, or nil if it can't figure out right now. We do all this because the Lisp chs
;; interface doesn't let us find out in any detail why an operation failed. Also, it's silly to keep getting prompt messages about a non-existent
;; domain, when we can certainly cache the answers.

```
    (PROG* ((ORG (fetch NSORGANIZATION of NAME))
            (TEST (create NSNAME
                          NSOBJECT _ ORG
                          NSDOMAIN _ "*"
                          NSORGANIZATION _ "*"))
            (RESULT (GETHASH TEST *REAL-NAME-CACHE*))
            ORGCACHE)
           (if (NOT RESULT)
               then ;; See if the org exists.  First check the chs cache, which is faster than asking a chs.
                    (CASE [OR (SETQ ORGCACHE (\CH.FIND.ORG.SERVER ORG T T))
                              (SETQ RESULT (\NSMT.LOOKUP1 (create NSNAME
                                                                  using TEST NSDOMAIN _ "CHServers" NSORGANIZATION _
                                                                    "CHServers"]
                        (:NONE )
                        ((NIL)                                    ; punt
                           (RETURN NIL))
                        (T (SETQ RESULT :OK)))                    ; And cache the result, copying the test object (which is ordinarily
                                                                  ; smashed further below)
                    (PUTHASH (create NSNAME using TEST)
                             RESULT *REAL-NAME-CACHE*))
           (if (NEQ RESULT :OK)
               then (printout T " (no such organization)")
                    (RETURN RESULT))
```

;; Ok, the org exists, shift right one

```
           (replace NSDOMAIN of TEST with (fetch NSOBJECT of TEST))
           (replace NSOBJECT of TEST with (fetch NSDOMAIN of NAME))
           (SETQ RESULT (GETHASH TEST *REAL-NAME-CACHE*))
           (if (NOT RESULT)
               then ;; See if the domain exists
                    (CASE [OR (AND (OR ORGCACHE (\CH.FIND.ORG.SERVER ORG T T))
                                   (CL:ASSOC (fetch NSDOMAIN of NAME)
                                             (CDDR ORGCACHE)
                                             :TEST
                                             'STRING-EQUAL))
                              (SETQ RESULT (\NSMT.LOOKUP1 (create NSNAME using TEST NSORGANIZATION _ "CHServers"]
                        (:NONE (if (NOT (SETQ RESULT (\NSMT.FOREIGN.DOMAINP NAME)))
                                   then                           ; punt
                                        (RETURN NIL)))
                        ((NIL)                                    ; punt
                           (RETURN NIL))
                        (T (SETQ RESULT :OK)))                    ; And cache the result
                    (PUTHASH TEST RESULT *REAL-NAME-CACHE*))
           (if (NEQ RESULT :OK)
               then (printout T " (" (if (EQ RESULT :FOREIGN)
                                         then "foreign"
```

```
                                    else "no such")
                            " domain)"))
          (RETURN RESULT])
```

## (\NSMT.DOMAIN.MAY.EXIST

```
[LAMBDA (DOMAIN)                                        ; Edited 14-Nov-90 18:03 by bvm
  (CASE (\NSMT.CHECK.DOMAIN DOMAIN)
      ((NIL :OK) T))])
```

## (\NSMT.FOREIGN.DOMAINP

```
[LAMBDA (NAME)                                          ; Edited 14-Nov-90 16:51 by bvm
  ;; Returns :foreign, :none, nil depending on whether name specifies a foreign domain, simply nonexistent domain, or we couldn't find out
  (LET* ((OBJ (create NSNAME
                      NSOBJECT _ (CONCAT (fetch NSDOMAIN of NAME)
                                         (fetch NSORGANIZATION of NAME))
                      NSDOMAIN _ "..."
                      NSORGANIZATION _ "..."))
         (RESULT (\NSMT.LOOKUP1 OBJ)))
        (CASE RESULT
            ((:NONE NIL) RESULT)
            (T ;; The object domainorganization:...:... exists.  Now retrieve the property that verifies that it's this domain and org, rather than
               ;; some other concatenation.
               (LET ((ADDRESS (CH.FINDSERVER RESULT T))
                     VALUE)
                    (if [AND ADDRESS (LISTP (SETQ VALUE (COURIER.EXPEDITED.CALL ADDRESS \CH.BROADCAST.SOCKET
                                                        'CLEARINGHOUSE
                                                        'RETRIEVE.ITEM RESULT (CH.PROPERTY
                                                                               '
                                                                               FOREIGNMAILSYSTEMNAME
                                                                               )
                                                        (CH.GETAUTHENTICATOR)
                                                        'RETURNERRORS]
                        then (if (NEQ (CAR VALUE)
                                      'ERROR)
                                 then (SETQ VALUE (COURIER.READ.REP (CADR VALUE)
                                                                    'CLEARINGHOUSE
                                                                    'NSNAME))
                                      (if (AND (STRING-EQUAL (fetch NSDOMAIN of NAME)
                                                             (fetch NSDOMAIN of VALUE))
                                               (STRING-EQUAL (fetch NSORGANIZATION of NAME)
                                                             (fetch NSORGANIZATION of VALUE)))
                                          then :FOREIGN
                                        else :NONE)
                               elseif (EQ (CADDR VALUE)
                                          'Missing)
                                 then :NONE)))))])
```

## (\NSMT.COLLECT.NAMES

```
[LAMBDA (PROMPT CHECK *OK)                              ; Edited 14-Aug-87 15:14 by bvm:
  ;; Prompt for an arbitrary number of names.  CHECK and *OK are the corresponding args to \nsmt.readfname.
  (bind NAME while (SETQ NAME (PROGN (TERPRI T)
                                     (\NSMT.READFNAME PROMPT NIL NIL NIL CHECK *OK)))
     collect NAME])
```

## (\NSMT.GET.REMARK

```
[LAMBDA (DEFAULT)                                       ; Edited 11-Aug-87 12:24 by bvm:
  ;; Prompt for a remark (an arbitrary string used to describe an object).  DEFAULT if any is usually the previous remark.
  (PROMPTFORWORD "Remark (terminate with CR):" DEFAULT NIL T NIL NIL (CHARCODE (CR])
```

## (\NSMT.GET.PASSWORD

```
[LAMBDA (PROMPT)                                        ; Edited 11-Aug-87 13:39 by bvm:
  ;; Read a password, prompting with PROMPT.  Ask user to retry password to verify that it was typed correctly.  Loop if the retype mismatches the
  ;; original.  Return NIL if user declines to enter a password in the first place.
  (PROG (PASS)
    LP  (COND
           ([NULL (SETQ PASS (PROMPTFORWORD PROMPT NIL NIL T '*]
            (RETURN NIL))
           ((STREQUAL PASS (PROMPTFORWORD " (retype password)" NIL NIL T '*))
            (RETURN PASS))
           (T (PRINTOUT T T "Mismatch.  Try again." T)
              (SETQ PROMPT "Password:")
              (GO LP])
```

## (\NSMT.LOGIN

```
[LAMBDA NIL                                             ; Edited 14-Nov-90 17:13 by bvm
  (bind LOGINFO FULLNAME until (OR (NULL (SETQ LOGINFO (\INTERNAL/GETPASSWORD '|NS::| T)))
```

```
                                    (COND
                                       [(AND [SETQ FULLNAME (\NSMT.LOOKUP1 (SETQ *USER* (PARSE.NSNAME
                                                                                         (CAR LOGINFO)
                                                                                         3 *DEFAULTDOMAIN*]
                                             (NEQ FULLNAME :NONE))
                                        (RPLACA LOGINFO (NSNAME.TO.STRING (SETQ *USER* FULLNAME)
                                                                          T))
                                                                          ; Make login canonical
                                        (\NSMT.SHOW.RESULT (NS.AUTHENTICATE (NS.MAKE.SIMPLE.CREDENTIALS LOGINFO]
                                       (T (CL:FORMAT T "  Invalid name ~A~%%" (NSNAME.TO.STRING *USER* T))
                                          NIL])
```

## (\NSMT.GETAUTHENTICATOR

```
[LAMBDA NIL                                                    ; Edited 14-Nov-90 11:57 by bvm
  (LET [(INFO (\INTERNAL/GETPASSWORD '|NS::|]
       (COND
          [INFO                                                ; Make sure we use the canonical user name here, not an alias
                (COURIER.CREATE (CLEARINGHOUSE . AUTHENTICATOR)
                       CREDENTIALS _ (COURIER.CREATE (AUTHENTICATION . CREDENTIALS)
                                          TYPE _ 'SIMPLE VALUE _ (COURIER.WRITE.REP *USER* 'AUTHENTICATION
                                                                                   'SIMPLE.CREDENTIALS))
                       VERIFIER _ (COURIER.WRITE.REP (HASH.PASSWORD (CDR INFO))
                                          'AUTHENTICATION
                                          'SIMPLE.VERIFIER]
          (T (ERROR!])
```

## (\NSMT.CHANGE.DOMAIN

```
[LAMBDA NIL                                                    ; Edited 18-Aug-89 17:12 by bvm
  (LET ((DOMAIN (\NSMT.READFNAME " (for name entry) to be:" *DEFAULTDOMAIN* T)))
       (COND
          (DOMAIN (TERPRI T)
                 [COND
                    ((CL:Y-OR-N-P "Set this default globally as well (i.e. for use outside Maintain)? ")
                     (SETQ CH.DEFAULT.DOMAIN (fetch NSDOMAIN of DOMAIN))
                     (SETQ CH.DEFAULT.ORGANIZATION (fetch NSORGANIZATION of DOMAIN]
                 (SETQ *LASTDOMAIN* (SETQ *DEFAULTDOMAIN* DOMAIN])
```

## (\NSMT.PRINT.LIST

```
[LAMBDA (LST PREFIX)                                           ; Edited 21-Nov-90 12:38 by bvm
  (if PREFIX
      then (PRINTOUT T .FONT BOLDFONT PREFIX .FONT DEFAULTFONT))
  (if (EQ (CAR LST)
          'ERROR)
      then (\NSMT.SHOW.RESULT LST)
    else (if (NULL LST)
             then (PRINTOUT T "(none)")
           else (MAPRINT LST T NIL NIL ", "))
         (TERPRI T])
```

## (\NSMT.PRINT.OBJECTS

```
[LAMBDA (OBJECTS)                                              ; Edited 15-Nov-90 18:04 by bvm
  (for OBJ in OBJECTS bind LASTDOMAIN LASTORG do (COND
                                                    ((AND LASTDOMAIN (STRING-EQUAL (fetch NSDOMAIN of OBJ)
                                                                                   LASTDOMAIN)
                                                          (STRING-EQUAL (fetch NSORGANIZATION of OBJ)
                                                                        LASTORG))
                                                     (PRINTOUT T ", "))
                                                    (T (PRINTOUT T T "[In " .FONT BOLDFONT
                                                               (SETQ LASTDOMAIN (fetch NSDOMAIN of OBJ))
                                                               ":"
                                                               (SETQ LASTORG (fetch NSORGANIZATION of OBJ))
                                                               .FONT DEFAULTFONT "] ")))
                                                  (PRIN1 (fetch NSOBJECT of OBJ)
                                                         T))
  (TERPRI T])
```

## (\NSMT.PROCESS.LIST

```
[LAMBDA (ITEMS *DOMAIN* LISTFN)                                ; Edited 26-Sep-90 17:26 by bvm
  (DECLARE (SPECVARS *DOMAIN*))                                ; Usable by LISTFN
```

;; Display a list of Clearinghouse objects. OBJECTS is the result of some sort of listing call. If the result is a list of strings, DOMAIN is supplied so
;; that future "Show Details" commands can use it. LISTFN is a function to call to print the list; it returns a possibly new list of objects to be saved
;; for later.

```
  (COND
     ((EQ (CAR ITEMS)
          'ERROR)
      (\NSMT.SHOW.RESULT ITEMS))
     (T (COND
           (LISTFN (SETQ ITEMS (CL:FUNCALL LISTFN ITEMS)))
           (T (\NSMT.PRINT.LIST ITEMS)))
        (COND
           (ITEMS                                              ; Save list for Show Details command.
```

```
                    (SETQ *LASTLIST* (CONS (AND *DOMAIN* (SETQ *LASTDOMAIN* (create NSNAME
                                                                                   using *DOMAIN* NSOBJECT _ NIL)))
                                           ITEMS])
```

## (\NSMT.READ.COMMA.LIST

```
[LAMBDA (PROMPT DEFAULT)                                    ; Edited 19-Nov-90 15:17 by bvm
```

;; Read a list of strings separated by commas. Return a list (or NIL) of the stuff between commas, with white space trimmed. DEFAULT is the old ;; list, offered as initial type in

```
(LET [(VALUE (TTYIN PROMPT NIL NIL '(STRING NORAISE)
                    NIL NIL (AND DEFAULT (if (CDR DEFAULT)
                                             then [CONCATLIST (CDR (for PIECE in DEFAULT
                                                                      join (LIST ", " (MKSTRING PIECE]
                                           else (MKSTRING (CAR DEFAULT]
      (AND VALUE (bind (START _ 1)
                       COMMA PIECE when (> [NCHARS (SETQ PIECE (CL:STRING-TRIM '(#\Space #\Tab)
                                                               (SUBSTRING VALUE START
                                                                      (AND (SETQ COMMA
                                                                                 (STRPOS "," VALUE START))
                                                                           (SUB1 COMMA]
                                           0)
                 collect                                    ; Parse stuff out from between the commas
                       PIECE
                 repeatwhile (AND COMMA (SETQ START (ADD1 COMMA])
```

## (\NSMT.SHOW.RESULT

```
[LAMBDA (RESULT PART FIRST SECOND)                          ; Edited 21-Aug-89 17:14 by bvm
```

;; Used to show the outcome of a typical clearinghouse operation. If RESULT is T or NIL, it succeeded, otherwise we print an error code. FIRST ;; and SECOND, if non-NIL, are the actual names we used in the call, in case error has a FIRST or SECOND identification.

```
(COND
   ((OR (EQ RESULT T)
        (NULL RESULT))
    (printout T " done" T)                                  ; Return T for success
    T)
   (T (COND
         (PART (PRINTOUT T " " PART)))
      (PRINTOUT T " failed: ")
      (if (EQ (CAR (LISTP RESULT))
              'ERROR)
          then (PRINTOUT T (CADDR RESULT))
               (LET [(CULPRIT (CASE (CADDDR RESULT)
                                    (FIRST FIRST)
                                    (SECOND SECOND))]
                    (if CULPRIT
                        then (PRINTOUT T " " CULPRIT)))
        else (PRINTOUT T RESULT))
      (TERPRI T)
      NIL])
```

## (\NSMT.CHOOSE

```
[LAMBDA (PROMPT ALTERNATIVES)                               ; Edited 19-Nov-90 14:50 by bvm
```

;; Prompt for one of alternatives. <cr> aborts.

```
(ASKUSER NIL NIL PROMPT (CONS *NSMAINTAIN-ABORT-ITEM* ALTERNATIVES)
       T])
```

## (\NSMT.COURIER.OPEN

```
[LAMBDA (DOMAIN)                                            ; Edited 14-Nov-90 19:11 by bvm
```

;; Open a courier connection to a server for this domain. Caller is responsible for closing it.

```
(PROG (SERVER STREAM LOOPED)
      (if (NOT (TYPENAMEP DOMAIN 'NSNAME))
          then (SETQ DOMAIN (PARSE.NSNAME DOMAIN 2)))
      (if (AND (CL:HASH-TABLE-P *REAL-NAME-CACHE*)
               (NOT (\NSMT.DOMAIN.MAY.EXIST DOMAIN)))
          then ;; Check up front whether domain is ok, rather than letting Lisp chs stuff go wild
               (RETURN NIL))
  TOP (if (SETQ SERVER (CH.FINDSERVER DOMAIN T))
          then (if (SETQ STREAM (COURIER.OPEN SERVER NIL T))
                   then                                     ; Ah, success
                        (RETURN STREAM))
               (if (NOT LOOPED)
                   then                                     ; Maybe time to refresh the cache
                        (\NSMT.CLEAR.CACHE DOMAIN)
                        (SETQ LOOPED T)
                        (GO TOP)))
      (PRINTOUT T "[Couldn't " (if SERVER
                                   then "contact"
                                  else "locate")
            " server for "
            (fetch NSDOMAIN of DOMAIN)
```

```
                ":"
                (fetch NSORGANIZATION of DOMAIN)
                "] ")
          (RETURN NIL])
```

## (\NSMT.CLEAR.CACHE

```
[LAMBDA (DOMAIN)                                                    ; Edited  2-Nov-90 14:51 by bvm
    ;; Clear the clearinghouse cache of servers for this domain.  NIL means everyone.  Domain can be *:org to clear all servers for a given org.
    ;; Returns T if it did anything.

    (if (NULL DOMAIN)
        then (SETQ \CH.CACHE (SETQ LOCAL.CLEARINGHOUSE NIL))
             (GETCLEARINGHOUSE)
             T
      else (SETQ DOMAIN (PARSE.NSNAME DOMAIN 2))
           (LET* ((ORG (fetch NSORGANIZATION of DOMAIN))
                  (ORGINFO (CL:ASSOC ORG \CH.CACHE :TEST 'STRING-EQUAL))
                  DOM DOMINFO)
                 (if (NULL ORGINFO)
                     then NIL
                  elseif (OR (STRING-EQUAL (SETQ DOM (fetch NSDOMAIN of DOMAIN))
                                           "*")
                             (STRING-EQUAL ORG "...")
                             (STRING-EQUAL ORG "CHServers"))
                    then (SETQ \CH.CACHE (DREMOVE ORGINFO \CH.CACHE))
                                                                    ; Get rid of them all
                         (if (AND LOCAL.CLEARINGHOUSE (EQUAL.NSADDRESS LOCAL.CLEARINGHOUSE (CAAADR ORGINFO)))
                             then                                   ; It was our primary server, so go get another.
                                  (SETQ LOCAL.CLEARINGHOUSE NIL)
                                  (GETCLEARINGHOUSE))
                         T
                  elseif (SETQ DOMINFO (CL:ASSOC DOM (CDDR ORGINFO)
                                                 :TEST
                                                 'STRING-EQUAL))
                    then (if [NULL (RPLACD (CDR ORGINFO)
                                           (DREMOVE DOMINFO (CDDR ORGINFO]
                             then                                   ; Get rid of org altogether if this was the only server cached
                                  (SETQ \CH.CACHE (DREMOVE ORGINFO \CH.CACHE)))
                         T])
```

## (EQUAL.NSADDRESS

```
[LAMBDA (A1 A2)                                                     ; Edited  2-Nov-90 14:50 by bvm
  (AND [EQ (ffetch NSHNM2 of (\DTEST A1 'NSADDRESS))
           (ffetch NSHNM2 of (\DTEST A2 'NSADDRESS]
       (EQ (ffetch NSHNM1 of A1)
           (ffetch NSHNM1 of A2))
       (EQ (ffetch NSHNM0 of A1)
           (ffetch NSHNM0 of A2))
       (EQ (ffetch NSNETLO of A1)
           (ffetch NSNETLO of A2))
       (EQ (ffetch NSNETHI of A1)
           (ffetch NSNETHI of A2))
       (EQ (ffetch NSSOCKET of A1)
           (ffetch NSSOCKET of A2])

)

;; Ordinary user commands

(DEFINEQ
```

## (\NSMT.CHANGE.PASSWORD

```
[LAMBDA NIL                                                         ; Edited 14-Nov-90 17:16 by bvm
  (LET ((NAME (\NSMT.READFNAME " for user:" (NSNAME.TO.STRING *USER* T)
                      NIL NIL T))
        PASS)
       (COND
          ((NULL NAME)
           NIL)
          ((NULL (SETQ PASS (\NSMT.GET.PASSWORD " to be:")))
           (printout T " xxx" T))
          (T (PRIN1 "..." T)
             (COND
                [[AND NAME (EQUAL.CH.NAMES *USER* (SETQ *LASTNAME* (SETQ *LASTSTRING* NAME]
                                                                    ; Changing own password
                 (COND
                    ([\NSMT.SHOW.RESULT (AS.CHANGE.OWN.PASSWORDS (\ENCRYPT.PWD (CONCAT PASS]
                     (\INTERNAL/SETPASSWORD '|NS::| (CONS (NSNAME.TO.STRING NAME T)
                                                          PASS]
                (T                                                  ; Changing someone else's password.  Only way to do this is to
                                                                    ; delete the old keys and create new ones.
                   (\NSMT.SHOW.RESULT (AS.REPLACE.PASSWORDS NAME (\ENCRYPT.PWD (CONCAT PASS])
```

## (\NSMT.DESCRIBE.ACL

```
[LAMBDA (NAME WHICH.LIST AUTH S PROPERTY)                    ; Edited 21-Nov-90 12:01 by bvm
  ;; Fetch and display the access control list WHICH.LIST for NAME.  PROPERTY is the property under control, defaulting to MEMBERS
  (LET ((MEMBERS (COURIER.CALL S 'CHACCESSCONTROL 'RETRIEVE.PROPERTY.ACL NAME (OR PROPERTY (CH.PROPERTY
                                                                                          'MEMBERS))
                         WHICH.LIST
                         '(CHACCESSCONTROL . ELEMENT.NAME)
                         (COURIER.FETCH (CLEARINGHOUSE . AUTHENTICATOR)
                                CREDENTIALS of AUTH)
                         (COURIER.FETCH (CLEARINGHOUSE . AUTHENTICATOR)
                                VERIFIER of AUTH)
                         'RETURNERRORS))
        ADMIN)
       (PRINTOUT T .FONT BOLDFONT (CASE WHICH.LIST
                                        (Administrators "Owners: ")
                                        (selfControllers "Friends: "))
              .FONT DEFAULTFONT)
       (if (AND (CDDDDR (LISTP MEMBERS))
                (SETQ ADMIN (\NSMT.FETCH.ADMINISTRATORS NAME T S))
                (EQ (LENGTH MEMBERS)
                    (LENGTH ADMIN))
                (CL:EVERY (FUNCTION EQUAL.CH.NAMES)
                       MEMBERS ADMIN))
           then ;; It's equal to the list of domain administrators, so guess that that's what it is.  It's really stupid that this interface doesn't let us
                ;; tell the difference between the acl being defaulted or not.
                (CL:FORMAT T "(Administrators of ~A:~A)~%%" (fetch NSDOMAIN of NAME)
                       (fetch NSORGANIZATION of NAME))
         else (\NSMT.PRINT.LIST MEMBERS])
```

## (\NSMT.DESCRIBE.OBJECT

```
[LAMBDA (NAME BRIEFLY)                                       ; Edited  8-Jan-92 10:57 by bvm
  ;; Identify name by type and show its interesting properties.  Return distinguished name if it exists, else NIL.
  (WITH-CHS
   (S NAME)
   (PROG* ((SIMPLE.AUTH (CH.GETAUTHENTICATOR))
           (NAME&PROPS (COURIER.CALL S 'CLEARINGHOUSE 'LIST.PROPERTIES NAME SIMPLE.AUTH 'RETURNERRORS))
           (PROP.MEMBERS (CH.PROPERTY 'MEMBERS))
           MAINPROPS PROPS ALIASES DESCR GOTSOME FORWARD GROUPP USERP USERGROUPP)
          (if (EQ (CAR NAME&PROPS)
                  'ERROR)
              then (RETURN (\NSMT.SHOW.RESULT NAME&PROPS))
            else                                             ; Pull out distinguished name
                 (SETQ NAME (CAR NAME&PROPS)))
          (FRESHLINE T)
          (printout T T .FONT BOLDFONT (NSNAME.TO.STRING NAME T)
                 .FONT DEFAULTFONT)
          (SETQ PROPS (CL:NSET-DIFFERENCE (CADR NAME&PROPS)
                             *NSMAINTAIN-IGNORE-PROPERTIES*))
          (SETQ MAINPROPS (\NSMT.DESCRIPTIVE.PROPS PROPS))
          (SETQ GROUPP (CL:MEMBER PROP.MEMBERS PROPS))
          (for P in MAINPROPS do (if (EQ P (CH.PROPERTY 'USER))
                                     then                    ; Note this for subsequent kludge
                                          (SETQ USERP T))
                                 [if (AND (EQ P (CH.PROPERTY 'USERGROUP))
                                          (PROGN (SETQ USERGROUPP T)
                                                 USERP)
                                          GROUPP)
                                     then
                                     ;; Both USER and group?  This is kludge to get NS mail forwarding, so don't mention USERGROUP
                                     ;; (the prop "describes" the forwarding, but is pretty uninteresting).  We depend on server returning
                                     ;; props in order, which means we got to USER before USERGROUP.
                                   else (CL:FORMAT T " ~A a ~A~@[ (~A)~]" (if GOTSOME
                                                                              then
                                                                              ; Multiple identities
                                                                                   (TERPRI T)
                                                                                   "   and"
                                                                          else
                                                                              ; First prop
                                                                                   (SETQ GOTSOME T)
                                                                                   "is")
                                                (\NSMT.PRETTY.PROPERTY P)
                                                (LET [(DESCR (COURIER.CALL S 'CLEARINGHOUSE 'RETRIEVE.ITEM NAME P
                                                                   SIMPLE.AUTH 'NOERROR]
                                                     ;; Description of object is stored as string on this descriptive property.  Sometimes the value is null,
                                                     ;; which is why we are careful about trying to interpret the result.
                                                     (AND DESCR (COURIER.READ.REP (CADR DESCR)
                                                                        NIL
                                                                        'STRING]
                                 (SETQ PROPS (CL:DELETE P PROPS)))
          [if GROUPP
              then (if USERP
                       then (SETQ FORWARD T)
```

```
                        (SETQ GROUPP NIL)
                  else (if (NOT USERGROUPP)
                           then (COND
                                    (GOTSOME (PRINTOUT T T " and"))
                                    (T (PRINTOUT T " is")))
                                (PRINTOUT T " a group"))
                       (SETQ PROPS (CL:DELETE PROP.MEMBERS PROPS]
          (SETQ *LASTSTRING* (if GROUPP
                                 then (SETQ *LASTGROUP* NAME)
                                 else (SETQ *LASTNAME* NAME)))
          [if (NOT BRIEFLY)
              then (TERPRI T)
                   (if (SETQ ALIASES (COURIER.CALL S 'CLEARINGHOUSE 'LIST.ALIASES.OF NAME '(CLEARINGHOUSE
                                                                                            . OBJECT.NAME)
                                                SIMPLE.AUTH
                                                'NOERROR))
                       then (\NSMT.PRINT.LIST ALIASES "Aliases: "))
                   (for P in PROPS do (\NSMT.DESCRIBE.PROPERTY NAME P S (AND FORWARD (EQ P PROP.MEMBERS)
                                                                             "Forwarding")))
                   (if (OR GROUPP (AND FORWARD (EQ *NSMAINTAIN-SHOW-GROUP-ACCESS* :ALWAYS)))
                       then                                  ; Show owners and friends
                            (\NSMT.DESCRIBE.ACL NAME 'Administrators SIMPLE.AUTH S)
                            (\NSMT.DESCRIBE.ACL NAME 'selfControllers SIMPLE.AUTH S)
                            (if (AND GROUPP (> *NSMAINTAIN-MEMBER-THRESHOLD* 0))
                                then                         ; Look at membership
                                     (PRINTOUT T .FONT BOLDFONT "Members: " .FONT DEFAULTFONT)
                                     (LET ([MEMBERS (LISTP (COURIER.CALL S 'CLEARINGHOUSE 'RETRIEVE.MEMBERS
                                                            NAME PROP.MEMBERS '(CLEARINGHOUSE
                                                                                . OBJECT.NAME)
                                                            SIMPLE.AUTH
                                                            'RETURNERRORS]
                                           (*PRINT-CASE* 10)
                                           N)
                                          (if (EQ (CAR MEMBERS)
                                                  'ERROR)
                                              then (\NSMT.SHOW.RESULT MEMBERS)
                                              else (if MEMBERS
                                                       then  ; Save for Type Members
                                                             (SETQ *LAST-MEMBERSHIP* (CONS NAME MEMBERS)))
                                                   (if (< (SETQ N (LENGTH MEMBERS))
                                                          *NSMAINTAIN-MEMBER-THRESHOLD*)
                                                       then  ; If there are few enough members, just show them
                                                             (\NSMT.PRINT.LIST MEMBERS)
                                                     else (PRINT N T]
          (RETURN NAME])
```

## (\NSMT.DESCRIPTIVE.PROPS

```
[LAMBDA (PROPS)                                              ; Edited 20-Nov-90 13:01 by bvm
```

;; PROPS is a list of property numbers. Return the subset that are "descriptive" properties, i.e., whose value is a remark string.

;; If we fail on the documented props, see if any props are in the 10000 range, which is conventionally allocated for descriptions

```
(OR (CL:INTERSECTION PROPS *NSMAINTAIN-DESCRIPTIVE-PROPERTIES*)
    (for P in PROPS collect P when (AND (>= P 10000)
                                        (<= P 20000])
```

## (\NSMT.DESCRIBE.PROPERTY

```
[LAMBDA (FNAME CHP S PROPNAME)                               ; Edited 20-Nov-90 14:47 by bvm
```

;; Called by \NSMT.TYPE.ENTRY to show one particular property.

```
(LET* [(GROUPP (MEMB CHP *NSMAINTAIN-MEMBER-PROPERTIES*))
       (VAL (COND
               (GROUPP                                       ; This is a group property, so get its value differently
                      (CH.RETRIEVE.MEMBERS FNAME CHP S))
               (T (COURIER.CALL S 'CLEARINGHOUSE 'RETRIEVE.ITEM FNAME CHP (CH.GETAUTHENTICATOR)
                                'RETURNERRORS]
      (PRINTOUT T .FONT BOLDFONT (OR PROPNAME (\NSMT.PRETTY.PROPERTY CHP T))
                ": " .FONT DEFAULTFONT)
      (if (EQ (CAR VAL)
              'ERROR)
          then (\NSMT.SHOW.RESULT VAL)
        elseif GROUPP
          then                                               ; Group property, print members as list
               (\NSMT.PRINT.LIST VAL)
        elseif (NULL (SETQ VAL (CADR VAL)))
          then                                               ; note that RETRIEVE.ITEM produced (name value)
               (PRINTOUT T "(null)" T)
        elseif (IGNORE-ERRORS (LET ((HOW (CDR (ASSOC CHP *NSMAINTAIN-PROPERTY-FORMATS*)))
                                    PGM)
                                   [while (AND (LISTP HOW)
                                               (LITATOM (CDR HOW))
                                               (CDR HOW))
                                     do                      ; Reduce to a less qualified name, to see if it gets down to a
                                                             ; record decl
                                        (SETQ HOW (\GET.COURIER.TYPE (SETQ PGM (CAR HOW))
                                                                     (CDR HOW]
```

```
                                        ;; Ok, now try to interpret the value
                                        (SETQ VAL (COURIER.READ.REP VAL PGM HOW))
                                        (if (EQ (CAR (LISTP HOW))
                                                'RECORD)
                                            then                    ; make records humanly intelligible
                                                 (for PAIR in (CDR HOW) as V in VAL bind (PREFIX _ "[")
                                                    do (PRIN1 PREFIX T)
                                                       (PRINTOUT T (CL:STRING-CAPITALIZE (STRING (CAR PAIR)))
                                                              ": "
                                                              (SELECTQ (CADR PAIR)
                                                                   (BOOLEAN (CL:IF V
                                                                                   "true"
                                                                                   "false"))
                                                                   (TIME (GDATE V))
                                                                   V))
                                                       (SETQ PREFIX "; ")
                                                    finally (PRINTOUT T "]" T))
                                                 T)))
          else                                                      ; just print what we've got
               (PRINTOUT T VAL T])
```

## (\NSMT.PRETTY.PROPERTY

```
[LAMBDA (P VERBOSE)                                                 ; Edited 20-Nov-90 14:27 by bvm
  (LET ((NAME (CH.NUMBER.TO.PROPERTY P)))
       (if NAME
           then (CL:STRING-CAPITALIZE NAME)
         else (CL:FORMAT NIL "#~D" P])
```

## (\NSMT.LIST.OBJECTS

```
[LAMBDA (PROP LISTFN)                                               ; Edited 14-Nov-90 18:04 by bvm
```

;;; given a clearinghouse property, lookup all objects with a user-specified pattern that have that property. Default pattern is * in recent domain.

```
    (LET (PATTERN)
         (COND
            ((AND (OR PROP (SETQ PROP (ASKUSER
                                       NIL NIL " having property "
                                       [OR *ALLTYPES*
                                           (SETQ *ALLTYPES*
                                            (CONS '("" "any" EXPLAINSTRING "<cr> - list ALL objects" RETURN
                                                  'ALL)
                                                  (CONS '        (* "" EXPLAINSTRING "* - list ALL objects" CONFIRMFLG T
                                                                    RETURN (QUOTE ALL))
                                                        (SORT (DREMOVE 'ALL (MAPCAR CH.PROPERTIES
                                                                                    (FUNCTION CAR]
                                       T)))
                  (SETQ PATTERN (\NSMT.READFNAME " by pattern:" (AND *LASTNAME*
                                                                     (create NSNAME using *LASTNAME* NSOBJECT _
                                                                             "*"))
                                       NIL T NIL T)))
             (AND (\NSMT.DOMAIN.MAY.EXIST PATTERN)
                  (\NSMT.PROCESS.LIST (CH.LIST.OBJECTS PATTERN PROP)
                         PATTERN LISTFN])
```

## (\NSMT.LIST.CLEARINGHOUSES

```
[LAMBDA NIL                                                         ; Edited 21-Aug-89 17:10 by bvm
  (DECLARE (USEDFREE *LASTDOMAIN*))
  (LET ((DOMAIN (\NSMT.READFNAME " serving domain:" *LASTDOMAIN* T))
        (CHSPART "CHServers")
        SERVERS)
       (COND
          (DOMAIN (SETQ *LASTDOMAIN* DOMAIN)
                  (TERPRI T)
                  [SETQ SERVERS (LISTP (CH.RETRIEVE.MEMBERS (create NSNAME
                                                                    NSOBJECT _ (fetch NSDOMAIN of DOMAIN)
                                                                    NSDOMAIN _ (fetch NSORGANIZATION of DOMAIN)
                                                                    NSORGANIZATION _ CHSPART]
                  (COND
                     [(EQ (CAR SERVERS)
                          'ERROR)
                      (\NSMT.SHOW.RESULT (COND
                                             ((EQ (CADDR SERVERS)
                                                  'NoSuchObject)
                                                                    ; translate this error
                                              "No Such Domain")
                                             (T SERVERS]
                     ([SETQ SERVERS (for S in SERVERS collect (COND
                                                                  ((AND (STRING-EQUAL (fetch NSDOMAIN of S)
                                                                                      CHSPART)
                                                                        (STRING-EQUAL (fetch NSORGANIZATION
                                                                                             of S)
                                                                                      CHSPART))
```

;; Clearinghouse names are usually of the form server:CHServers:CHServers. The domain here is

```
                                        ;; thus junk--print the name only.  Hope for not too much confusion if user tries to type name by hand,
                                        ;; rather than using Show Details command.
                                                                (fetch NSOBJECT of S))
                                                               (T
                                                                ; An aberrant name--punt by printing all full names
                                                                  (\NSMT.PROCESS.LIST SERVERS)
                                                                  (RETURN NIL]
                                                                ; Show short names, preserve domain for Show Details
                                (\NSMT.PROCESS.LIST SERVERS (create NSNAME
                                                                    NSDOMAIN _ CHSPART
                                                                    NSORGANIZATION _ CHSPART])
```

## (\NSMT.LIST.SERVERS

```
[LAMBDA NIL                                                             ; Edited 19-Nov-90 14:53 by bvm
```

;; List Objects specialized to servers.  We offer as choices those properties with SERVICE in their name, plus the oddly generic "SERVER".
;; CLEARINGHOUSE.SERVICE is excluded because its name space doesn't work as you'd expect.

```
  (LET [(PROP (\NSMT.CHOOSE " of type "
                    (OR *SERVERTYPES* (SETQ *SERVERTYPES*
                                            (SORT [CONS '("Server" "" RETURN 'SERVER)
                                                        (for P in CH.PROPERTIES
                                                           when (AND (STRPOS "SERVICE" (CAR P)
                                                                             -7)
                                                                     (NEQ (CAR P)
                                                                          'CLEARINGHOUSE.SERVICE))
                                                           collect `(,(CL:STRING-CAPITALIZE (SUBSTRING (CAR P)
                                                                                                       1 -9))
                                                                     "" RETURN ',(CAR P]
                                                  T]
        (AND PROP (\NSMT.LIST.OBJECTS PROP])
```

## (\NSMT.SHOW.DETAILS

```
[LAMBDA NIL                                                             ; Edited 20-Nov-90 17:19 by bvm
  (COND
     ((NULL *LASTLIST*)
      (PRINTOUT T " (no previous list)" T))
     (T (DESTRUCTURING-BIND (DOMAIN . OBJECTS)
               *LASTLIST*
               (COND
                  [(NULL (CDR OBJECTS))                                 ; only one, describe it straight away
                   (TERPRI T)
                   (\NSMT.DESCRIBE.OBJECT (COND
                                             (DOMAIN (create NSNAME using DOMAIN NSOBJECT _ (CAR OBJECTS)))
                                             (T (PARSE.NSNAME (CAR OBJECTS]
                  (T [COND
                        ((NOT (STRINGP (CAR OBJECTS)))                  ; Turn ns names into strings
                         (RPLACD *LASTLIST* (SETQ OBJECTS (for N in OBJECTS collect (NSNAME.TO.STRING N T]
                     (bind (CMDS _ (CONS *NSMAINTAIN-ABORT-ITEM* OBJECTS))
                           NAME while (SETQ NAME (PROGN (TERPRI T)
                                                        (ASKUSER NIL NIL "  name: " CMDS T)))
                        do (\NSMT.DESCRIBE.OBJECT (COND
                                                     (DOMAIN (create NSNAME using DOMAIN NSOBJECT _ NAME))
                                                     (T (PARSE.NSNAME NAME])
```

## (\NSMT.GROUP.FILTER

```
[LAMBDA (NAMES)                                                         ; Edited 26-Sep-90 17:47 by bvm
  (DECLARE (USEDFREE *DOMAIN*))
```

;; List function for List Objects -- NAMES is a list of objects that have a members prop.  Filter out those that also have a USER prop, assuming that
;; these "groups" are merely for forwarding, and print the rest.

;; We could ask for each object whether it's a user, but it's much faster to just ask the server to enumerate the users and take the difference.

```
  (LET [(USERS (CH.LIST.OBJECTS *DOMAIN* 'USER]
       (\NSMT.PRINT.LIST (CL:SET-DIFFERENCE NAMES USERS :TEST 'STRING-EQUAL])
```

## (\NSMT.LIST.ADMINISTRATORS

```
[LAMBDA NIL                                                             ; Edited 20-Nov-90 16:12 by bvm
  (LET ((DOMAIN (\NSMT.READFNAME " of domain:" *LASTDOMAIN* T T)))
       (if (AND DOMAIN (\NSMT.DOMAIN.MAY.EXIST DOMAIN))
           then (\NSMT.PROCESS.LIST (\NSMT.FETCH.ADMINISTRATORS (SETQ *LASTDOMAIN* DOMAIN])
```

## (\NSMT.FETCH.ADMINISTRATORS

```
[LAMBDA (DOMAIN CACHEOK S)                                              ; Edited 20-Nov-90 16:05 by bvm
```

;; Return the list of administrators for domain.  If CACHEOK is true, we're allowed to find the answer in the cache.  S is appropriate courier stream,
;; or NIL.

```
  (SETQ DOMAIN (create NSNAME using DOMAIN NSOBJECT _ "*"))             ; Copy just in case
  (OR (AND CACHEOK (GETHASH DOMAIN *REAL-NAME-CACHE*))
      (LET [(ADMIN (if S
                       then (\NSMT.FETCH.ADMINISTRATORS1 S DOMAIN)
                     else (WITH-CHS (S DOMAIN)
                                    (\NSMT.FETCH.ADMINISTRATORS1 S DOMAIN]
```

```
            (if (AND ADMIN (NEQ (CAR (LISTP ADMIN))
                                'ERROR))
                then (PUTHASH DOMAIN ADMIN *REAL-NAME-CACHE*)    ; Cache the results
                     ADMIN])
```

## (\NSMT.FETCH.ADMINISTRATORS1

```
[LAMBDA (S DOMAIN)                                               ; Edited 20-Nov-90 16:03 by bvm
  (COURIER.CALL S 'CHACCESSCONTROL 'RETRIEVE.DOMAIN.ACL DOMAIN 'Administrators '(CHACCESSCONTROL
                                                                                  . ELEMENT.NAME)
        '(SIMPLE NIL)
        '(0)
        'RETURNERRORS])
```

## (\NSMT.LIST.DOMAINS

```
[LAMBDA NIL                                                      ; Edited 18-Aug-89 17:12 by bvm
  (LET ((DOMAIN (\NSMT.READFNAME " by pattern:" (create NSNAME using *LASTDOMAIN* NSDOMAIN _ "*")
                       T T NIL T)))
       (COND
          (DOMAIN (\NSMT.PRINT.LIST (CH.LIST.DOMAINS DOMAIN])
```

## (\NSMT.TYPE.ENTRY

```
[LAMBDA NIL                                                      ; Edited 18-Aug-89 17:12 by bvm
  (LET (NAME)
       (COND
          ((SETQ NAME (\NSMT.READFNAME " name:" *LASTSTRING* NIL T NIL T))
           (\NSMT.DESCRIBE.OBJECT NAME])
```

## (\NSMT.TYPE.MEMBERS

```
[LAMBDA NIL                                                      ; Edited 21-Nov-90 12:53 by bvm
  (DECLARE (USEDFREE *LASTGROUP* *LASTSTRING*))
  (PROG ((NAME (\NSMT.READFNAME " of group:" *LASTGROUP* NIL T))
         ITEMS)
        (if (NOT NAME)
            then (RETURN))
        (SETQ *LASTSTRING* NAME)
        [if (AND *LAST-MEMBERSHIP* (EQUAL.CH.NAMES NAME (CAR *LAST-MEMBERSHIP*)))
            then (SETQ ITEMS (CDR *LAST-MEMBERSHIP*))
          elseif (NOT (\NSMT.DOMAIN.MAY.EXIST NAME))
            then (RETURN)
          elseif (EQ [CAR (SETQ ITEMS (LISTP (CH.RETRIEVE.MEMBERS NAME 'MEMBERS]
                     'ERROR)
            then                                                 ; Failure.  Translate the "Missing" error into English
                 (RETURN (\NSMT.SHOW.RESULT (if (EQ (CADDR ITEMS)
                                                    'Missing)
                                                then "Not A Group"
                                               else ITEMS]
        (SETQ *LASTGROUP* NAME)
        (if (NULL ITEMS)
            then (PRIN1 "(No members)" T)
          else (if (CDR ITEMS)
                   then (CL:FORMAT T "~2%%(~D members)~%%" (LENGTH ITEMS))
                        (\NSMT.PRINT.OBJECTS ITEMS)
                 else                                            ; Just one
                      (PRINTOUT T (CAR ITEMS)
                                T))                              ; Save list for Show Details command.
               (SETQ *LASTLIST* (CONS NIL ITEMS])
```

## (\NSMT.UNCACHE

```
[LAMBDA (ALLP)                                                   ; Edited 14-Nov-90 18:09 by bvm
  (LET (DOMAIN)
       (if (OR ALLP (SETQ DOMAIN (\NSMT.READFNAME ":" *LASTDOMAIN* T T)))
           then (if (NOT ALLP)
                    then (SETQ *LASTDOMAIN* DOMAIN))
                (PRINTOUT T (if (\NSMT.CLEAR.CACHE (AND (NOT ALLP)
                                                        DOMAIN))
                                then "done"
                               else "nothing cached")
                          T])
```

## (\NSMT.CLEAR.NAME.CACHE

```
[LAMBDA NIL                                                      ; Edited 21-Nov-90 13:06 by bvm
  (LET ((CNT (CL:HASH-TABLE-COUNT *REAL-NAME-CACHE*)))
       (TERPRI T)
       (if *LAST-MEMBERSHIP*
           then                                                  ; This is another cache
                (add CNT 1))
       (if (EQ CNT 0)
           then (PRINTOUT T "nothing cached" T)
         else (CLRHASH *REAL-NAME-CACHE*)
              (SETQ *LAST-MEMBERSHIP* NIL)
              (CL:FORMAT T "Ok, ~D cache entries cleared.~%%" CNT])
```

)

;; Administrator commands

(DEFINEQ

## (\NSMT.ADD.ALIAS

```
  [LAMBDA NIL                                                    ; Edited 14-Nov-90 12:13 by bvm
    (LET (OBJECT ALIAS)
         (COND
            ([AND (SETQ OBJECT (\NSMT.READFNAME " for object:" *LASTSTRING*))
                  (LET ((*DEFAULTDOMAIN* (create NSNAME using OBJECT NSOBJECT _ NIL)))
                       (DECLARE (CL:SPECIAL *DEFAULTDOMAIN*))    ; Read the alias by default in the same domain as object
                       (TERPRI T)
                       (SETQ ALIAS (\NSMT.READFNAME " Alias:" NIL NIL T]
             (OR [\NSMT.SHOW.RESULT (LISTP (SETQ *LASTSTRING* (WITH-CHS (S OBJECT)
                                                                        (COURIER.CALL S 'CLEARINGHOUSE
                                                                               'CREATE.ALIAS ALIAS OBJECT (
                                                                                      \NSMT.GETAUTHENTICATOR
                                                                                                    )
                                                                               'RETURNERRORS]
                 (SETQ *LASTSTRING* OBJECT])
```

## (\NSMT.ADD.GROUP

```
  [LAMBDA NIL                                                    ; Edited 15-Nov-90 18:01 by bvm
    ;; Create a new group
    (LET ((GROUP (\NSMT.READFNAME "  New group name:" NIL NIL T))
          AUTH REMARK RESULT MEMBERS OWNERS FRIENDS)
         (if (NULL GROUP)
           elseif [LISTP (SETQ RESULT (WITH-CHS (S GROUP)
                                        ;; Note: two calls on with-chs, because we want to create the object first, to assure it can be done, but
                                        ;; then user can take arbitrarily long supplying the group components
                                              (COURIER.CALL S 'CLEARINGHOUSE 'CREATE.OBJECT GROUP (SETQ AUTH (
                                                                                      \NSMT.GETAUTHENTICATOR
                                                                                                    ))
                                                     'RETURNERRORS]
             then                                                ; Failed to create object
                  (\NSMT.SHOW.RESULT RESULT)
           else (SETQ *LASTSTRING* (SETQ *LASTGROUP* GROUP))
                ;; Assume if user had access rights to create the object, then calls below don't fail.  Gather all the info before taking the time to call
                ;; the Clearinghouse, since sometimes these update calls are very slow.
                (TERPRI T)
                (SETQ REMARK (\NSMT.GET.REMARK))
                (CL:FORMAT T "~%%~%%Enter names of members, owners and friends, one per line, terminated with a
                       blank line.~%%")
                (SETQ MEMBERS (\NSMT.COLLECT.NAMES "Member:" :FOREIGN :ANY))
                (CL:FORMAT T "~%%(If you enter no owners, the group will be owned by the administrators of
                       ~A.)~%%" (create NSNAME using GROUP NSOBJECT _ NIL))
                (SETQ OWNERS (\NSMT.COLLECT.NAMES "Owner:" T :ANY))
                (SETQ FRIENDS (\NSMT.COLLECT.NAMES "Friend:" T :ANY))
                (TERPRI T)
                ;; Ok, we're ready to roll...
                (WITH-CHS (S GROUP)
                       (LET ((USERADMIN (create NSNAME using GROUP NSOBJECT _ "UserAdministration")))
                            (if (AND (NOT (CL:MEMBER USERADMIN OWNERS))
                                     (SETQ USERADMIN (COURIER.CALL S 'CLEARINGHOUSE 'LOOKUP.OBJECT USERADMIN AUTH
                                                                   'NOERROR))
                                     (CL:Y-OR-N-P "Do you want to include, as is conventional, ~A as an owner? "
                                            USERADMIN))
                                then (push OWNERS USERADMIN))
                            [LET* [(SELF *USER*)
                                   (FOUNDSELF (CL:MEMBER SELF OWNERS :TEST 'EQUAL.CH.NAMES]
                                  ;; Have to make user be first owner, because as soon as we add one administrator, we override the default
                                  ;; administrators, which means user is no longer empowered to add the rest of the owners!  Stupid
                                  ;; @#&#!!@ Clearinghouse design.
                                  (if FOUNDSELF
                                      then [if (NEQ FOUNDSELF OWNERS)
                                               then (SETQ OWNERS (CONS SELF (CL:REMOVE (CAR FOUNDSELF)
                                                                                       OWNERS]
                                    elseif (CL:Y-OR-N-P "Do you want to include yourself as an owner? ")
                                      then (SETQ OWNERS (CONS SELF OWNERS]
                                  (PRINTOUT T "Setting remark...")
                                  [\NSMT.SHOW.RESULT (LISTP (COURIER.CALL S 'CLEARINGHOUSE 'ADD.ITEM.PROPERTY GROUP
                                                                          (CH.PROPERTY 'USERGROUP)
                                                                          (COURIER.WRITE.REP REMARK 'CLEARINGHOUSE
                                                                                 'STRING)
                                                                          AUTH
                                                                          'RETURNERRORS]
                                  [if MEMBERS
                                      then (PRINTOUT T "Adding members...")
```

```
                                    (\NSMT.SHOW.RESULT (LISTP (COURIER.CALL S 'CLEARINGHOUSE
                                                                  'ADD.GROUP.PROPERTY GROUP
                                                                  (CH.PROPERTY 'MEMBERS)
                                                                  [FUNCTION (LAMBDA (DATASTREAM)
                                                          ; Function to write the membership onto the bulk data stream
                                                                      (COURIER.WRITE.BULKDATA
                                                                       DATASTREAM MEMBERS NIL
                                                                       'NSNAME]
                                                                  AUTH
                                                                  'RETURNERRORS]
                                (if OWNERS
                                    then (\NSMT.SET.INITIAL.ACL GROUP OWNERS 'Administrators AUTH S))
                                (if FRIENDS
                                    then (\NSMT.SET.INITIAL.ACL GROUP FRIENDS 'selfControllers AUTH S])
```

## (\NSMT.SET.INITIAL.ACL

```
[LAMBDA (GROUP MEMBERS WHICH.LIST AUTH S)                    ; Edited 31-Oct-90 16:59 by bvm
```

;; Set the initial access control list WHICH.LIST for GROUP to be MEMBERS

```
(PRINTOUT T "Adding " (CASE WHICH.LIST
                            (Administrators "owners")
                            (selfControllers "friends")
                            (T WHICH.LIST))
       "...")
(\NSMT.SHOW.RESULT (for NAME in MEMBERS thereis (SETQ $$VAL (LISTP (COURIER.CALL S 'CHACCESSCONTROL
                                                                   'ADD.MEMBER.TO.PROPERTY.ACL GROUP
                                                                   (CH.PROPERTY 'MEMBERS)
                                                                   WHICH.LIST NAME
                                                                   (COURIER.FETCH (CLEARINGHOUSE
                                                                                   . AUTHENTICATOR)
                                                                          CREDENTIALS of AUTH)
                                                                   (COURIER.FETCH (CLEARINGHOUSE
                                                                                   . AUTHENTICATOR)
                                                                          VERIFIER of AUTH)
                                                                   'RETURNERRORS])
```

## (\NSMT.ADD.USER

```
[LAMBDA NIL                                                  ; Edited 19-Nov-90 15:48 by bvm
```

;; Create new user

```
(PROG (AUTH NAME PASS ERROR)
      (DECLARE (USEDFREE *LASTNAME* *LASTSTRING*))
      (TERPRI T)
      (if (NOT (SETQ NAME (\NSMT.READFNAME "Name for new object:" *LASTNAME* NIL T)))
          then (RETURN))
      (SETQ *LASTSTRING* (SETQ *LASTNAME* NAME))
      (if [LISTP (SETQ ERROR (\NSMT.CREATE.OBJECT NAME (SETQ AUTH (\NSMT.GETAUTHENTICATOR]
          then                                               ; Error
               (RETURN (\NSMT.SHOW.RESULT ERROR)))
```

;; Having created the object, get all the other parts. We assume that if the creation succeeded, we'll be able to do the rest, so gather all the info
;; first, then do the calls.

```
      (TERPRI T)
      (if (NULL (SETQ PASS (\NSMT.GET.PASSWORD "Initial password:")))
          then (printout T " (no password stored; use Change Password to create one)" T))
      (\NSMT.ADD.OBJECT.GENERIC NAME AUTH (CH.PROPERTY 'USER))
```

;; Unfortunately, can't use the same Clearinghouse stream to do the passwords, since that requires an Authentication service. The two are
;; usually the same, but we can't assume so.

```
      (if PASS
          then (PRINTOUT T "Setting password...")
               (\NSMT.SHOW.RESULT (AS.CREATE.PASSWORDS NAME (\ENCRYPT.PWD PASS])
```

## (\NSMT.ADD.OBJECT

```
[LAMBDA NIL                                                  ; Edited 19-Nov-90 15:04 by bvm
```

;; Create new object of arbitrary type

```
(PROG (TYPE AUTH NAME PASS ERROR)
      (DECLARE (USEDFREE *LASTNAME* *LASTSTRING*))
      (if [NOT (AND (SETQ TYPE (\NSMT.GET.OBJECT.TYPE " of type: "))
                    (SETQ NAME (\NSMT.READFNAME "Name for new object:" *LASTNAME* NIL T]
          then (RETURN))
      (SETQ *LASTSTRING* (SETQ *LASTNAME* NAME))
      (if [LISTP (SETQ ERROR (\NSMT.CREATE.OBJECT NAME (SETQ AUTH (\NSMT.GETAUTHENTICATOR]
          then                                               ; Error
               (RETURN (\NSMT.SHOW.RESULT ERROR)))
```

;; Having created the object, get all the other parts. We assume that if the creation succeeded, we'll be able to do the rest, so gather all the info
;; first, then do the calls.

```
      (TERPRI T)
      (\NSMT.ADD.OBJECT.GENERIC NAME AUTH (CH.PROPERTY TYPE])
```

## (\NSMT.CREATE.OBJECT

```
  [LAMBDA (NAME AUTH)                                    ; Edited 19-Nov-90 14:17 by bvm
    ;; Create object and return its name or error
    (WITH-CHS (S NAME)
           (COURIER.CALL S 'CLEARINGHOUSE 'CREATE.OBJECT NAME AUTH 'RETURNERRORS])
```

## (\NSMT.ADD.OBJECT.GENERIC

```
  [LAMBDA (NAME AUTH TYPE)                               ; Edited 19-Nov-90 15:00 by bvm
    ;; Add the "generic" parts of a new object -- remark, aliases.
    (LET [(DESC (\NSMT.GET.REMARK))
          (ALIASES (LET ((*DEFAULTDOMAIN* (create NSNAME using NAME NSOBJECT _ NIL)))
                        (DECLARE (CL:SPECIAL *DEFAULTDOMAIN*))  ; Read the aliases by default in the same domain as object
                        (\NSMT.COLLECT.NAMES "Alias:"]
         (PRIN1 "... " T)
         (WITH-CHS (S NAME)
                (LET (ERROR)
                     (PRINTOUT T "Setting remark...")
                     [\NSMT.SHOW.RESULT (LISTP (COURIER.CALL S 'CLEARINGHOUSE 'ADD.ITEM.PROPERTY NAME TYPE
                                                    (AND DESC (COURIER.WRITE.REP DESC 'CLEARINGHOUSE
                                                                                      'STRING))
                                                    AUTH
                                                    'RETURNERRORS]
                     (if ALIASES
                         then (PRINTOUT T "Setting aliases...")
                              (\NSMT.SHOW.RESULT (AND [for A in ALIASES
                                                         thereis (SETQ ERROR (LISTP (COURIER.CALL S
                                                                                                  'CLEARINGHOUSE
                                                                                                  'CREATE.ALIAS A NAME
                                                                                                  AUTH 'RETURNERRORS]
                                                      ERROR])
```

## (\NSMT.CHANGE.ADDRESS

```
  [LAMBDA NIL                                            ; Edited 19-Nov-90 15:45 by bvm
    ;; Change the Address.list property of a machine.
    (PROG ([ADDRESS.PROPERTY (CONSTANT (CH.PROPERTY 'ADDRESS.LIST]
           PROPS NAME INPUT OLDADDRESSES NEWADDRESSES HADADDRESS)
          (DECLARE (USEDFREE *LASTSTRING* *LASTNAME*))
          (if (NOT (SETQ NAME (\NSMT.READFNAME " of machine:" *LASTNAME*)))
              then (RETURN))
          (TERPRI T)
          (SETQ PROPS (CH.LIST.PROPERTIES NAME))                ; returns (realname props)
          (if (EQ (SETQ NAME (CAR PROPS))
                  'ERROR)
              then                                              ; Object does not exist, probably
                   (RETURN (\NSMT.SHOW.RESULT PROPS)))
          (if (AND (SETQ HADADDRESS (MEMB ADDRESS.PROPERTY PROPS))
                   (SETQ OLDADDRESSES (CH.RETRIEVE.ITEM NAME ADDRESS.PROPERTY)))
              then (SETQ OLDADDRESSES (COURIER.READ.REP OLDADDRESSES 'CLEARINGHOUSE 'NETWORK.ADDRESS.LIST))
            else (PRINTOUT T NAME " does not yet have an address." T))
          (SETQ *LASTSTRING* (SETQ *LASTNAME* NAME))
      RETRY
          (PRINTOUT T "Type one or more NS addresses, separated by commas." T "Octal format: oo#o...o#oo or
                 Decimal: n-nnn#nnn-...-nnn#nnn" T)
          (if [NULL (SETQ INPUT (for X in (\NSMT.READ.COMMA.LIST "Address(es): " (OR INPUT OLDADDRESSES))
                                   collect (PARSE-NSADDRESS X 0]
              then                                              ; No new address...delete old?
                   [if (NOT HADADDRESS)
                       then (RETURN (PRINTOUT T " (not changed)" T))
                     elseif (CL:Y-OR-N-P "Remove address list for ~A? " NAME)
                       then (\NSMT.SHOW.RESULT (LISTP (CH.DELETE.PROPERTY ADDRESS.PROPERTY]
            elseif [MEMB NIL (SETQ NEWADDRESSES (for X in INPUT collect (PARSE-NSADDRESS X 0]
              then (PRINTOUT T "Illegal address:")
                   (for I in INPUT as A in NEWADDRESSES unless A bind (SEPR _ " ")
                      do (PRINTOUT T SEPR I)
                         (SETQ SEPR ","))
                   (TERPRI T)
            elseif (AND (EQ (LENGTH OLDADDRESSES)
                            (LENGTH NEWADDRESSES))
                        (for O in OLDADDRESSES as N in NEWADDRESSES always (EQUAL.NSADDRESS O N)))
              then (RETURN (PRINTOUT T " (not changed)" T))
            else (\NSMT.SHOW.RESULT (LISTP (CL:FUNCALL (if HADADDRESS
                                                           then (FUNCTION CH.CHANGE.ITEM)
                                                         else (FUNCTION CH.ADD.ITEM.PROPERTY))
                                                       NAME ADDRESS.PROPERTY NEWADDRESSES '(CLEARINGHOUSE
                                                                                            . NETWORK.ADDRESS.LIST]
)
```

## (\NSMT.CHANGE.ADMINISTRATORS

```
  [LAMBDA (CHACCESSFN OPERATION)                         ; Edited 20-Nov-90 16:15 by bvm
    ;; Add/remove a domain administrator
    (LET (DOMAIN INDIVIDUAL)
```

```
        (DECLARE (USEDFREE *LASTNAME* *LASTDOMAIN* *LASTSTRING*))
        (COND
           ((AND (SETQ INDIVIDUAL (\NSMT.READFNAME " name:" *LASTNAME*))
                 (SETQ DOMAIN (\NSMT.READFNAME (SELECTQ OPERATION
                                                   (ADD " to domain:")
                                                   (REMOVE " from domain:")
                                                   (SHOULDNT))
                                               *LASTDOMAIN* T T)))
            (REMHASH (create NSNAME using DOMAIN NSOBJECT _ "*")
                     *REAL-NAME-CACHE*)                    ; We're about to invalidate this cache entry
            (\NSMT.SHOW.RESULT (CL:FUNCALL CHACCESSFN DOMAIN 'Administrators INDIVIDUAL))
            (SETQ *LASTSTRING* (SETQ *LASTNAME* INDIVIDUAL))
            (SETQ *LASTDOMAIN* DOMAIN])
```

## (\NSMT.CHANGE.FORWARDING

```
  [LAMBDA NIL                                              ; Edited 20-Nov-90 13:00 by bvm
```

;; Change the "Forwarding" list for a user. Since NS doesn't really have forwarding, it is faked by giving an object a MEMBERS property--the mail
;; system, finding no mailbox, looks at the members and sends the message to all of them.

```
    (PROG (PROPS GOODPROPS NAME REALNAME OLDFORWARDING NEWFORWARDING HADFORWARDING HADUSERGROUP)
          (DECLARE (USEDFREE *LASTSTRING* *LASTNAME* *LASTGROUP*))
          (if (NOT (SETQ NAME (\NSMT.READFNAME " for user:" *LASTNAME*)))
              then (RETURN))
          (TERPRI T)
          (SETQ PROPS (CH.LIST.PROPERTIES NAME))           ; returns (realname props)
          (if (EQ (SETQ REALNAME (CAR PROPS))
                  'ERROR)
              then                                         ; Object does not exist, probably
                   (RETURN (\NSMT.SHOW.RESULT PROPS)))
          (if (MEMB (CH.PROPERTY 'USER)
                    (SETQ PROPS (CADR PROPS)))
              then                                         ; Ok, it's a user
            else (PRINTOUT T T REALNAME " is not a User")
                 (if (NULL (SETQ GOODPROPS (\NSMT.DESCRIPTIVE.PROPS PROPS)))
                     then (RETURN (PRINTOUT T ", or any other type I know about." T)))
                 (PRINTOUT T ", but a " (\NSMT.PRETTY.PROPERTY (CAR GOODPROPS)))
                 (if (CDR GOODPROPS)
                     then (PRINTOUT T " (also " [CONCATLIST (CDR (for P in (CDR GOODPROPS)
                                                                    join (LIST ", " (\NSMT.PRETTY.PROPERTY P]
                                ")"))
                 (if (EQ (CAR GOODPROPS)
                         (CH.PROPERTY 'USERGROUP))
                     then (RETURN (PRINTOUT T "  Groups %"forward%" to their members." T))
                   elseif (NOT (CL:Y-OR-N-P "Are you sure you want to change the Forwarding? "))
                     then (RETURN)))
          (if (SETQ HADFORWARDING (MEMB (CH.PROPERTY 'MEMBERS)
                                        PROPS))
              then                                         ; There's already forwarding, so fetch it
                   (SETQ OLDFORWARDING (CH.RETRIEVE.MEMBERS REALNAME))
            else (PRINTOUT T REALNAME " does not yet have Forwarding." T))
          (SETQ HADUSERGROUP (MEMB (CH.PROPERTY 'USERGROUP)
                                   PROPS))
          (SETQ *LASTSTRING* (SETQ *LASTNAME* REALNAME))
          (PRINTOUT T "Type one or more NS names, separated by commas." T)
          (if [NULL (SETQ NEWFORWARDING (MAPCAR (\NSMT.READ.COMMA.LIST "Forward to: "
                                                    (for NAME in OLDFORWARDING collect (NSNAME.TO.STRING NAME T))
                                                    )
                                                (FUNCTION PARSE.NSNAME]
              then                                         ; No new forwarding...delete old?
                   (if (NOT HADFORWARDING)
                       then (RETURN (PRINTOUT T " (not changed)" T))
                     elseif (NOT (CL:Y-OR-N-P "Remove forwarding for ~A? " REALNAME))
                       then (RETURN))
            elseif [AND (EQ (LENGTH OLDFORWARDING)
                            (LENGTH NEWFORWARDING))
                        (for O in OLDFORWARDING as N in NEWFORWARDING
                           always                          ; See if the lists are the same.  Could use EQUAL.CH.NAMES,
                                                           ; but want to be able to recognize case differences
                                (AND (CL:STRING= (fetch NSOBJECT of O)
                                                 (fetch NSOBJECT of N))
                                     (CL:STRING= (fetch NSDOMAIN of O)
                                                 (fetch NSDOMAIN of N))
                                     (CL:STRING= (fetch NSORGANIZATION of O)
                                                 (fetch NSORGANIZATION of N]
              then (RETURN (PRINTOUT T " (not changed)")))
          (WITH-CHS (S REALNAME)
                ;; Ok, ready to either delete old forwarding or change it.  Since there is no command to replace group membership, the easiest
                ;; thing when prop already existed is to delete the old one and add the new one
                (PROG ((AUTH (\NSMT.GETAUTHENTICATOR))
                       RESULT)
                      [if HADFORWARDING
                          then
                               ;; In either case, we want to delete the old members prop.
                               (SETQ RESULT (COURIER.CALL S 'CLEARINGHOUSE 'DELETE.PROPERTY REALNAME
                                                          (CH.PROPERTY 'MEMBERS)
```

```
                                        AUTH
                                        'RETURNERRORS))
                              (if (EQ (CAR RESULT)
                                      'ERROR)
                                  then (RETURN (\NSMT.SHOW.RESULT RESULT]
                         (if (NOT NEWFORWARDING)
                             then (PRINTOUT T "Forwarding removed")
                                  (if (AND HADUSERGROUP (EQ [CAR (SETQ RESULT (COURIER.CALL S 'CLEARINGHOUSE
                                                                                    'DELETE.PROPERTY REALNAME
                                                                                    (CH.PROPERTY 'USERGROUP)
                                                                                    AUTH
                                                                                    'RETURNERRORS]
                                                                      'ERROR))
                                      then                                    ; Failed to delete the "group" comment
                                           (PRINTOUT T ", but failed to remove the forwarding comment because: "
                                                  (CADDR RESULT)
                                                  T)
                                    else (PRINTOUT T "." T))
                           else ;; Create new membership
                                (SETQ RESULT (COURIER.CALL S 'CLEARINGHOUSE 'ADD.GROUP.PROPERTY REALNAME
                                                    (CH.PROPERTY 'MEMBERS)
                                                    [FUNCTION (LAMBDA (DATASTREAM)
                                                                                    ; Function to write the membership onto the bulk data stream
                                                                (COURIER.WRITE.BULKDATA DATASTREAM NEWFORWARDING
                                                                      NIL 'NSNAME]
                                                    AUTH
                                                    'RETURNERRORS))
                                (if (EQ (CAR RESULT)
                                        'ERROR)
                                    then (\NSMT.SHOW.RESULT RESULT)
                                  else (if (NOT HADUSERGROUP)
                                           then (SETQ RESULT (COURIER.CALL S 'CLEARINGHOUSE 'ADD.ITEM.PROPERTY
                                                                   REALNAME (CH.PROPERTY 'USERGROUP)
                                                                   (COURIER.WRITE.REP (CONCAT "Forwarding for "
                                                                                              (fetch NSOBJECT
                                                                                                 of REALNAME))
                                                                         'CLEARINGHOUSE
                                                                         'STRING)
                                                                   AUTH
                                                                   'RETURNERRORS))
                                                                   ; This isn't strictly necessary, but I think some tools expect it to
                                                                   ; be there
                                                (if (EQ (CAR RESULT)
                                                        'ERROR)
                                                    then (PRINTOUT T "(Failed to set usergroup comment)" T)))
                                (PRINTOUT T "Done, forwarding set to ")
                                (\NSMT.PRINT.LIST NEWFORWARDING)
                                (TERPRI T])
```

## (\NSMT.CHANGE.GROUP.COMPONENT

```
[LAMBDA (CHFN OPERATION SELF/LIST)                    ; Edited 21-Nov-90 13:06 by bvm
```

;; Add or remove a member from to/from a group. CHACCESSFN is the CH function that will make the change, OPERATION is ADD or REMOVE, ;; and SELF/LIST is one of T (self), NIL (general member) or the name of an access list property.

```
(LET (GROUP INDIVIDUAL ORIGINAL)
     (if [AND (OR (EQ SELF/LIST T)
                  (SETQ INDIVIDUAL (\NSMT.READFNAME " name:" *LASTNAME* NIL NIL (COND
                                                                                    ((EQ OPERATION
                                                                                         'REMOVE)
                                                                                    ; Want to be able to remove bogus names if they got on there
                                                                                    ; somehow, so let's do the processing ourselves
                                                                                     NIL)
                                                                                    (SELF/LIST
                                                                                    ; must be valid ns name
                                                                                               T)
                                                                                    (T
                                                                                    ; use canonical name, but foreign names ok
                                                                                       :FOREIGN))
                                                                   :ANY)))
              (PROGN (if [AND (EQ OPERATION 'REMOVE)
                              (NEQ SELF/LIST T)
                              (NOT (STRPOS "*" (NSNAME.TO.STRING INDIVIDUAL]
                         then                                     ; Do name fixing ourselves so we can keep track of the original
                                                                  ; (below)
                              (SETQ INDIVIDUAL (OR (\NSMT.LOOKUP (SETQ ORIGINAL INDIVIDUAL))
                                                   INDIVIDUAL)))
                     (SETQ GROUP (\NSMT.READFNAME (SELECTQ OPERATION
                                                      (ADD " to group:")
                                                      (REMOVE " from group:")
                                                      (SHOULDNT))
                                                  *LASTGROUP* NIL T]
       then ;; Ok, here's a name and a group, try the desired operation
            (CASE SELF/LIST
                ((T NIL)                                      ; We're about to spoil the cache
```

```
                (SETQ *LAST-MEMBERSHIP* NIL)))
          [WITH-CHS (S GROUP)
                (PROG ((AUTH (\NSMT.GETAUTHENTICATOR))
                       (MEMBER INDIVIDUAL)
                       RESULT)
                  RETRY
                      [SETQ RESULT (CASE SELF/LIST
                                         ((T)              ; adding/removing self
                                            (COURIER.CALL S 'CLEARINGHOUSE CHFN GROUP (CH.PROPERTY
                                                                                       'MEMBERS)
                                                   AUTH
                                                   'RETURNERRORS))
                                         ((NIL)            ; adding/removing member
                                            (COURIER.CALL S 'CLEARINGHOUSE CHFN GROUP (CH.PROPERTY
                                                                                       'MEMBERS)
                                                   MEMBER AUTH 'RETURNERRORS))
                                         (T                ; Adding/removing from access list
                                            (COURIER.CALL S 'CHACCESSCONTROL CHFN GROUP
                                                   (CH.PROPERTY 'MEMBERS)
                                                   SELF/LIST MEMBER (COURIER.FETCH (CLEARINGHOUSE
                                                                                   . AUTHENTICATOR)
                                                                                   CREDENTIALS of AUTH)
                                                   (COURIER.FETCH (CLEARINGHOUSE . AUTHENTICATOR)
                                                          VERIFIER of AUTH)
                                                   'RETURNERRORS)))]
                      (if (AND (LISTP RESULT)
                               (EQ (CADDR RESULT)
                                   'NoChange)
                               ORIGINAL
                               (EQ MEMBER INDIVIDUAL)
                               (NOT (EQUAL.CH.NAMES INDIVIDUAL ORIGINAL)))
                          then
                              ;; Command was to remove something.  We first tried the full name, but CH said nothing happened.
                              ;; So try original name, just in case someone got an alias on the list by mistake.
                              (SETQ MEMBER ORIGINAL)
                              (GO RETRY))
                      (if (\NSMT.SHOW.RESULT (LISTP RESULT)
                                 NIL GROUP MEMBER)
                          then                             ; Success
                              (if (NEQ MEMBER INDIVIDUAL)
                                  then (PRINTOUT T "(removed " (NSNAME.TO.STRING ORIGINAL T)
                                               ")" T))
                              (SETQ *LASTSTRING* (SETQ *LASTGROUP* RESULT]
          (if INDIVIDUAL
              then (SETQ *LASTNAME* INDIVIDUAL])
```

## (\NSMT.CHANGE.REMARK

```
[LAMBDA NIL                                                  ; Edited 20-Nov-90 12:58 by bvm
  (PROG (PROPS GOODPROPS MAINPROP NAME REALNAME RESULT REMARK OLDREMARK)
        (DECLARE (USEDFREE *LASTSTRING* *LASTNAME* *LASTGROUP*))
        (if (NOT (SETQ NAME (\NSMT.READFNAME " for object:" *LASTSTRING*)))
            then (RETURN))
        (SETQ PROPS (CH.LIST.PROPERTIES NAME))               ; returns (realname props)
        (if (EQ (SETQ REALNAME (CAR PROPS))
                'ERROR)
            then                                             ; Object does not exist, probably
                 (RETURN (\NSMT.SHOW.RESULT PROPS)))
        [if [NULL (SETQ GOODPROPS (\NSMT.DESCRIPTIVE.PROPS (CADR PROPS]
            then (printout T T (SETQ *LASTSTRING* REALNAME)
                         " has no remarkable properties." T)
                 (if (NULL (SETQ MAINPROP (\NSMT.GET.OBJECT.TYPE "Add remark of type (<cr> to abort): ")))
                     then (RETURN))
          else (if [OR (NULL (CDR GOODPROPS))
                       (AND (EQ (CAR GOODPROPS)
                                (CH.PROPERTY 'USER))
                            (EQ (CADR GOODPROPS)
                                (CH.PROPERTY 'USERGROUP))
                            (NULL (CDDR GOODPROPS]
                   then                                      ; only one, the normal case (or both user & usergroup, in which
                                                             ; case we ignore the boring forwarding remark)
                        [CL:FORMAT T " (~@[~A -- ~]a ~A)" (AND (NOT (EQUAL.CH.NAMES REALNAME NAME))
                                                               (NSNAME.TO.STRING REALNAME))
                                (\NSMT.PRETTY.PROPERTY (SETQ MAINPROP (CAR GOODPROPS]
                 else (PRINTOUT T T (NSNAME.TO.STRING REALNAME)
                                " has the descriptive properties ")
                      [\NSMT.PRINT.LIST (SETQ GOODPROPS (for P in GOODPROPS collect (OR (CH.NUMBER.TO.PROPERTY P)
                                                                                        P]
                      (if (NULL (SETQ MAINPROP (\NSMT.CHOOSE "Specify property to modify: " GOODPROPS)))
                          then (RETURN)))
              (TERPRI T)
              (if (SETQ OLDREMARK (CADR (CH.RETRIEVE.ITEM REALNAME MAINPROP)))
                  then                                       ; Retrieve carefully in case the prop is null
                       (SETQ OLDREMARK (COURIER.READ.REP OLDREMARK NIL 'STRING]
        (if (NOT (FIXP MAINPROP))
            then                                             ; Convert prop we got from interaction back to number
```

```
                (SETQ MAINPROP (CH.PROPERTY MAINPROP)))
          (if (SETQ REMARK (\NSMT.GET.REMARK OLDREMARK))
              then (PRIN1 "..." T)
                   [\NSMT.SHOW.RESULT (LISTP (if GOODPROPS
                                                 then (CH.CHANGE.ITEM REALNAME MAINPROP REMARK 'STRING)
                                                 else (CH.ADD.ITEM.PROPERTY REALNAME MAINPROP REMARK 'STRING]
            else (PRINTOUT T " xxx" T))
          (SETQ *LASTSTRING* (if (EQ MAINPROP (CH.PROPERTY 'USERGROUP))
                                 then (SETQ *LASTGROUP* REALNAME)
                               else (SETQ *LASTNAME* REALNAME])
```

## (\NSMT.GET.OBJECT.TYPE

```
  [LAMBDA (PROMPT)                                                    ; Edited 19-Nov-90 14:50 by bvm
    (\NSMT.CHOOSE PROMPT (OR *OBJECTTYPES* (SETQ *OBJECTTYPES* (SORT (for P in
                                                                        *NSMAINTAIN-DESCRIPTIVE-PROPERTIES*
                                                                     collect (OR (CH.NUMBER.TO.PROPERTY P)
                                                                                 P])
```

## (\NSMT.REMOVE.ALIAS

```
  [LAMBDA NIL                                                         ; Edited 18-Aug-89 17:12 by bvm
    (LET (ALIAS)
         (COND
            [(NULL (SETQ ALIAS (\NSMT.READFNAME " alias:" NIL NIL T]
            ((NLISTP (SETQ ALIAS (CH.DELETE.ALIAS ALIAS)))       ; Success, returned canonical name
             (CL:FORMAT T "done, alias was removed from ~S~%%" (SETQ *LASTSTRING* ALIAS)))
            (T (\NSMT.SHOW.RESULT ALIAS])
```

## (\NSMT.REMOVE.OBJECT

```
  [LAMBDA (NAME)                                                      ; Edited 18-Aug-89 17:12 by bvm
    (COND
       ((AND (OR NAME (SETQ NAME (\NSMT.READFNAME ":" *LASTSTRING* NIL T)))
             (SETQ NAME (\NSMT.DESCRIBE.OBJECT NAME T))
             (CL:Y-OR-N-P " Confirm deletion (y or n): "))
        (\NSMT.SHOW.RESULT (LISTP (CH.DELETE.OBJECT NAME])
```

## (\NSMT.REMOVE.USER

```
  [LAMBDA NIL                                                         ; Edited 18-Aug-89 17:12 by bvm
    (LET (USER INFO)
         (COND
            [(NULL (SETQ USER (\NSMT.READFNAME ":" *LASTNAME* NIL T]
            ([NULL (SETQ INFO (CH.RETRIEVE.ITEM USER 'USER]
             (PRINTOUT T " not a user." T))
            (T (PRINTOUT T T (NSNAME.TO.STRING (CAR INFO)
                                               T))
               [COND
                  ((CADR INFO)
                   (CL:FORMAT T " (~A)" (COURIER.READ.REP (CADR INFO)
                                                          NIL
                                                          'STRING]
               (COND
                  ((CL:Y-OR-N-P " Confirm deletion (y or n): ")
                   (\NSMT.SHOW.RESULT (LISTP (CH.DELETE.OBJECT USER])

)

(FILESLOAD (SYSLOAD)
       DES AUTHENTICATION)
```

;; Patch to clearinghouse

```
(DEFINEQ
```

## (CH.FINDSERVER

```
  [LAMBDA (DOMAINPATTERN NOERRORFLG DONTPROBEFLG)                     ; Edited 20-Feb-91 16:16 by bvm
    ;; Find a Clearinghouse which serves the specified domain and return its NS address.  If DONTPROBEFLG is T, just search the cache.
    (OR (type? NSNAME DOMAINPATTERN)
        (SETQ DOMAINPATTERN (PARSE.NSNAME DOMAINPATTERN 2)))
    (LET ((ORGANIZATION (fetch NSORGANIZATION of DOMAINPATTERN))
          (DOMAIN (fetch NSDOMAIN of DOMAINPATTERN))
          (GLUE "CHServers")
          ORGANIZATION.INFO)
         (if (STRING-EQUAL ORGANIZATION GLUE)
             then                                                     ; Shift right
                  [if (STRING-EQUAL DOMAIN GLUE)
                      then                                            ; Everyone handles this
                           (GETCLEARINGHOUSE)
                    else (CAR (CAR (fetch OCALLSERVERS of (\CH.FIND.ORG.SERVER DOMAIN NOERRORFLG DONTPROBEFLG]
           else (SETQ ORGANIZATION.INFO (\CH.FIND.ORG.SERVER ORGANIZATION NOERRORFLG DONTPROBEFLG))
                (if (STRING-EQUAL DOMAIN "*")
                    then                                              ; Any server in the org will do.
                         (CAR (CAR (fetch OCALLSERVERS of ORGANIZATION.INFO)))
                  elseif [for DOMAIN.INFO in (fetch OCDOMAINS of ORGANIZATION.INFO)
```

```
                        when (STRING-EQUAL (fetch DCDOMAIN of DOMAIN.INFO)
                                           DOMAIN)
                        do (RETURN (CAR (CAR (fetch DCKNOWNSERVERS of DOMAIN.INFO]
                  elseif DONTPROBEFLG
                    then (AND (NOT NOERRORFLG)
                              (ERROR "Couldn't find Clearinghouse server for domain" DOMAINPATTERN T))
                  else ;; Ask a clearinghouse in ORGANIZATION to find servers for this domain.  For simplicity, assume the first one will tell us.
                       ;; This should be 'Local Clearinghouse' if it serves ORGANIZATION
                       (\CH.LOCATE.SERVERS (CAR (CAR (fetch OCALLSERVERS of ORGANIZATION.INFO)))
                              (create NSNAME
                                      NSOBJECT _ DOMAIN
                                      NSDOMAIN _ ORGANIZATION
                                      NSORGANIZATION _ GLUE)
                             NOERRORFLG ORGANIZATION DOMAIN)
                       (CH.FINDSERVER DOMAINPATTERN NOERRORFLG T])

)

(RPAQQ *NSMAINTAIN-COMMANDS*
       (("?" "" RETURN (FUNCTION \NSMT.HELP))
        ("Add Alias" "" RETURN (FUNCTION \NSMT.ADD.ALIAS))
        ("Add Domain Administrator" "" RETURN '(\NSMT.CHANGE.ADMINISTRATORS CH.ADD.MEMBER.TO.DOMAIN.ACL ADD))
        ("Add Friend" "" RETURN '(\NSMT.CHANGE.GROUP.COMPONENT ADD.MEMBER.TO.PROPERTY.ACL ADD selfControllers))
        ("Add Group" "" RETURN (FUNCTION \NSMT.ADD.GROUP))
        ("Add Member" "" RETURN '(\NSMT.CHANGE.GROUP.COMPONENT ADD.MEMBER ADD))
        ("Add Owner" "" RETURN '(\NSMT.CHANGE.GROUP.COMPONENT ADD.MEMBER.TO.PROPERTY.ACL ADD Administrators))
        ("Add Registered Object" "" RETURN (FUNCTION \NSMT.ADD.OBJECT))
        ("Add Self" "" RETURN '(\NSMT.CHANGE.GROUP.COMPONENT ADD.SELF ADD T))
        ("Add User" "" RETURN (FUNCTION \NSMT.ADD.USER))
        ("Remove Alias" "" RETURN (FUNCTION \NSMT.REMOVE.ALIAS))
        ("Remove Domain Administrator" "" RETURN '(\NSMT.CHANGE.ADMINISTRATORS CH.DELETE.MEMBER.FROM.DOMAIN.ACL
                                                                              REMOVE))
        ("Remove Friend" "" RETURN '(\NSMT.CHANGE.GROUP.COMPONENT DELETE.MEMBER.FROM.PROPERTY.ACL REMOVE
                                                selfControllers))
        ("Remove Member" "" RETURN '(\NSMT.CHANGE.GROUP.COMPONENT DELETE.MEMBER REMOVE))
        ("Remove Owner" "" RETURN '(\NSMT.CHANGE.GROUP.COMPONENT DELETE.MEMBER.FROM.PROPERTY.ACL REMOVE
                                              Administrators))
        ("Remove Registered Object" "" RETURN (FUNCTION \NSMT.REMOVE.OBJECT))
        ("Remove Self" "" RETURN '(\NSMT.CHANGE.GROUP.COMPONENT DELETE.SELF REMOVE T))
        ("Remove User" "" RETURN (FUNCTION \NSMT.REMOVE.USER))
        ("Change Address" "" RETURN (FUNCTION \NSMT.CHANGE.ADDRESS))
        ("Change Default Domain" "" RETURN (FUNCTION \NSMT.CHANGE.DOMAIN))
        ("Change Forwarding" "" RETURN (FUNCTION \NSMT.CHANGE.FORWARDING))
        ("Change Login" "" RETURN (FUNCTION \NSMT.LOGIN))
        ("Change Password" "" RETURN (FUNCTION \NSMT.CHANGE.PASSWORD))
        ("Change Remark" "" RETURN (FUNCTION \NSMT.CHANGE.REMARK))
        ("Describe" "" RETURN (FUNCTION \NSMT.TYPE.ENTRY))
        ("List Aliases" "" RETURN '(\NSMT.LIST.OBJECTS ALIAS))
        ("List Administrators" "" RETURN (FUNCTION \NSMT.LIST.ADMINISTRATORS))
        ("List Clearinghouses" "" RETURN (FUNCTION \NSMT.LIST.CLEARINGHOUSES))
        ("List Domains" "" RETURN (FUNCTION \NSMT.LIST.DOMAINS))
        ("List Groups" "" RETURN '(\NSMT.LIST.OBJECTS MEMBERS))
        ("List Members" "" RETURN (FUNCTION \NSMT.TYPE.MEMBERS))
        ("List Objects" "" RETURN (FUNCTION \NSMT.LIST.OBJECTS))
        ("List Servers" "" RETURN (FUNCTION \NSMT.LIST.SERVERS))
        ("List True Groups" "" RETURN '(\NSMT.LIST.OBJECTS MEMBERS \NSMT.GROUP.FILTER))
        ("List Users" "" RETURN '(\NSMT.LIST.OBJECTS USER))
        ("Show Details of previously listed names" "" RETURN (FUNCTION \NSMT.SHOW.DETAILS))
        ("Type Entry" "" RETURN (FUNCTION \NSMT.TYPE.ENTRY)
              EXPLAINSTRING "Type Entry -- same as Describe")
        ("Type Members" "" RETURN (FUNCTION \NSMT.TYPE.MEMBERS)
              EXPLAINSTRING "Type Members -- same as List Members")
        ("Uncache All Clearinghouses" " [confirm]" CONFIRMFLG T RETURN '(\NSMT.UNCACHE T))
        ("Uncache Clearinghouse for domain" "" RETURN (FUNCTION \NSMT.UNCACHE))
        ("Uncache Local (force Maintain to refetch some info)" " [confirm]" CONFIRMFLG T RETURN
              (FUNCTION \NSMT.CLEAR.NAME.CACHE))
        ("Quit" " [confirm]" CONFIRMFLG T RETURN NIL)))

(RPAQQ *NSMAINTAIN-ABORT-ITEM* ("" "" EXPLAINSTRING "<cr> - abort" RETURN NIL))

(ADDTOVAR CH.PROPERTIES (ALIAS 1)
                        (BOOT.SERVICE 10026))

(ADDTOVAR *NSMAINTAIN-DESCRIPTIVE-PROPERTIES* 10000 10001 10002 10003 10004 10005 10006 10007 10008 10009 10010
                                                    10011 10012 10013 10014 10015 10016 10017 10018 10019 10020
                                                    10021 10022 10023 10024 10026)

(ADDTOVAR *NSMAINTAIN-IGNORE-PROPERTIES* 6 7 10027 20003 20002 20101)

(ADDTOVAR *NSMAINTAIN-PROPERTY-FORMATS*
          (4 CLEARINGHOUSE . NETWORK.ADDRESS.LIST)
          (8 RECORD (SIMPLE BOOLEAN)
             (STRONG BOOLEAN))
          (30 . NSNAME)
          (31 CLEARINGHOUSE . MAILBOX.VALUES)
          (10000 . STRING)
```

```
          (10001 . STRING)
          (10002 . STRING)
          (10003 . STRING)
          (10004 . STRING)
          (10005 . STRING)
          (10006 . STRING)
          (10007 . STRING)
          (10008 . STRING)
          (10009 . STRING)
          (10010 . STRING)
          (10011 . STRING)
          (10012 . STRING)
          (10013 . STRING)
          (10014 . STRING)
          (10015 . STRING)
          (10016 . STRING)
          (10017 . STRING)
          (10018 . STRING)
          (10019 . STRING)
          (10020 . STRING)
          (10021 . STRING)
          (10022 . STRING)
          (10023 . STRING)
          (10024 . STRING)
          (10026 . STRING)
          (10029 . STRING)
          (10030 . STRING)
          (10032 . STRING)
          (10034 . STRING)
          (10035 . STRING)
          (15002 . STRING)
          (20000 CLEARINGHOUSE . USERDATA.VALUE)
          (20001 GAP . RS232CData)
          (20006 SEQUENCE NSNAME)
          (20007 . NSNAME)
          (20102 GAP . RS232CBack)
          (29965 . STRING)
          (30005 . NSNAME))

(ADDTOVAR *NSMAINTAIN-MEMBER-PROPERTIES* 3 20006)

(RPAQ? *NSMAINTAIN-MEMBER-THRESHOLD* 3)

(RPAQ? *NSMAINTAIN-SHOW-GROUP-ACCESS* )

(DECLARE%: EVAL@COMPILE

(CL:PROCLAIM '(CL:SPECIAL *NSMAINTAIN-MEMBER-THRESHOLD* *NSMAINTAIN-SHOW-GROUP-ACCESS*))

(CL:PROCLAIM '(GLOBAL *NSMAINTAIN-MEMBER-PROPERTIES* *NSMAINTAIN-PROPERTY-FORMATS*
                     *NSMAINTAIN-IGNORE-PROPERTIES* *NSMAINTAIN-DESCRIPTIVE-PROPERTIES*))



(DEFMACRO WITH-CHS ((STREAMVAR DOMAIN)
                    &BODY BODY)
   `[LET [(,STREAMVAR (\NSMT.COURIER.OPEN ,DOMAIN]
         (AND ,STREAMVAR (CL:UNWIND-PROTECT
                           (PROGN ,@BODY)
                           (CLOSEF? ,STREAMVAR))])

(FILESLOAD (LOADCOMP)
       CLEARINGHOUSE)

(DECLARE%: EVAL@COMPILE

(RPAQQ \CH.BROADCAST.SOCKET 20)

(CONSTANTS \CH.BROADCAST.SOCKET)
)

(DECLARE%: DOEVAL@COMPILE DONTCOPY

(GLOBALVARS *NSMAINTAIN-COMMANDS* *NSMAINTAIN-ABORT-ITEM* CH.PROPERTIES)
)

(CL:PROCLAIM '(CL:SPECIAL *USER* *LASTDOMAIN* *LASTNAME* *LASTGROUP* *LASTLIST* *LASTSTRING* *LAST-MEMBERSHIP*
                     *SERVERTYPES* *ALLTYPES* *OBJECTTYPES* *DEFAULTDOMAIN* *REAL-NAME-CACHE* *DOMAIN*))

(DECLARE%: DOEVAL@COMPILE DONTCOPY

(LOCALVARS . T)
)

(RPAQ *NSMT-MENU-FNS*
      [CL:REMOVE-DUPLICATES (FOR ENTRY IN *NSMAINTAIN-COMMANDS* WHEN [LISTP (SETQ ENTRY
                                                                                 (CADR (MEMB 'RETURN ENTRY]
```

```
                                    COLLECT
                                    (IF (EQ (CAR ENTRY)
                                            'FUNCTION)
                                        THEN
                                        (CADR ENTRY)
                                        ELSEIF
                                        (EQ (CAR ENTRY)
                                            'QUOTE)
                                        THEN
                                        (CAR (LISTP (CADR ENTRY])
)

(PUTPROPS NSMAINTAIN COPYRIGHT ("Xerox Corporation" 1985 1986 1987 1989 1990 1991 1992))
```

## FUNCTION INDEX



## VARIABLE INDEX



## CONSTANT INDEX



## MACRO INDEX