Release Notes{ XE "Installation" }

NOIS is National On-Line Information Sharing. It is used to track requests for service. It is used on Forum and can be implemented for local use. NOIS can be used as only an M application or as a client/server application with M as the server and Windows as the client.

NOIS software is available via FTP from the CIOFO anonymous servers.

Hines	152.129.1.110
Albany	152.127.1.5
Salt Lake	152.131.2.1

These files are available

NOIS1_0.KID	the package install	
NOIS1_0.ZIP	the client files (includes these files)	
	NOIS.EXE	NOIS client for Windows
	NOIS.HLP	Windows help for NOIS
	NOISWIN.HLP	Windows help for client
	NOISWIN.CNT	file used by NOISWIN.HLP
	README.TXT	install instructions for client
NOIS1_0_SRC.ZIP	source code for client (not necessary for using NOIS)	
NOIS1_0_UM.PDF	user manual	
NOIS1_0_TM.PDF	technical manual (with release notes, install, and security)	

Note: This documentation may have some older names for files that have been changed to more generic names. Ex. NOIS ISC is now NOIS OFFICE file, NOIS SPECIALIST file is now NOIS USER DEFAULTS file.

Installation{ XE "Installation" }

Get NOIS software (NOIS1_0.KID) from anonymous directory using FTP.

The NOIS build use an environment check routine to check if a much older version exists. If you have an old version of NOIS (<1.0T1) the environment check will prevent you from installing the software, see instructions to uninstall NOIS on page 5.

Account has to have:
VA FileMan 21
Kernel 8
Mail 7.1

Globals used by NOIS (take any appropriate action):
FSC (used as definition files with NOIS)
FSCD (data files with NOIS)

Remove old routines
Delete all FSC* routines.

Install NOIS
Load NOIS1_0.KID and install.
If you have a previous test version of NOIS you may be prompted to rename some files - answer YES.

Set up NOIS for your Configuration:
Local (see page 3) or National (see page 4)

Test NOIS
Use the NOIS options to exercise the package. You can practice entering data and using the NOIS features while working out any adjustments to the setup. If you want to clear all NOIS data (but leave your files set up) and start over before 'going live' see Clear All Data on page 5.

Client/Server
To use NOIS as a client/server application with Windows, you must have Broker 1.1 installed. See readme.txt file in NOIS1_0.ZIP for using the client.

Local Configuration{ XE "Local Configuration" }

Set up a Resource Device{ XE "Resource Device" }:
Create one new device - NOIS Updates
Example entry in Device file:

```
NAME: NOIS UPDATES                    $I: NOIS UPDATES
LOCATION OF TERMINAL: NA              RESOURCE SLOTS: 1
TYPE: RESOURCE
```

Set up NOIS Parameters:
In the NOIS Parameter file enter the NOIS UPDATES resource in the Default Resource field. Change the Use field to SITE.

Set up NOIS Specialists:
Enter in the people to be editing NOIS calls in the NOIS User Defaults file. You may wish to include ADPACs as NOIS Specialists. Only the name of the person is required to be entered in order for them to begin using NOIS. If the person is to be able to set up other's characteristics then they should have 'YES' entered in the NOIS Coordinator field.

Set up the Office (optional):
If you have multiple sections in your IRM service or have specific people that specialize in certain areas or packages, you can configure the NOIS Office file similar to how CIOFOs support national software. You can have one entry of IRM or create multiple entries you have multiple supporting offices.

Set up the NOIS Location file:
Make entries for services or locations in your hospital. Each entry should have a name and abbreviation. Enter the Site Contacts and Site Phone #s (these will be the contacts at these locations). Enter the office.

Set up the NOIS menu{ XE "NOIS menu" }:
Enter the FSC MENU NOIS option in the XUCOMMAND menu option or assign this as a secondary menu to those that will use NOIS.

Set up the Nightly Job{ XE "Nightly Job" }:
Use TaskMan management to schedule the FSC TASK AGE option the Queued to run at T+1@0100 with a Reschedule Frequency of 1D.

Set up Common Lists:
See page 13.

National Configuration{ XE "National Configuration" }

Set up Resource Device{ XE "Resource Device" }s:
Create 7 devices - NOIS Updates 1 through NOIS Updates 7
Example:

```
NAME: NOIS UPDATES 2                    $I: NOIS UPDATES 2
LOCATION OF TERMINAL: NA                RESOURCE SLOTS: 1
TYPE: RESOURCE
```

Set up Resources for the Offices:
In the NOIS Office file enter an appropriate Resource.
Example: for SLC enter NOIS Updates 5, for Albany enter NOIS Updates 1, etc.

Set up Default Resource:
In the NOIS Parameter file enter in a resource in the Default Resource field.

Set up the NOIS Specialists:
Enter in the names of those to be editing NOIS calls in the NOIS User Defaults file. If the person is to be able to set up other's characteristics then they should have 'YES' entered in the NOIS Coordinator field.

Set up the NOIS Location file:
Enter the Site Contacts and Site Phone #s in the NOIS Location file.

Set up the NOIS menu{ XE "NOIS menu" }:
Enter the FSC MENU NOIS option in the XUCOMMAND menu option.

Set up the Nightly Job{ XE "Nightly Job" }:
Use TaskMan management to schedule the FSC TASK AGE option the Queued to run at T+1@0100 and the Reschedule Frequency at 1D.

Set up Common Lists:
See page 13.

UnInstall{ XE "UnInstall" }

If you have an old version{ XE "old version" } (<1.0T1) of NOIS or if you want to remove your installation of NOIS, you can remove the package by using the routine D CLEAN^FSCCLEAN from programmer mode. This routine is the environment check routine and will be loaded on the system before you install. **Warning!! This removes all NOIS files, dictionaries and data**. You should also delete all FSC namespaced options, protocols, list templates, and routines.

Clear All Data{ XE "Clear All Data" }

If you want to keep your NOIS setup but start over entering data, you can wipe out all of the NOIS calls by using the routine D FRESH^FSCCLEAN from programmer mode. **Warning!! This wipes out all NOIS calls.** It does not delete any NOIS locations, specialists, packages, or list definitions.

Design{ XE "Design" }

NOIS is basically organized around two types of files - data and definition.

The data files are in the FSCD global. The data files store the NOIS calls and notifications and processing information. These files are flat. The NOIS Call file has several cross references for fast access.

The definition files are in the FSC global. These files define packages, NOIS Specialists, priorities, etc. The definition files are rather simple, but some of the files do have multiples.

NOIS is designed to be a very open system, that is, many people can access the information in NOIS. To get to information easily, calls are grouped by using lists. NOIS also provides a query capability for searching for calls. The query statements define a type of NOIS call. As calls are edited, the list definitions are checked for including calls on lists. The calls are also checked to trigger notifications to interested users. Query statements (in the form of list definitions) are again used to define these requests for notification.

List Manager is used as the user interface for most NOIS activity.

Data is entered using conventional methods (DIE and DIR calls). After data is stored, a series of processes are performed in the background (see Background NOIS in the User's Guide).

There is little effort to do elaborate reporting with NOIS. Rather, it's assumed that information can be off loaded to spreadsheets or other applications better suited for in-depth review. NOIS does provide a way of displaying commonly used information and focuses on providing samples of data.

Globals{ XE "Files" }

Data files{ XE "Data files" } -- the FSCD global{ XE "FSCD global" }

The FSCD global is used to store the dynamic NOIS information. This global may get large proportional to how much NOIS is used. An average NOIS call is about 6 Kbytes. The NOIS Call file is the main data file in NOIS. It's cross references are used to create Index type lists.

Definition files{ XE "Definition files" } -- the FSC global{ XE "FSC global" }

The FSC global is used to define NOIS elements. This global is relatively small with little growth.

TMP global{ XE "TMP global" }

NOIS uses the TMP global to store job specific arrays that are too large for the local partition. These always use a FSC namespace before the job number. These arrays are cleaned up when leaving a NOIS option.

XTMP global{ XE "XTMP global" }

NOIS uses the XTMP global to temporally store information.
^XTMP("FSC COUNTS"   stores option and protocol counts
^XTMP("FSC LIST DEF"   stores a processed copy of list definitions
^XTMP("FSC UPDATE"   stores the processing times of the nightly job
The information in this global is either of temporary value (usage studies) or can be rebuilt from NOIS files. The "FSC LIST DEF" node is used to optimize performance (it stores list definitions that are processed for faster queries). The list definitions are rebuilt if they do not exist. This information is purged periodically by Kernel utilities.

Definition Files{ XE "Files" }

These are the Definition files that you can set up.

Generally these are NOT EDITED:{ XE "**FSCD global**" **}**

Status
Priority
Fields
Condition
Work Group Type (used to categorized Work Groups)
Format Type (used to categorize Formats)
Entry Person Type (used to categorize Users)

Files to set up first:

Parameter
Office (used to categorize Locations)
Location Type (used to categorize Locations)
System (used to categorize Locations)
Work Group (used to categorize Packages)
Functional Area (used to categorize Specialists)
Task

The set up these files:

Location
Specialist (User Defaults)
Package

then set up:

Module (a version of a Package)
Subcomponents (sub categories of a Package)

finally set up:

List Definitions
Format
Reports

File Setup{ XE "Files" }

Configure and test NOIS until it works the way you want it to at your site. Use FRESH^FSCCLEAN to wipe out any test data. This does not wipe out your definition files. You can do this repeatedly until you plan to 'go live'. Once you have implemented NOIS without any intentions of removing the data, you should NOT MAKE DELETIONS to these definition files (List Definitions should only be deleted using the hidden action 'Define Lists', do not delete them using FileMan). You can make additions to the Locations, User Defaults, Package, and Module at any time (note: all file names start with the acronym NOIS)

Files that should not need editing

If you must edit these files be aware of potential problems.

Status file entries are hard coded in NOIS. This is because of the two-tier level of status, regular and referred (this used to be support and development). These two tiers were designed for national use and may not apply to your site, but it is very important to not remove or add any status entries. It is also important to not rename OPEN, CLOSED, CANCELLED, REFERRED, or AWAITING PATCH since the software uses these to control the processing of NOIS calls..

Priority entries can be edited so that only certain entries are allowed when non-specialists log calls. The priority, MISC, is used in the software to exclude alerting on these calls when using notify by affiliation. The priority, ROUTINE, is hardcoded as a default.

The files Field and Condition have synonyms that can be changed but do not add or remove entries to these files.

Work Group Type, Format Type, and Entry Person Type files should not be edited.

Files to set up first

The Parameter file should have the NAME field as NOIS (the name of this application), and USE field should be SITE for local use. The LOCATION field can be the name of your site (Ex. Boise, Denver, etc.). Enter the other fields at your discretion.

Office (formerly NOIS IRMFO) should be set up with at least one entry. The entries are used to categorize locations (each entry in the location file should have an office entered). The internal entry is used in the NOIS call ID. Ex. LAB-0997-10012 has a 1 for the location LAB. If no office was entered for the location, the call ID would be LAB-0997-N0012. Office is also useful for doing load balancing of NOIS processing (see documentation on Background NOIS in User Guide). Use multiple offices for

multidivisions, remote locations, different sites at a consolidated site. Do not enter more than 9 entries.

The Location Type is used on a national level to differentiate between the sites configured with Alpha or 486 hardware. At your site you will want to create any categories that may be useful. Ex. Clinics, Wards, Administrative, etc. or use it for defining the sections or buildings at your facility. This field may be useful for searching and clumping together locations.

The System file is very similar to the Location Type. Populate it with entries to help categorize your locations.

The Work Group file is used primarily to categorize packages (Package Groups). If you have groups of packages that are supported by certain people or teams, then define those here. Ex. Hardware, Network, Printers, Clinical, Fiscal, Management, Security. Be aware that any package can only belong to one Package Group (a package can belong to hardware or printers but not both).

Functional Area and Task are only applied to a call when it is closed so they are not useful for categorizing open calls. The functional area is used to categorize the specialist that worked with the call. The task is a category of what was done (problem resolution, information dissemination, etc.). Add any entries you want to these files.

Files that require quite a bit of setup

Location should be set up with any locations you support. For national use these entries are medical centers, for your use these will be wards, clinics, and any other other locations you may wish to add. Get a printout of your hospital location file to get a feel for what locations you want to include. Each location must have the 3 character abbreviation field entered. This field is what is displayed in the call ID. If an abbreviation is not entered, the call ID will begin with 'ZZZ'. Include a OFFICE with each location. The contact may be the ADPAC responsible for the most common software being used at that location. Include a TYPE and/or SYSTEM to categorize the location. Ignore the DOMAIN, NT DOMAIN fields for local use. Use synonyms to make lookups easier for other names besides the name and abbreviation field. Usually the NOTIFY METHOD and NOTIFY EVENT are not entered.

User defaults (formerly NOIS SPECIALIST) should include anyone that edits NOIS calls. This would include anyone in IRM and may include ADPACs depending how you implement who is using NOIS. Enter the Office, Primary Functional Area, Notify Method as Alert, Notify Event as Edited. If the Specialist supports a group of packages then fill in those under Group Affiliations. If he/she supports packages not included in those groups they can enter those packages under Package Affiliations. An easy way to receive alerts on calls from any of these packages is to use the Notify by Affiliation field.

Package entries come with some current VistA software. You may include other entries, particularly hardware devices (Printer, Barcode, PC, etc.) or software (word processing, spreadsheet). Enter the Group field with a package group.

The Module/Version# is actually what is used when logging a NOIS call. Every module should have a package associated with it (so make the package entry first so that when you define the module you can assign it a package). Usually you use this file to define packages with a particular version number. You can clump related modules to a package. Ex. Printer and Barcode modules could be assigned the Hardware package. It's OK to change the package on a module after NOIS has been in use.

Subcomponets are used to categorize a module. Ex. Kernel may have subcomponents TaskMan, MenuMan, etc.

List Definitions should be set up with any common lists. You can then add any Storage, Active-Update, or Manual-Update lists using the Define List (DL) action on the hidden actions of NOIS. Become familiar with doing queries before creating an Active-Update or Manual-Update list.

Formats and Reports can be set up as needed. They act like templates to display only certain fields.

Options/Protocols/List Templates{ XE "Options/Protocols/List Templates" }

NOIS uses the namespace FSC on all options, protocols, and list templates. Option and protocols can be restricted to specific types of NOIS users by including a parameter in the entry action. There are very few options. FSC MENU NOIS is the main NOIS menu (actually it's the only NOIS option menu). There is one tasked option that is run nightly. List templates are used for List Manager screens. These list templates contain the menu protocols used by the NOIS Screens. Code for options and protocols are controlled by just two routines.

Remote Procedures{ XE "Options/Protocols/List Templates" }

NOIS uses only one remote procedure file entry, FSC RPC, for client server operation. The option for this remote procedure is included on the main menu for NOIS. Anyone having access to the NOIS application will also have access to using the Windows client.

Routines{ XE "Routines" }

NOIS uses the namespace FSC. There are two routines that are helpful in tracking down NOIS processing.

FSCOPT - All NOIS options (from the entry action) call this routine. It routes the option to the processing routine.

FSCLMP - All NOIS protocol actions call this routine. It is organized by the different List Manager screens used by NOIS. It routes the action to the processing routine.

Both of these routines check for access that can be set as a parameter from the option or protocol entry action (see Security on page 16).

Common Lists{ XE "Common Lists" }

Indexed Lists{ XE "**Indexed Lists**" **}**

NOIS calls are cross referenced by several fields. These cross references can be used to create index type lists (see the User Guides discussion of list types). Lists that can be used to select entries require the field Index Field to be entered (these selections should be named ending in a colon). List that explicitly define an entry require the internal number in the Index.

Examples:

```
NAME: PACKAGE:                          TYPE: INDEXED
  INDEX: "APAC"                         INDEX FIELD: PACKAGE


NAME: SPECIALIST::                      TYPE: INDEXED
  INDEX: "ASPEC"                        INDEX FIELD: SPECIALIST


NAME: ISC:                              TYPE: INDEXED
  INDEX: "AISC"                         INDEX FIELD: ISC


NAME: ISC-SLC                           TYPE: INDEXED
  INDEX: "AISC",5
```

Note: Use cross references in the NOIS Call file for making lists. To find any entry values (the example ISC-SLC uses the 5 in the index "AISC",5), you will need to know the internal entry for the cross reference. You'll find this by looking at the global for that entry.

The best way to see examples of these files is to look at examples on Forum. Use the File Setup option on NOIS and print these types of lists. Ex:

```
Select file: LIST
Select (P)rint, (S)earch, (I)nquire/Edit: PRINT
LIST NOIS LIST DEFINITION BY: NAME// TYPE
START WITH TYPE: FIRST// INDEXED    USES INTERNAL CODE: I
GO TO TYPE: LAST// INDEXED    USES INTERNAL CODE: I
  WITHIN TYPE, LIST NOIS LIST DEFINITION BY:
FIRST PRINT FIELD: [CAPTIONED

DEVICE


NOIS LIST DEFINITION
```

```
PAGE 1
---------------------------------------------------------------------

        TYPE: INDEXED

NAME: ALL CALLS                             TYPE: INDEXED
  INDEX: "ALL"                              RESTRICTION LIMIT: -1

NAME: EMPTY                                 TYPE: INDEXED
  INDEX: "AEMPTY"

NAME: CIOFO, DEV OPEN:                      TYPE: INDEXED
  INDEX: "ADID"                             INDEX FIELD: DEVELOPING OFFICE

NAME: CIOFO, DEV REFERRED:                  TYPE: INDEXED
  INDEX: "ARID"                             INDEX FIELD: DEVELOPING OFFICE

NAME: CIOFO, DEV:                           TYPE: INDEXED
  INDEX: "AISCD"                            INDEX FIELD: DEVELOPING OFFICE
  RESTRICTION LIMIT: -1
```

The restriction limit can be set to -1 which prompts the user to make a restriction on the list. This is used for lists with a large number of calls.

Storage-Only, Manual-Update, and Active-Update lists can be created by users using the Define List action on any of the NOIS Screens.

Special Fields for Grouping Calls{ XE "**Grouping Calls**" **}**

NOIS calls store data in fields that are useful for grouping calls with common characteristics. These files can be used to orgainze working with calls:

NOIS Office	clumps locations (supporting office) and packages (developing office)
NOIS Package Group	clumps packages - ex. support teams, work groups
NOIS Location Type	clumps locations - ex. DSM vs MSM or clinic vs ward
NOIS Subcomponent	Subcomponents of packages (used only on Refer to Dev)

Maintenance{ XE "Maintenance" }

These events tend to require updating:

New Packages or Updates{ XE "New Packages or Updates" } to Module/Versions.
A new package requires that it be entered in the NOIS Package file. Do not delete entries
from this file. A new entry needs to be made in the Module/Version file whenever their
is a new package or a new version of an existing package. The version type (current,
beta, old) should be updated on the new version and any previous entries.

Changing NOIS Specialists{ XE "NOIS Specialists" }
New NOIS Specialists can be added by entering their name in the NOIS Specialist file.
These entries should not be deleted. If a NOIS Specialist no longer is valid then the field
INACTIVE should be set to YES.

Purging{ XE "Purging" } and Archiving{ XE "Archiving" }

NOIS calls can be individually purged by changing the status to canceled. The field
PURGE DAYS, CANCEL in the NOIS Parameter file can be set to the number of days
the call will remain on the system after being canceled. Calls can be automatically
purged by setting the PURGE DAYS, CLOSE field. Calls will be purged by the nightly
job whenever calls of old module/versions have been closed for the number of days set in
this field.

NOIS does not have an archiving utility. VA FileMan archiving can be used to remove
NOIS Calls, but these calls cannot be restored.

Security{ XE "Security" }

Information entered into NOIS calls should not contain patient identifications or information that would not normally available to any user.

NOIS does not require any security keys{ XE "security keys" }. Local keys may be made if you would like to restrict certain menu options. NOIS does restrict options and protocol actions by using a parameter in the entry action field. The options can be restricted to User (default), NOIS Specialist, or NOIS Coordinator.
Examples:
The option FSC IMPORT DATA with the entry action D
OPT^FSCOPT("IMPORT","SPEC") restricts the use of this option to NOIS Specialists.

The protocol FSC EDIT NEW with the entry action D EDIT^FSCLMP("NEW","SPEC") restricts the use of this protocol action to NOIS Specialists.

The protocol FSC EDIT NOTE with the entry action D
EDIT^FSCLMP("NOTE","USER") allows anyone to use this protocol action.

Note: If the 2nd parameter is not used, it is assumed to be "USER".

Support{ XE "Support" }

Those that use NOIS the most are the ones that enter the information on the NOIS calls. Your IRM or CIOFO support person usually knows most of the features of NOIS. The NOIS package also has NOIS Coordinators that can make changes to how NOIS works.

Support and questions on how to use NOIS can be found by calling the National Help Desk (888) 596-4357 or logging a NOIS on Forum

Index

NOIS

Technical Reference

Department of Veterans Affairs
VistA
February 1998

Table of Contents