VistA National Patch Module Guide

Department of Veterans Affairs

September 2021

Version 2.9

Revision History

Date	Version	Description	Author
September 2021	2.9	Correct process flow and update Host file content.	HSP
June 2021	2.8	Remove reference to Health Product Support Release Coordinator and expand information on Verifier role.	Enterprise Portfolio Management Division (EPMD) – Health Services patch release teams and VistA Office
March 2021	2.7	Update for version removal	SACWG
March 2020	2.6	Updated to use new Patch Templates	Software Engineering/HPS
July 2019	2.5	Updated to correspond with name changes made to developer checklists	Software Engineering/HPS
August 2018	2.4	Updated links to the VistA Patch Template Document without Instructions and the VistA Patch Template Document with Instructions on Page 3.	Quality Continuous Improvement Organization
March 2017	2.3	Updated to combine NPM Guide and Template into a single document. Updates also include instructions related to Host Files, roles, and other cleanup resulting from changes to process over several years	Software Engineering/HPS
July 2014	2.2	Upgraded to MS Office 2007-2010 format, update to current ProPath documentation standards, and edit to latest Section 508 guidelines	Process Management
December 2010	2.1	Updated document to current roles, made minor cosmetic changes, remove outdated information, updated information concerning the new National Patch Module Patch Template	Process Management
9/22/2010	2.0	Changed references to E3Rs to New Service Requests (NSRs). Added references to the new National Patch Module Patch Description template.	Process Management Service
12/10/2009	1.6	Updated name of file as well as the title page. Removed intentionally blank pages	Process Management Service
10/02/2008	1.5	Updated per organizational changes	Process Management Service

Date	Version	Description	Author
4/20/2006	1.4	Removed references to routine mapping and updated the link to VA Directive 2001-023 to the ASIS link	SEPG/SQA
4/1/2005	1.3	Updates to Developer and Second Developer responsibilities	Paul Rowny
6/28/2004	1.2	Updated Organizational Names	SEPG
3/25/2002	1.1	SQA Manager title changed to SQA Analyst	PDS SQA Team
10/17/2001	1.0	Initial Version	SQA/SEPG

Table of Contents

National Patch Module Guide

1. Purpose

The National Patch Module Guide describes the purpose, roles, responsibilities, and steps for the initiation, development, creation and release of patches to VHA Information Systems and Technology Architecture (VistA) products via the National Patch Module (NPM). NPM resides on FORUM and is a software package that provides a database for the distribution of software patches and updates. The last update to the NPM software was in 1992, when VistA was still known as the Department of Veterans Affairs' Decentralized Hospital Computer Program (DHCP). As a result, any package output examples provided in this guide will still have reference to DHCP.

2. Responsibilities

The following roles and responsibilities are involved during the initiation, development and release of VistA patches.

2.1. Team Leader

For the purpose of this document, the Team Leader will be defined as the Development Project Manager or Team Leader responsible for the patch. The team at the least will consist of a Developer, Software Quality Assurance (SQA), and Verifier. Other personnel that may be relevant to the patch being developed should be added to ensure the successful development, testing, and release of the patch. The Team Leader will be responsible for the oversight and management of the patch development process as follows:

- Identify the Verifier responsible for performing final review and release of patch
- Identify additional relevant personnel, including their roles and responsibilities
- Provide access to the document repository to appropriate team members
- Include appropriate team members on project and test site calls

2.2. Developer

When the Developer receives the assigned defect or enhancement, the following tasks will be performed, not necessarily in the order listed. However, after creating the patch stub in the NPM, and before modifying any routine, it is the responsibility of the developer to coordinate any potential routine conflicts. The developer of the other patch(es) listed must be contacted in order to coordinate the development and release of the affected patches. At any time, however, the Developer may inspect the “Routines that Overlap in Patches” report on the Developer’s menu on FORUM to review all of the overlaps. The Developer is responsible for completing the

VistA Primary Developer Review Checklist and saving an electronic copy of the completed checklist.

- Investigating the defect to determine if a patch is warranted.
- Analyzing the Business Requirements and Software Requirements.
- Creating the patch stub in the NPM.
- Updating the defects status of the appropriate problem incident(s).
- Alerting other team members (Development Project Manager, Verifier, SQA Analyst, Secondary Developer, Technical Writer, etc.) that the patch is under development.
- Making modifications.
- Unit testing all modifications.
- Creating the patch-tracking message and sending the message on FORUM to test sites, and the appropriate Verifier.
- Coordinating Initial Operating Capability (IOC) Testing.
- Coordinating completion of patch with the SQA Analyst and Verifier.

2.3. SQA Analyst

The SQA Analyst is responsible for:

- Ensuring that all SQA policies, practices and guidelines are followed for a specific project.
- Being a member of the development team if applicable, as determined by the Development Project Manager or Team Leader having responsibility for the patch.
- Ensuring the VistA SQA Checklist is completed for each test version they review.
- Coordinating completion of patch with the Developer and Verifier.

2.4. Secondary Developer (Optional)

The Secondary Developer is optional, depending on the complexity of the project. The need for a secondary review is determined as early in the process as possible.

When done, the second developer is responsible for:

- Providing an independent peer review of specific components of the patch focusing on the completeness and appropriateness of the patch's ability to address the issues or problems it is supposed to correct.
- When applicable, the Secondary Developer is responsible for completing the VistA Secondary Developer Review Checklist and saving an electronic copy of the completed checklist.

2.5. Technical Writer (Optional)

The Technical Writer is responsible for:

- Preparing any new or modified documentation applicable to the patch. (e.g., Technical, User Manuals, Release Notes, etc.)

2.6. Functional Analyst (Optional)

The Functional Analyst is responsible for:

- Reviewing and updating function point data, if applicable to the patch.

2.7. Verifier

The Verifier is responsible for assisting with the coordination of activities related to testing and release of the patch and is responsible for:

- When applicable work with IOC sites during testing.
- When applicable complete the Verifier checklist
- When applicable work with SQA, to ensure the patch hold date is removed.
- Coordinating with Developer and SQA on completion of patch
- Changing the status of a 'completed/unverified' patch to 'verified'
- At time of release, (marking verified), patch compliance date is set with the agreed upon date from development team.
- When applicable move the build/executables and documentation to the National File Server [SOFTWARE] directory.

3. Steps

3.1. Develop the Patch Resolution

For defect repair patches:

- When applicable review the problem incident(s)
- Attempt to duplicate the issue.
- Perform thorough assessment and document findings in problem incident(s)
- Begin work on the patch

For enhancement patches:

- Review user stories/requirements.

For all patches (defect repairs or enhancement patches):

- **Create Patch Stub**. Detailed instructions for creating the patch stub can be found in Appendix 1.
- The patch stub should be as complete as possible, given the amount of information known at this time.
- The Patch Description contained in patch stub should include all the required sections as outlined in the VistA Patch Templates.

  There are two templates. One template is used for patches deployed using PackMan. The second template is used for patches deployed using Host Files.

  VistA Patch Template Document - PackMan:
  https://vaww.oed.wss.va.gov/process/Library/vista_patch_template_with_instructions_packman.docx

  VistA Patch Template Document – Host Files:
  https://vaww.oed.wss.va.gov/process/Library/vista_patch_template_with_instructions_host_file.docx
- **Move KIDS Build to NPM on FORUM.** If the development account connection to the NPM on FORUM hasn't been established yet, see detailed instructions in Appendix 5.
- **Update the Problem Incident**. Make a note within the problem incident entry and notate the patch number. This is beneficial if information regarding our progress is available. This is an opportune time to get feedback and recruit test sites.
- **Coordinate with New Development**. If new development is underway on the product being patched, it is necessary to review the final patch resolution with the appropriate Verifier or Functional Analyst prior to releasing the patch to the test sites.
- **Modification of Routines**. Prior to making any modifications, pre-patch checksums should be determined. This is done to validate that you are starting with the proper version of the routine as found in an account that reflects the nationally released version. A backup copy of the routines prior to beginning modifications should be made in the event that they need to be restored. Routines should be saved to a Host File System (HFS) file or in a MailMan message.
- **Update Routine File**. The Routine file may be updated to add any new routines included in the patch by doing one of the following:

  a. Use the Kernel Installation & Distribution System (KIDS) Utility [XPD UTILITY], Update Routine File [XPD ROUTINE UPDATE].

  b. Run ^XINDEX and answer "YES" to the "Save parameters in ROUTINE file?" prompt.
- **Modify Data Dictionary**. If the patch contains changes to data dictionaries, regardless of significance, a mail message must be sent to the Database Administrator (DBA) for concurrence. This message should include a description of the change and request formal

approval to make the change. The message should also be sent to the Team Leader, SQA Analyst, and Verifier. This request should be made as early as possible in the process.

NOTE: Correspondence to the DBA should be directed to the **VA OIT VISTA DBA GROUP OEDVistADBA@va.gov** mail group for concurrence/approval. The message should also indicate the **Patch Number** the change(s) will be exported in and the approximate schedule release date (i.e., Late/early January 2017, etc.…) of the patch.

- **Check for Integration Control Registrations (ICRs)**. Check each component included in the patch for appropriate Integration Control Registrations (ICRs). Obtain updates to existing ICRs when necessary. Review all external references for ICRs in all components of this patch.

  NOTE: Requests for ICRs shall be sent to the **OIT PD Integration Control Registrations ICRS@va.gov** mail group, appropriate custodial development team(s) if known. The message should identify the planned Patch Number and approximate date for release. This should be done as early in the development process as possible to reduce the risk of rework due to disapproval.

- **Review Messaging**. If any form of messaging is affected, a mail message must be sent to HL7 Message Administrator **VA OIT EPMO Messaging Administration VAOITPDADCTA_MSGADM@va.gov** mail group.

- **FDA Clearance**. Patches that alter the underlying platform, or interface with devices that are FDA regulated, must be submitted to the developer of those applications for impact review and analysis. The analysis will result in one of the following actions:

  a. Analysis determines the patch does not impact the performance of the FDA regulated device and a "clearance" will be issued and documented in the patch description.

  b. Analysis determines the patch does impact the performance of the FDA regulated device and successful integration/regression testing will be performed and documented in the patch description.

- **Analysis of External Impacts**. Patches should be analyzed for their potential impact on external systems (e.g., national databases at AITC or elsewhere). When a potential impact on an external system is discovered, appropriate personnel should be notified for requirements gathering, cost estimation, and work delivery activities.

- **Technical Reviews**. If changes to the structure of VistA are expected (i.e. adding a new module, significantly changing an existing one, etc.), you must submit the changes for an Architecture Review.

- **Unit Testing of Patch**. Patches shall be tested thoroughly by the developer.

- **Secondary Developer Reviews Patch**. When applicable, the Secondary Developer reviews components of the patch identified by the primary developer as needing a peer review.

- **SQA Analyst Reviews Patch**. The SQA Analyst thoroughly reviews the patch using the VistA Software Quality Assurance (SQA) Checklist.

Each Distribution external to development to Software Quality Assurance (SQA), and User Acceptance Testing (UAT) will have the following completed:

- ✓ All required components in the NPM completed (Patch Description, installation instructions, Defect Description, etc.)
- ✓ Primary Developer Review Checklist (REQUIRED for submission to SQA) *
- ✓ Secondary Developer Review Checklist (OPTIONAL) *
- ✓ VistA Software Quality Assurance (SQA) Checklist *
- ✓ Supporting Documentation
- ✓ Host File Build with the following
  - Header Comment with each build within the host file incorporating the versioning. For example:
    Header Comment: Package namespace*V.V*#### v#, Package namespace*V.V*#### v#...
    NOTE: The Header Comment with versioning is required for all KIDS host file distributions regardless if they are a single or a multi-build patch.
  - Host File name with the package namespace, package version, patch number, and the version of the patch being deployed

- **Preparing for Initial Operating Capacity (IOC) Testing**. You must enter your VistA Office (VO) Change Control Board (VOCCB) Patch Decision Form to VOCCB at the time your patch enters IOC or if an informational patch only.

https://dvagov.sharepoint.com/sites/OITEPMOVistAOffice/SitePages/VOCCB%20PD.aspx

The patch is provided to the test sites and Verifier for testing and reporting of anomalies and other related issues/problems.

For IOC Testing, the developer will complete the following:

- ✓ During IOC testing, the version numbers will be retained for the Host File name as well as the header comments within the file. Subsequent iterations of the build will increment the version numbers accordingly and be reflected in the Header Comment(s).
- ✓ Versioning will be tracked using the build deployed in the Header Comment and in the Host File name. Verify all components are not changed only Header Comment field
- ✓ Primary Developer Review Checklist (REQUIRED for submission to SQA) *
- ✓ Secondary Developer Review Checklist (OPTIONAL) *
- ✓ VistA Software Quality Assurance (SQA) Checklist *
- ✓ Supporting Documentation such as the Release Notes, the Version Description Document, the Deployment, Installation, Back-out, Rollback Guide (DIBRG), and the Technical and User Manual updates

- **Create Patch Tracking Message**. The primary developer of the patch will initiate one message in order to track patch progress. It will be sent to all patch team members (other developers), SQA Analyst, the appropriate Verifier, and a contact person at each test site.

Test site contacts are identified when the IOC MOU is provided from each site. For most projects, the Patch Tracking Message is sent to the recipients on FORUM or alternately MS Outlook. Some projects may use an alternate approach, agreed upon between the development team, SQA, and Verifier.

a. Include any information team members may need to know in the original message.

b. Each new test version should be documented on the patch-tracking message.

c. All pertinent dialogs dealing with the patch should be contained in the patch-tracking message.

d. The name of the message must be "Patch Designation - Patch Tracking Message" (e.g. XYZ*2*12 – Patch Tracking Message).

- **Additional Updates to the Problem Incident**. Notes should be added to appropriate Incident throughout the patch development and testing processes.
- **Completion of the Patch**. Ensure the patch is accurate, complete and correct.
- **Update Function Point Data**. If the patch alters functionality, function point data should be reviewed and revised if necessary.
- **Patch Release Check**. Ensure all associated patch(es) are identified and entered in the Patch Release Check multiple. If there are patch dependencies, enter the required or associated patch(es) in this field. You should designate if the patch(es) entered in this field should be verified/installed prior to the installation of the current patch you are creating. See the following example.

```
Select PATCH RELEASE CHECK: XYZ*2*7 EXCEPTION MESSAGE
Are you adding 'XYZ*2*7' as a new PATCH RELEASE CHECK (the 1ST for this
DHCP PATCHES)? No// Y  (Yes)

     PATCH RELEASE CHECK REQUIRED FOR VERIFICATION:??

      Enter yes if the Patch in the 'PATCH RELEASE CHECK' must be verified and
installed prior to verification and installation of this Patch.


 PATCH RELEASE CHECK REQUIRED FOR VERIFICATION: Y YES

  REQUIRED FOR VERIFICATION: YES// <RET>

Select PATCH RELEASE CHECK:  <RET>


If you answer "YES", the following is displayed on the top of the patch:

Associated patches: (v)XYZ*2*7  <<= must be installed BEFORE 'XYZ*2*12'
```

3.2 Release the Patch

- When applicable complete Verifier Checklist final review and recommend release of patch on the Patch Tracking message.

POST Initial Operating Capacity (IOC) Testing and prior to National Release, the developer, SQA, and the verifier will complete the following:

- ✓ For Packman, SQA completes the patch for Verifier to release.
- ✓ For Host the developer will Transport the build to remove the patch versioning from both the KID filename and the Header Comment of the build. This should be a new file ready to release.
- ✓ The last test version of the KIDS build file, signed off on by the IOC sites, will be retained for comparison purposes against the file ready to release file noted in the previous step.
- ✓ The host file will be moved by the agreed upon directory for Verifier to retrieve
- ✓ SQA will review the checksums and compare the ready to release host file against the previous version and make sure just the patch version was removed and nothing else was included or excluded from the ready to release build.
- ✓ Notify VA SQA for VA Review with e-mail with Subject Heading: "DG_53_P####.KID Ready For Release" and cc the appropriate Product team or Verifier.
- ✓ VA SQA will review the routine checksums with the option Verify Checksums in Transport Global [XPD PRINT CHECKSUM] and compare the build using option Compare Transport Global to Current System [XPD COMPARE TO SYSTEM].
- ✓ Once VA SQA Review has cleared the ready to release host file, they will respond on the DG_53_P####.KID Ready For Release" message that Host File cleared SQA.
- ✓ SQA will change the patch status to "COMPLETE/NOT RELEASED".
- ✓ Verifier will perform final checks on the Host File against the last test version of the patch to ensure the only the expected differences are present.
- ✓ Verifier will move the build to /srv/vista/patches/SOFTWARE and update patch status to "RELEASED" once the file replication is verified.

- Using the RELEASER MENU [A1AE VERIFIER] on FORUM select Release a Patch (and/or edit internal comments) [A1AE PHVER] option.

```
Releaser Menu

Select Releaser Menu Option: Release a Patch (and/or edit internal comments)

Select PATCH: XYZ*2*7

Internal Comments to developers/releasers for: XYZ*2*7

INTERNAL COMMENTS:

 Edit? NO//

Continue and Display Patch ? No// YES

DEVICE: HOME//

DHCP Completed/NotReleased Patch Display             Page: 1
================================================================
Run Date: DEC 20, 2011           Designation: XYZ*2*7
Package : XYZ                    Priority : MANDATORY
Version : 2                      Status : COMPLETE/NOT RELEASED
```

```
=============================================================

Associated patches: (v)XYZ*2*5   <<= must be installed BEFORE `XYZ*2*7'


Subject:  ENHANCEMENT TO COPAYMENT

Category:  ROUTINE                          ← Reference Appendix 2 for descriptions
           DATA DICTIONARY                  of Patch Categories.
           ENHANCEMENT

Description:
===========
  ……
Routine Information:
====================
……


=============================================================
User Information:
 Entered By  :   ANY,PERSON1      Date Entered  :   MAY 31,2011
  Completed By:   ANY,PERSON2     Date Completed:   DEC 19,2011
  Released By :                   Date Released :
=============================================================

Releasing Patch: XYZ*2*7
COMPLIANCE DATE: Jan 20, 2012//   (JAN 20, 2012)  ← Take the default unless the
Project Manager provides a different date.
STATUS OF PATCH: COMPLETED/UNVERIFIED//
   Choose from:
      c   COMPLETED/UNVERIFIED
      e   ENTERED IN ERROR
      u   UNDER DEVELOPMENT
      v   VERIFIED
      r   RETIRED
      x   X-Cancelled
STATUS OF PATCH: COMPLETED/UNVERIFIED// V  VERIFIED

Are you sure you want to change status to 'Released'? No// Yes
  ...status changed to 'Released'
  . .. .

NOTE: A bulletin has been sent to select users for this package
  informing them of this 'Released' patch.
```

4. Reference and Related Links

- VistA Primary Developer Review Checklist
- VistA Secondary Developer Review Checklist
- VistA SQA Checklist
- VistA Verifier Checklist

5. Follow Up Responsibility and Follow Up Date

VistA Office (VO) June 2023

A. Appendix 1: Create New Patch Entry

This appendix provides step-by-step instructions for creating a new entry in the National Patch Module. It includes displays of the prompts displayed when using the option ***Add a Patch [A1AE PHADD]*** within the ***Developer's Menu [A1AE DEVELOPER]*** on **FORUM**.

Step 1: Create Initial Required Information

After selecting the option ***Add a Patch*** within the ***Developer's Menu***, the software will prompt for key information required to create the patch. This includes ***Package***, ***Version***, ***DHCP Patches Number***, and ***Subject***.

- **Package:** When prompted "Select PACKAGE:, " enter the name of the VistA Package you are modifying. The software will only allow you to select from a predefined list of packages for which you have been granted privilege.
- **Version:** When prompted "VERSION :," the software defaults with the current version of the software. Unless you are creating the first patch for a new version of the product, press <Enter> to select the default.
- **DHCP Patches Number:** This is the Internal Entry Number (IEN) of the patch. Always press <Enter> at this prompt. Do NOT edit the number provided.
- **Patch Subject:** When prompted "DHCP PATCHES PATCH SUBJECT:, " enter the subject of the patch using all UPPERCASE characters. The subject should be as succinct as possible. Often this reflects a Project Name, a key function added, or specific problem corrected.

Example: Adding Initial Required Fields

```
Select Developer's Menu Option:  Add a Patch

Select PACKAGE: PHARMACY PRESCRIPTION PRACTICE    PPP   National

Select VERSION: 1// <Enter>   Date Verified: 04-01-95
Are you adding 'PPP*1*45' as a new DHCP PATCHES (the 16908TH)? No// Y  (Yes)
DHCP PATCHES NUMBER: 17860// <Enter>
DHCP PATCHES PATCH SUBJECT: THIS IS THE PATCH SUBJECT

Patch Added: PPP*1*45
```

Step 2: Enter Additional Attributes for the Patch

After entering the initial required fields, the stub for the new record is stored in the National Patch Module. The software will then prompt for additional information.

- **Patch Subject:** The software re-displays the Patch Subject field so that edits can be made if necessary.
- **Holding Date:** This field is not typically entered when the patch is created. When the patch is completed and ready for hand off for National Release, the HOLDING DATE is added to allow time for processing.

- **Priority:** This field determines whether the patch is a MANDATORY or EMERGENCY patch. EMERGENCY is only used if there is a reason to expedite the installation of the patch beyond the normal installation compliance date.
- **Category of Patch:** This field identifies the categories that apply to the patch. Refer to Appendix 2 for a detailed description of each category.

  Note: Include all categories that apply to the patch.

```
PATCH SUBJECT: THIS IS THE PATCH SUBJECT Replace <Enter>
HOLDING DATE: <Enter>
PRIORITY: M  MANDATORY

Select CATEGORY OF PATCH: ?
 You may enter a new CATEGORY OF PATCH, if you wish
  Choose from:
   d   DATA DICTIONARY
   i   INPUT TEMPLATE
   p   PRINT TEMPLATE
   r   ROUTINE
   s   SORT TEMPLATE
   o   OTHER
   db DATABASE
   e   ENHANCEMENT
   pp PATCH FOR A PATCH
   inf INFORMATIONAL

Select CATEGORY OF PATCH: R  (r   ROUTINE)
Select CATEGORY OF PATCH: E  (e ENHANCEMENT)
Select CATEGORY OF PATCH: <Enter>
```

Step 3: Add the Patch Description

The software will next prompt for the Patch Description.

- The software first asks whether you want to copy lines from a message into the description.

```
Do you want to copy lines from a message into the Patch Description? No//
<Enter>
```

- The software then prompts for the Patch Description. It is at this point where you would use one of the embedded Patch Templates in section 3.1 of this document.
  See Appendix 3 for complete descriptions of each section of the template.

```
PATCH DESCRIPTION:
```

Step 4: Add Remaining Fields to finalize the Patch Stub

- **Copy Packman Message:** The software will next ask whether you want to copy a packman message into the Message text and defaults to "No". This is used to link a Packman message that contains the KIDS build to be distributed via the National Patch Module on FORUM. If distributing via a HOST file, answer "NO". Refer to the National Patch Module Guide for more information regarding distribution alternatives.

```
Do you want to copy a packman message into the Message Text? No// <Enter>

MESSAGE TEXT: <Enter>
```

```
No routines included
Edit? NO// <Enter>
```

- **Routines:** The software will then prompt for Routine's that are included in the patch.

```
Select ROUTINE NAME: <Enter>

DISPLAY ROUTINE PATCH LIST: Yes// <Enter>
```

- **Internal Comments:** The software will then prompt for internal comments that are only seen by the Developer and Verifier.

```
editing comments only seen by releasers/developers

INTERNAL COMMENTS:
 No existing text
Edit? NO//  <Enter>
```

- **Patch Release Check:** Ensure all associated patch(es) are identified and entered in the Patch Release Check multiple. If there are patch dependencies, enter the required or associated patch(es) in this field. You should designate if the patch(es) entered in this field should be verified/installed prior to the installation of the current patch you are creating.

```
Select PATCH RELEASE CHECK: <Enter>
```

- **Status of patch:** The STATUS OF PATCH defaults to "UNDER DEVELOPMENT" and will remain in that status until the patch is ready for release.

```
STATUS OF PATCH: UNDER DEVELOPMENT//<Enter>
```

B. Appendix 2: Category of Patch Definitions

PATCH FOR A PATCH

This category is only used when a critical defect is identified after release of an enhancement or defect repair patch that requires the initial patch to be backed out or not installed until the new "Patch to a Patch" is released. If the original patch is replaced in its entirety, the new patch should not use this category.

DATA DICTIONARY

This category is used when a defect repair or enhancement patch contains changes to data dictionaries.

INPUT TEMPLATE

This category is used when a defect repair or enhancement patch adds new Input Templates or contains changes to existing Input Templates.

PRINT TEMPLATE

This category is used when a defect repair or enhancement patch adds new Print Templates or contains changes to existing Print Templates.

ROUTINE

This category is used when a defect repair or enhancement patch adds new routines or contains changes to existing routines.

SORT TEMPLATE

This category is used when a defect repair or enhancement patch adds new Sort Templates or contains changes to existing Sort Templates.

OTHER

This category is used to indicate that other components are included in the patch. This includes, but is not limited to, additions or changes to Forms, Functions, Dialog, Bulletins, Mail Groups, Help Frames, Options, Remote Procedures, Security Keys, Protocols, List Templates, HL7 components, Parameter Definitions and Templates, and XULM Lock Dictionaries.

DATABASE

This category is used when a defect repair or enhancement patch adds or modifies data in the database.

ENHANCEMENT

This category is used when the patch contains changes to software functions not related to defect repairs.

INFORMATIONAL

This category is only used when the patch provides information but does not require installation of software at the site. An example of this would be the release of an enterprise application build or the release of updated documentation being added to the VDL. An Informational Patch could also include instructions for updating or verifying data or other components manually by local technical staff.

C. Appendix 3: Status of Patch Definitions

COMPLETED/UNVERIFIED

Once the developer is ready to have the patch verified s/he can change the status of the patch to 'completed/unverified'. When this status is assigned to the patch a bulletin is sent to the verifier(s) of the package notifying this patch is ready for verification. A patch message is also sent to the verifiers at this time.

ENTERED IN ERROR

If an error is made, or a developer decides not to release an 'under development' or 'completed/unverified' patch it may be deleted, since only the developer and verifier have viewed them. If an error is made in a 'verified' patch the status should be changed to 'entered in error', and an error description should be entered to indicate why the patch is in error, and where the correct patch can be found.

UNDER DEVELOPMENT

A status of 'under development' is assigned to the patch via the option to 'Add a Patch'. It is assigned by the system when the patch is first added.

VERIFIED

When the verifier is ready to release the patch to the field, s/he can change the status of the patch to 'verified'.

RETIRED

Certain patches may also be 'retired', specifically the database or DBA type patches may be retired after a specific version. Additional comments about the retired patch can be put in the retirement comment field.

X-Cancelled

Patches that have been started, and have had substantial work, but ultimately never released are updated to this status. All information in the patch remains intact. The status can be changed back to UNDER DEVELOPMENT at any time if access to the information in this canceled patch is needed.

D. Appendix 4: Required Build Examples

If a patch requires another patch to be installed prior than this patch to be able to function, that patch should be listed as a required build in the KIDS build and as an associated patch in the National Patch Module.

Example 1: The patch has one or more required builds from the same package as a result of previously released patches including the same routines.

Patch XYZ*1*100 contains both ^XYZ123 and ^XYZMNT. Patch XYZ*1*49 contained the latest changes to the routine ^XYZ123 and patch XYZ*1*56 contained changes to the routine ^XYZMNT. It would require both patches 49 and 56 regardless of when the previous patches were released (unless the previous patch was not created with KIDS). If patch XYZ*1*56 required patch 49, you would only need to list patch 56 as a required for patch XYZ*1*100.

Example 2: The patch requires another patch from another package to function properly. The other package's patch is either not released yet or released but has not passed the compliance date for installation.

Patch XYZ*1*100 requires changes in DG*5.3*342 to function properly. DG*5.3*342 would be included as a REQUIRED BUILD in KIDS and ASSOCIATED PATCH in the National Patch Module for patch XYZ*1*100.

Example 3: Two patches from different packages must be installed together for both to function properly.

If two patches have to be installed together, you should consider creating them in a combined build. For example, XYZ*1*100 and DG*5.3*342 are mutually dependent upon each other.

NOTE: Combined builds should be used judiciously. Combining multiple patches that depend on one specific patch but not on each other may create risks in testing and deployment. When doing a combined build, consultation with the Verifier is required. If uncertain, consultation with a package expert is warranted.

E. Appendix 5: Establish Connection between Development and FORUM Accounts.

To move the KIDS build to the patch entry on the NPM you should use the following steps.

a. Create KIDS Build.

1. Ensure all required builds are listed. Additional information regarding Required Builds can be found Appendix 4.
2. Track Package nationally field should be set to "YES."
3. Alpha/Beta Testing field should be set to "NO" prior to completing the patch.
4. Package File Link field must be defined.
5. Description must be entered.

b. Store in MailMan Message or Host File for Transport.

1. Subject title should be a combination of patch number and description (e.g., XYZ*2*12 Test Version 1).
2. Send MailMan message to XXX@Q-PATCH at FORUM.

NOTE: If distribution to be created is a Host File, then the VistA development account will need to be setup so that the development MailMan domain and package(s) are linked in FORUM.

- This will need to be done **prior** to creating the Host File distribution.
- There is no easy way to determine if this setup has been previously completed until you try to Edit the Patch in FORUM to copy in the routines and checksums from the Host File. (See below.)
- If the linkage needs to be created, then submit a new incident request to the Enterprise Service Desk (ESD) ServiceNow requesting the MailMan Domain to Package setup in FORUM. See Appendix 5 for details.
- If the link from the development MailMan domain to the package in FORUM has been setup, then when a Host File is created from the development account with that domain and package, it will automatically send a message to FORUM with the patch contents, including routines and checksums, for selection.
- Once package to MailMan domain associations have been established on FORUM, they will remain unless specifically requested to be removed. (uncommon)

c. **Mailman:** To send the test patch to the test sites you must link the MailMan message containing the build to the patch. Choose the Edit a Patch [A1AE PHEDIT] option on the Developer's Menu [A1AE DEVELOPER] under the Patch User Menu [A1AE USER] on FORUM to edit the patch. At the prompt Patch Status, you must enter "U" for "Under

Development." Accepting the default will not provide the following prompts. See example below.

```
PATCH DESCRIPTION:

347> Install Message sent #12345678
EDIT Option: <RET>
Select ROUTINE NAME: <RET>
       Editing MESSAGE TEXT
Do you want to copy a packman message into the Message Text? No//  Yes
          (1) XYZ*2*12 Test Version 1  <DOE.JANE@DEVXYZ.  1438
          (2) XYZ*2*12 Test Version 2  <DOE.JANE@DEVXYZ.  1722

Select Message to copy: 2// <RET>
  Using message 'XYZ*2*12 Test Version 2' Checking the input ..
  Deleting old text..
         Merge KIDS message into patch message text
MESSAGE TEXT: . . .
1361>
1362>$END KID XYZ*2.0*12
EDIT Option: <RET>
Editing comments only seen by releasers/developers
INTERNAL COMMENTS: <RET>
  1> <RET>

Select PATCH RELEASE CHECK: <RET>

STATUS OF PATCH: UNDER DEVELOPMENT// U UNDER DEVELOPMENT

Option to create a Patch message to send to test sites.
TEST v2   will be added to the Patch message subject.

You may change the TEST v[#] if necessary.:  (1-99): 2//  <RET>
<you are prompted for recipients, message is created and sent>
```

d. **Host:** To send the test patch to the test sites you must link the Host File message that was automatically sent to FORUM from the development account to the patch to capture the routine name and corresponding checksums.

Choose the Edit a Patch [A1AE PHEDIT] option on the Developer's Menu [A1AE DEVELOPER] under the Patch User Menu [A1AE USER] on FORUM to edit the patch. At the prompt Patch Status, you must enter "U" for "Under Development". Accepting the default at the Patch Status will not provide the following prompts. See Example below.

NOTE: This assumes that the MailMan Domain to Package setup mentioned previously has been completed in FORUM prior to creation of the Host File distribution

```
PATCH DESCRIPTION:

347> Install Message sent #12345678
EDIT Option: <RET>                  Editing MESSAGE TEXT

Do you want to copy a packman message into the Message Text? No// <No>
Want to copy HFS cache,   From: CDMS.FO-BIRM.MED.VA.GOV
     On: Oct 19, 2016@14:27:43

Do you want to copy Routine Names and Checksums from the HFS cache file. No//
<Yes>
Routine Name's and Checksums moved.
```

```
 MESSAGE TEXT: <RET>
  1>  <RET>
DISPLAY ROUTINE PATCH LIST: Yes// <RET>
editing comments only seen by releasers/developers
INTERNAL COMMENTS: <RET>
   1> <RET>
Select PATCH RELEASE CHECK: <RET>

STATUS OF PATCH: UNDER DEVELOPMENT// <U> UNDER DEVELOPMENT

Option to create a Patch message to send to test sites.
TEST v2   will be added to the Patch message subject.

You may change the TEST v[#] if necessary.:  (1-99): 2//  <RET>
<you are prompted for recipients, message is created and sent>
```

F. Appendix 6: Mailman Domain to Package Setup

When entering a ServiceNow requesting the MailMan domain to package setup on FORUM, the following should be entered as follows:

- **Group:** NTL SUP Health Systems Platform
- **Summary:** MailMan Domain and Package Setup in FORUM for <*AccountName*>
- **Description:**

  Example:

  The <*xxx*> team is requesting the following VistA development package/domain association(s) to be setup in FORUM for capturing the routine(s)/checksum(s) in the patch.

  <*IB (Integrated Billing)*> package with the <*DEVCRN.FO-ALBANY.MED.VA.GOV*> domain

NOTE: Include all packages in the Incident request description that should be associated with the MailMan domain in FORUM.

- How to determine the MailMan Domain in the VistA development account using VA FileMan (FM):

```
VA FileMan 22.0

Select OPTION: <INQUIRE TO FILE ENTRIES>

OUTPUT FROM WHAT FILE: MAILMAN SITE PARAMETERS// <RET>
Select MAILMAN SITE PARAMETERS DOMAIN NAME: 1   DEVCRN.FO-ALBANY.MED.VA.GOV

ANOTHER ONE:  <RET>
STANDARD CAPTIONED OUTPUT? Yes// <N>  (No)
FIRST PRINT FIELD: <DOMAIN NAME>
THEN PRINT FIELD:
Heading (S/C): MAILMAN SITE PARAMETERS LIST  Replace
DEVICE:   VIRTUAL TELNET   Right Margin: 80// <RET>
MAILMAN SITE PARAMETERS LIST   DEC  5,2016  11:22   PAGE 1
DOMAIN NAME
------------------------------------------------------------------------------

DEVCRN.FO-ALBANY.MED.VA.GOV
```