

V I S T A

**OSEHRA Patch Module 2.5**  
**User Manual: Patch Verifiers**  
June 2015

prepared for



by



©Copyright 2015 by VISTA Expertise Network. Licensed under Creative Commons Attribution-ShareAlike 4.0 International. Details are available at <http://creativecommons.org/licenses/by-sa/4.0/>

## Revision History

Date	Description	Language	Authors
June 2015	OSEHRA Version 2.5 release	English (US)	Kathy Ice and Frederick D.S. Marshall

# Contents

Orientation.....	1
Introduction.....	2
Chapter 1: The Verification Process.....	3
Workflow Management.....	4
Chapter 2: Verification Tools.....	6
Completed Patch Summary for Assigned Packages.....	6
Display a Completed/Not Released Patch.....	7
Forward a Completed/Not Released Patch Message.....	8
Release a Patch (and/or Edit Comments).....	9
Routines that Overlap in Patches.....	9
Completed Patch Summary Report by Date.....	10
Package Release (and/or Edit Comments).....	10
Compliance Date Tools.....	11
Export Patch(es) to the Filesystem.....	13
Appendix A: The Lineage of VistA.....	14
Glossary.....	15

# Orientation

This manual is one of four user manuals for version 2.5 of the OSEHRA Patch Module. There is a different user manual for each essential role associated with the OSEHRA Patch Module; this manual is for patch verifiers.

If you are not sure whether or not you are a patch verifier, you probably are not. Patch verifier is a very specific role, and somebody would have mentioned it to you.

If you will be evaluating patches that have been released, to determine whether those patches will be installed at your organization, you are performing a step that is part of secondary development. Please consult the User Manual for Secondary Developers.

The OSEHRA Patch Module documentation suite:

- Release Notes
- Value Proposition
- Installation Guide
- User Manual: Primary Developers
- User Manual: Secondary Developers
- User Manual: Verifiers
- User Manual: Patch Subscribers
- Technical Manual
- Security and Privacy Manual

# Introduction

The OSEHRA Patch Module, as the name implies, is a software package that allows users and developers to create, revise, distribute, review, and receive software patches and updates for VistA. Options are provided for systematic entry, revision, and review of patches by developers, review and release of patches by verifiers, and display and distribution of the released patches to the users.

But before we get to talking about all the things we can do with patches, it's probably best to take a moment and make sure we're all on the same page about what a patch is, and what it does. Patches began as a way of fixing problems in an active VistA system, and many patches released today are simple bug fixes. However, most patches include more than fixes. Developers found that patches were the easiest way to add the enhancements and new features that their users wanted without taking the system offline or releasing a new version. Today, patches are the primary method for making updates and improvements to VistA.

A patch, then, can include bug fixes, upgrades, enhancements, new features, or all of the above. Its main feature is that it can be installed on an active, running VistA system with minimal disruption.

Patches are created in KIDS (the Kernel Installation and Distribution System), but are packaged and distributed via the OSEHRA Patch Module. Until relatively recently, only the developers at the Department of Veterans Affairs (VA) had access to the Patch Module. With this release, the OSEHRA Patch Module becomes more widely available, and more developers have the opportunity to create and release patches.

# Chapter 1: The Verification Process

Patch verification is the last step in the process of creating and releasing a patch. Once the verifier is satisfied that the patch meets all standards, he or she completes the verification, which triggers the release of the patch. So, why isn't this person called the patch releaser or the patch distributor? Well, sometimes they are—like in the menu system. However, although releasing the patch is probably the more exciting task, verifying the patch is far more important. In this chapter, we review the steps that most verifiers take when reviewing a patch for verification.

To begin the patch verification process, you will first need to have copies of the standards. These are:

- the Verification Standard,
- the Programming Standards and Conventions (the SAC), and
- the MUMPS Standard.

The last of these, the MUMPS Standard, is the same for everyone. However, different organizations have different Verification Standards and SACs. If you are uncertain which one you should be using, contact your supervisor or QA manager.

When you receive a patch for verification, your first step is to install it into a known baseline VISTA environment. Different verifiers have different methods they prefer to use for this. Some like to create a new, empty VISTA environment whenever they need to install a new patch. Others prefer to run a script that can set up such an environment for them. If you are interested in using a script but don't have one, contact your system manager for assistance.

Install the patch using the installation instructions provided. Make a note of anywhere the instructions are confusing or ambiguous, and of course

make a note anytime the instructions don't actually work to install the software.

Next, look at the patch description. Much of it will be the familiar patch description boilerplate, but some of it will have been written by the developers for this specific patch. Create a list of things in the description that could be inaccurate or confusing.

Once the patch is installed, review the code. Compare the software in the patch with the software in your baseline environment to ensure that the changes made are the changes described in the description. Run the software to make sure it works the way it is supposed to.

Finally, apply your standards (Verification Standard, SAC, and MUMPS Standard) to the code to ensure that it all conforms.

When your review is complete, create a Report of Findings. List everything you reviewed, and make a note of anything that is inaccurate, confusing, or not up to standards. Distribute your report to the developers, and to any appropriate managers or supervisors. If the patch is ready for release, use the Release a Patch option to change its status and distribute the patch.

## ***Workflow Management***

The patch verifier is the last step in the patch creation process. Therefore, if the team is working to a deadline, the patch verifier is often in the "make or break" position. This is not a comfortable position to be in, but there are things you can do as a verifier to help manage your workflow and prevent bottlenecks.

The first and most important thing you can do is communicate with the developers. Have some idea how many patches are currently in development, and when they are expected to be completed. This can help you spot potential problems before they develop. If you realize that you're going to get 5 patches to verify at the end of the month, it's probably time

to talk to those developers, explain the situation, and see whose deadlines can be pushed back.

In addition, take advantage of the tools and reports available to you in the OSEHRA Patch Module. They can help you manage your workflow and spot potential problem areas. These tools and reports are described in the next chapters.



## Chapter 2: Verification Tools

Most of the tools for verifying primary development can be found on the Patch Releaser menu, accessible to anyone with the A1AE PHVER key:

```
Select Releaser Menu Option:
  CDE      Compliance Date Edit
           Completed Patch Summary for Assigned Packages
           Completed Patch Summary Report by Date
           Completed/NotReleased Patches Report
           Compliance Date Display
           Compliance Date Print
           Display a Completed/NotReleased Patch
           Export Patch(es) to the Filesystem
           Forward a Completed/NotReleased Patch Message
           Package Release (and/or edit internal comments)
           Release a Patch (and/or edit internal comments)
           Routines that overlap in patches
```

In the menu, the options appear in alphabetical order (Compliance Data Edit sorts to the top because it has an abbreviation, CDE). However, alphabetical order is probably not the order in which you will need these options. Below, we present the options you will most likely use as part of your normal workflow, followed by the less-used options.

### ***Completed Patch Summary for Assigned Packages***

Use this option to see a snapshot of all the patches that area ready for you to verify.

```
Select Releaser Menu Option:  Completed Patch Summary for
Assigned Packages

Select a Stream: FOIA VISTA
DEVICE: HOME//
```

When you choose this option, you are prompted for which patch stream you want to see, and then asked about a device. You can see the report on your screen (“home”), or send it to another device such as a printer.

## ***Display a Completed/Not Released Patch***

Once you’ve taken a look at your snapshot (the Completed Patch Summary), your next step is often to look at one patch in detail—because you want to verify it, or because you just want to look at it to see what verification would be like.

When you select this option, you are first prompted for the patch to be displayed. You can look up a patch by patch designation, package, patch subject, or routine.

```
Select Releaser Menu Option: Display a Completed/NotReleased
Patch

Select PATCH: ZZJI*1.0*9          [OV]KBAP TEST PATCH ONE
COM LGC

DEVICE: HOME// 0;1 80;9999
DHCP Completed/NotReleased Patch Display
Page: 1
=====
Run Date: MAY 12,2015    Stream: OV Designation: ZZJI*1*9
Package : ZZJLITESTS      Priority   : MANDATORY
Version : 1               Status: Completed/NotReleased
=====

Source: OSEHRA VISTA

Subject:  KBAP TEST PATCH ONE

Category:  ROUTINE

Description:
=====

$TXT Created by CARLSON,LARRY G at FORUM.OSEHRA.ORG  (KIDS) on
Tuesday, 05/12/15 at 16:49
```

```

This is the text written into the description when the patch was
being transported to a Packman message

Routine Information:
=====

The second line of each of these routines now looks like:
;;1.0;ZZJI;**[Patch List]**;APR 17, 2015;Build 1

The checksums below are new checksums, and
can be checked with CHECK1^XTSUMBLD.

Routine Name: KBAPKID1
  Before:      B8721   After:      B8721
Routine Name: KBAPKID2
  Before:      n/a     After:      B8730

=====
User Information:
  Entered By   : CARLSON,LARRY G   Date Entered   : MAY 12,2015
  Completed By: IVEY,JOEL          Date Completed: MAY 12,2015
  Released By  :                   Date Released  :

Reviewed By   :                               Date Reviewed  :
=====

Select PATCH:
    
```

Once you have selected a patch, you are then prompted for a device. The resulting display contains the patch subject, description, category, priority, and status, as well as information about the users who entered and completed the patch and a routine information section showing coding changes. You can then select another patch to view, or exit the option.

## ***Forward a Completed/Not Released Patch Message***

Most verifiers use this option to forward a copy of the patch to themselves in order to install it in their baseline environment.

```
Select Releasers Menu Option: Forward a Completed/NotReleased  
Patch Message  
Select PATCH to forward: ZZJI*1.0*9      [OV]KBAP TEST PATCH ONE  
COM LGC  
Send mail to: CARLSON,LARRY G//          CARLSON,LARRY G  
Select basket to send to: IN//  
And Send to:  
                Message 34156 has been forwarded.
```

After selecting this option, you are prompted for the patch to be forwarded. You can then select one or more recipients.

## ***Release a Patch (and/or Edit Comments)***

Patch verifiers can use this option to release a patch once it has been verified. You can also use this option to edit internal comments, without releasing the patch.

After selecting the patch to be released, you are first given the opportunity to edit internal comments, using a word-processing tool. Once the internal comments are complete, the OSEHRA Patch Module asks if you want to continue.

If all you wanted to do was edit the internal comments, answer "No" to this prompt, and you will exit to the menu. To proceed with verification and release, answer "Yes" to this prompt.

When you change the status of a patch to "verified", a bulletin is sent to user(s) who have requested notification of patches for the package. The patch is then available to all users on that particular patch stream.

## ***Routines that Overlap in Patches***

Use this option to get a list of routines that are affected by more than one patch. Although this isn't something you'll necessarily use for an individual patch, it is a useful tool for overall workflow management. If,

for example, you can spot a group of two or three patches that have routines in common, you might want to plan to verify them in a group, or at least consecutively, so you have a good grasp of what is changing and how.

```
Select Releaser Menu Option:  Routines that overlap in patches

Select DHCP PATCHES PATCH DESIGNATION: ZZJI*
  1  ZZJI*1*10      [OV]KBAP TEST KIDS 2      UND LGC
  2  ZZJI*1*6       [OV]TEST A                UND  JI
  3  ZZJI*1*        [OV]ONE ROUTINE PATCH TEST  COM unk
  4  ZZJI*1*9       [OV]KBAP TEST PATCH ONE    COM LGC

CHOOSE 1-4: 1  ZZJI*1*10      [OV]KBAP TEST KIDS 2      UND LGC

Include patches released within how many months? (0 for only
active overlaps):
(0-999): 0//

Routines in entered or completed patches that overlap with
ZZJI*1*10
Routine      Patch      Release Date      Patch Entered by
KBAPKID1     ZZJI*1*9      CARLSON,LARRY G
KBAPKID2     ZZJI*1*9      CARLSON,LARRY G
```

Once you select this option, you are asked to select a specific patch. You can then view a report of any patches that have changes in overlapping routines, whether those patches have been released or not.

## ***Completed Patch Summary Report by Date***

Although you will use your report of patches from assigned packages more often, you may from time to time want to see this report, listing all patches ready for verification, not just the ones assigned to you.

## ***Package Release (and/or Edit Comments)***

Most of the verification work you do will involve patches. Once in a great while, however, the developers in your organization may release a new

version of the package. Use this option to release the new version once it has been verified.

This option works exactly like the Release a Patch option, including the ability to enter or edit comments without actually releasing the package. See the Release a Patch option (above) for more details.

## ***Compliance Date Tools***

The compliance date—the deadline for the patch to be installed—is not used much outside VA. Most organizations simply install patches at their earliest convenience, without using a specific deadline. Within VA, however, the compliance date is still an important piece of information about a patch. As a verifier, you have three tools to help you manage compliance dates.

### **Compliance Date Display**

This option allows you to view a standard report of compliance dates for selected patches:

```
Select Releaser Menu Option: COMPLIANCE DATE DISPLAY
Start with PATCH STREAM: FOIA VISTA//
Go to PATCH STREAM: FOIA VISTA//
  Start with DATE PATCH VERIFIED: JUN 1,2010//    (JUN 01, 2010)
  Go to DATE PATCH VERIFIED: LAST//
DEVICE:    HOME
```

When you invoke the option, you can choose which patch stream(s) you want to see, and further narrow your selection by date. Once you have selected a range of patches, you will see a standard report like this one:

PATCH COMPL/COMMENT RPT	SEP 12,2014@17:16	PAGE 1
COMPLIANCE		
DATE	CATEGORY	PACKAGE
REASON FOR CHANGE		
-----		

DATE PATCH VERIFIED: JAN 6,2011			
ICPT*6*54	MANDATORY	INFORMATIONAL	CPT/HCPCS CODES
ICD*18*52	MANDATORY	INFORMATIONAL	DRG GROUPER
LEX*2*74	MANDATORY	ROUTINE ENHANCEMENT DATA DICTIONARY OTHER	LEXICON UTILITY

## Compliance Date Print

This option allows you to see the compliance date for one or more specific patches you select:

Select Releaser Menu Option: COMPLIANCE DATE PRINT			
Select DHCP PATCHES PATCH DESIGNATION: DI*22*10165 derived from [FV]DI*22*165			
DEVICE: HOME			
PATCH COMPL/COMMENT RPT			SEP
12,2014@17:16 PAGE 1			
		COMPLIANCE DATE	PAC
KAGE		CATEGORY	
REASON FOR CHANGE			
-----			
DI*22*10165 MANDATORY		ROUTINE	
		OTHER	VA
FILEMAN			
Select DHCP PATCHES PATCH DESIGNATION:			

## **Compliance Date Edit**

Use this option to change the compliance date of any of the patches you are assigned to verify. Making changes to a patch using the Edit a Patch option would normally change the patch's status from "Completed/Unverified" to "Under Development," which you usually do not want. However, changing the compliance date using the Compliance Date Edit option will not change the patch's status.

## ***Export Patch(es) to the Filesystem***

Use this option to turn one or more patches into a flat file that can be placed on a server and accessed via FTP. This is the preferred method of obtaining patches for sites that do not use Mailman/Packman.

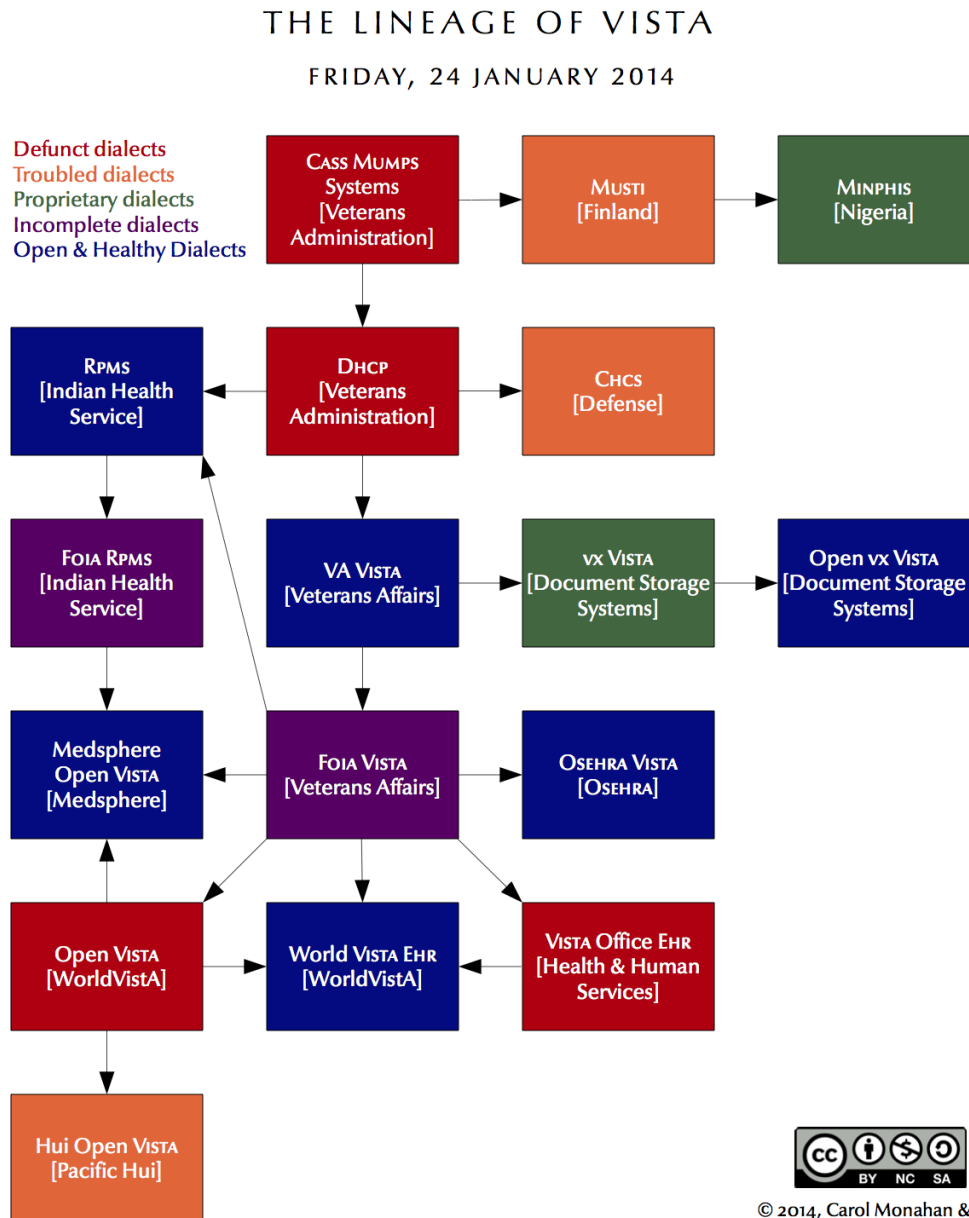
You will need to know the Patch ID(s) of the patch(es) you want to use, and the full path for the export destination.

The OSEHRA Patch Module will allow you to select any patch, even patches still under development, so you can export patches for use by testers. Use caution when selecting and exporting patches; unverified patches should not be released to anyone other than testers.

Note that if you select more than one patch to export, they will automatically be combined into a multi-build. If you have multiple patches and do not want a multi-build, you will need to run this option more than once.



# Appendix A: The Lineage of VistA



VISTA Expertise Network's *The Lineage of VistA* is licensed under the *Creative Commons Attribution-Noncommercial-Share Alike 3.0 United States License* (<http://creativecommons.org/licenses/by-nc-sa/3.0/us/>).

© 2014, Carol Monahan & Frederick D. S. Marshall

# Glossary

Application	An administrative division of VistA that automates part or all of one hospital or clinical service. Pharmacy and Nursing are examples of applications.
Application Version	A complete new release of an application. Versions are numbered sequentially.
Build	See KIDS Build
Checksum	A number unique to any given version of a software element. Even a small change to the software will change the checksum, so checksums are used to detect changes and verify a particular version.
Dialect	See VistA Dialect
Distribution	See KIDS Distribution
FOIA	Freedom of Information Act. The term “FOIA” can refer to the Act itself, or to a request sent to the government under the auspices of the Act.
Forum	A VistA system used as the hub of an organization’s VistA software lifecycle. VA and OSEHRA each have their own Forums.
Gerrit	A code-review system for use with a repository such as Github.
Git	A version-control system.

Github	A platform that hosts repositories using the Git system.
Host-File Format	A file-based format for a KIDS distribution. It consists of two parts: the KIDS file and the text file.
KIDS	The Kernel Installation and Distribution System. KIDS is the primary method for preparing a patch for the OSEHRA Patch Module, as well as the mechanism for installing patches.
KIDS Build	The “manifest” of a KIDS distribution, which lists all the components included in the distribution.
KIDS Distribution	A host file or Packman message containing a software update and associated tools and conversions for applying it.
KIDS File	In a host-file format, the portion of the file containing the software.
KIDS Install	A record describing what happened during each installation of a KIDS distribution at a specific site.
Local Modification	A change to VistA made for a specific facility or organization. Local modifications are necessary in VistA, but result in changes to checksums that make version control more challenging.
Mailman	VistA’s native email system. Patches can be distributed using Mailman’s Packman module.
Namespace	A convention for naming VistA package elements. Each developer or organization is assigned a

	namespace, which is a unique character string, to be used use in naming routines, options, and other package elements. Namespacing helps keep similar elements from different developers distinct and easily identifiable.
Numberspace	Similar to a namespace, a numberspace is a unique numeric string assigned to a developer or organization. Numberspaces are used for VistA elements that have numbers rather than names.
Package	A distribution of a new version of an application.
Packman	A module of Mailman used to ship patches and other software.
Packman Format	A format for a KIDS distribution designed for use with Packman.
Packman Message	Any email message that contains a KIDS distribution in Packman format.
Patch	Any small change or update intended for installation in an active VistA system. Most patches can be installed with minimal disruption to the system or its users.
Patch Completer	The developer who reviews the patch developer's work, then updates the status of the patch in the OSEHRA Patch Module to "completed."
Patch Developer	Person who initially entered the information on the patch into the OSEHRA Patch Module. That person will be listed as the "developer" in the OSEHRA Patch Module, whether they did any actual development work or not.

Patch ID	A multi-part identification number for a patch, which includes the application namespace, the application version number, and the patch number.
Patch Message	An email message that contains a patch description and a Packman-format KIDS distribution. This is the default method for the OSEHRA Patch Module to distribute patches.
Patch Number	Unique number given to a patch, as it relates to the specific application and version. Patch numbers are numberspaced, so patches from different sources can be immediately distinguished.
Patch Reviewer	In secondary development, the developer who reviews the released patch to determine what kind of secondary development might be needed.
Patch Stream	The series of patches developed and released for a specific application or dialect.
Patch Subscriber	A person or organization who has signed up to receive a particular patch stream.
Patch Verifier	Specialist who confirms that the patch is functionally complete, and meets all standards. Verifiers make the decision to release the patch.
Primary Developer	The person or team who initiates the patching process and releases a new patch.
Primary Development	The actions involved in creating and distributing a new patch.

Repository	Online electronic storage which houses reference versions of a specific software. Generally, one version is designated as the “official” version. OSEHRA provides repositories for OSEHRA VistA and FOIA VistA.
Required Patch	A prerequisite patch. All patches should list their required patches—that is, their prerequisites—for installation.
Secondary Developer	The person or team who re-purposes a released patch for their VistA dialect.
Secondary Development	The actions involved in re-purposing a patch for a different VistA dialect.
Sequence Number	Unique number assigned when a patch is verified. It determines the default order in which patches should be installed.
Text File	In host-file format, the portion of the file that contains the patch description.
Version	See Application Version
Version Control	A system or methodology for ensuring that all software within a given organization is the same version.
Version Number	The sequential number of the current application version. Each VistA application has its own version number. For example, the current version of Laboratory is 5.2, while the current version of Fileman is 22.2.
VistA Dialect	A unique, stable version of VistA supported by a

	specific vendor. Popular VistA dialects include OSEHRA VistA, vxVistA (Document Storage Systems), Medsphere Open VistA and WorldVistA EHR.
VistA Service Pack	A bundle of VistA packages and patches, which can be used to upgrade an existing VistA system.
VistA Snapshot	A copy of an existing VistA system. A VistA snapshot is most commonly used to clone a new VistA instance.