**AWARE (Alert Watch And Response Engine)**

VHA Innovation project 20082

The Alert Watch and Response Engine (AWARE) is an automated tool that enables clinicians, supervisors and administrators to monitor and track responses to CPRS critical alerts, with an initial emphasis on cancer-related alerts. It consists of three basic functions: (1) an alert monitor (2) an alert tracker for follow-up actions and (3) a viewing/reporting engine.

AWARE will:

* Track critical alerts and monitor follow-up action.
* Identify critical lab and imaging test result alerts that lack timely follow-up.

**AWARE Application Components**

At present, there is little to no special notification of normal, abnormal or incomplete test results, such as those indicative of a cancer. Critical alerts occur in the system, but very little tracking, follow-up information, special notification or decision support occurs, other than what is present in VA’s base system.

From a CPRS users’ perspective, the AWARE system:

* Provides an AWARE prompt to notify the user that an unacknowledged, abnormal result alert type exists for a patient that has not had follow-up.
* Allows for a decision-support workflow that guides the user towards appropriate follow-up actions desired for that alert through the use of regular Reminder dialogs.

AWARE can identify a list of certain critical lab and imaging test result alerts that lack timely follow-up in a web-based Alert Viewer, which is accessible as a standalone application or via a call-up though the Tools menu in CPRS.

AWARE also consists of a web-based Knowledge Base (KB) editor to define critical alert types and categories and a web-based Quality Indicating (QI) reporting tool.

**AWARE Technical Approaches**

* **CPRS with AWARE Extension** – Consists of AWARE Follow-up actions during a Patient Close-out event in a modified CPRS instance using an AWARE COM Object (dll) extension. AWARE CPRS modifications are made in Delphi with MUMPS logic used for follow-up tracking via RPCs.
* **Alert Viewer** – Consists of an Intersystems CSP web-based application to view a VistA Alert Cache file, which holds alerts and their follow-up actions gathered via an alert processing background task. MUMPS logic with Java Script for viewing is utilized.
* **SQL Transporter** – Consists of a C# console application on a Windows server used to copy a VistA Alert Cache file data to an MSSQL database via an RPC.
* **QI Manager** – Consists of ASPX applications on a Windows server to view SSRS AWARE designed reports having specific access privileges.
* **Knowledge Based Editor –** Consists of an Intersystems CSP web-based application utilizingJava script and MUMPS logic.