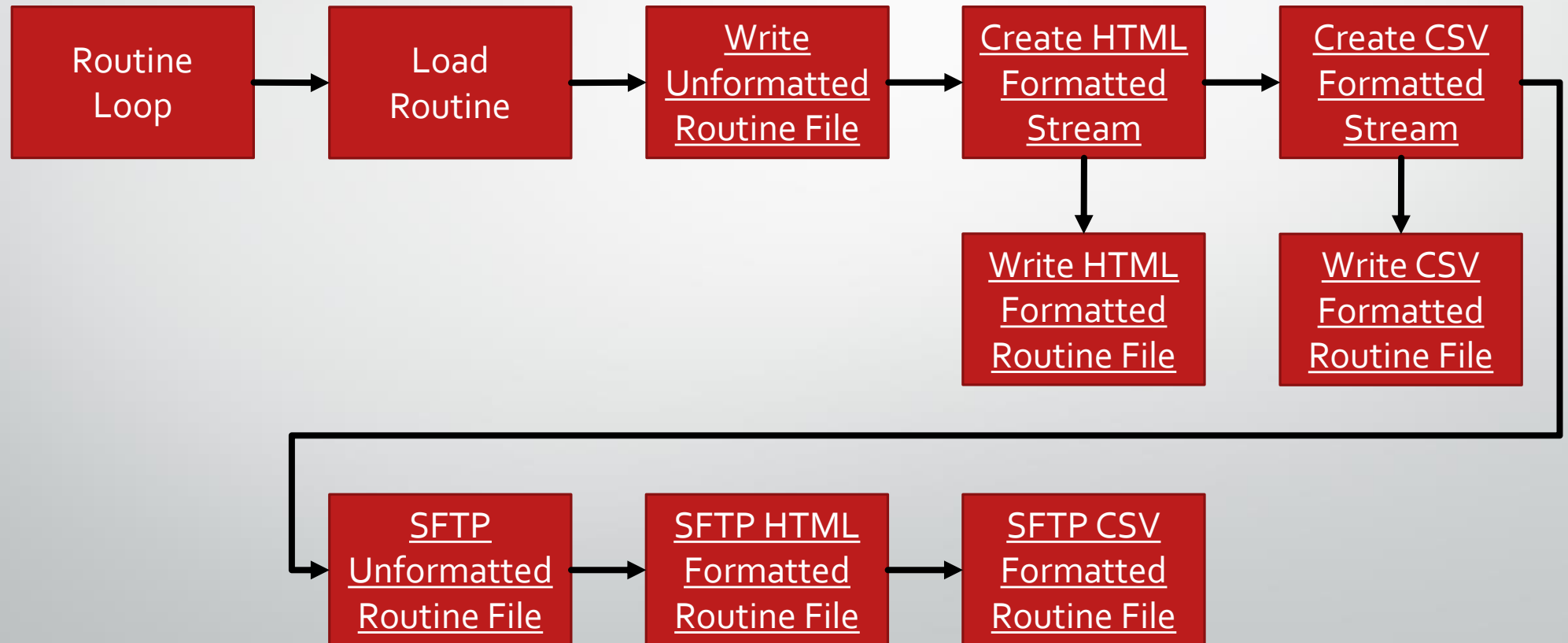


Diagramming VistA Routines

Ben Irwin
Software Engineer

Process Overview - Cache



Routine Loop

```
// Loop through all the routines starting with "A"
For asc=65:1:90
{
Set Rname=$C(asc)
Set Rname=$Order(^[UCI]ROUTINE(Rname),-1)
For
{
Set Rname=$Order(^[UCI]ROUTINE(Rname))
Quit:$Extract(Rname,1)'=$C(asc)
Quit:Rname=""
Continue:Rname["."]
Write Rname,!
...
}
```

Load Routine

```
// Load the individual routine into the instr class
Set instr=##class(%GlobalCharacterStream).%New()
For LineNumber=1:1:[UCI]ROUTINE(Rname,0,0)
{
  Do instr.WriteLine(^[UCI]ROUTINE(Rname,0,LineNumber))
}
```

Write Unformatted Routine File

```
// Write the ROUTINE unformatted to a local file
Do instr.Rewind()
Set RnameT="C:\SftpFiles\"_Rname_"_T.php"
Open RnameT:("NRW"):5
While 'instr.AtEnd
  { Use RnameT Write instr.ReadLine(),! }
Close RnameT
```

Create & Write HTML Formatted File

```
// Create the HTML color formatting for each individual routine
Set outstrR=##class(%GlobalCharacterStream).%New()
Set ret=colorer.Color(instr,outstrR,"COS","p",,,,lang,.coloringerrors)
//
// Write the HTML color formatting to a local file
Set (rcount,tcount)=0
Do outstrR.Rewind()
Set RnameR="C:\SftpFiles\"_Rname_"_R.php"
Open RnameR:("NRW"):5
While 'outstrR.AtEnd
{
  Set LineOut=""
  Set rcount=rcount+1
  If $Extract(^[UCI]ROUTINE(Rname,0,rcount),1,1)'=" " { Set tcount=0 }
  Set LineOut=LineOut_ $Extract(" ",1,4-$Length(rcount))_ "<a id=""Line"_"_rcount_"_"">"_rcount_"_</a>"
  Set LineOut=LineOut_ $Extract(" ",1,4-$Length(tcount))_ "<a>"_tcount_"_</a>"
  Set LineOut=LineOut_" "_outstrR.ReadLine()
  Set tcount=tcount+1
  Use RnameR Write LineOut,!
}
Close RnameR
Kill outstrR
```

Create & Write CSV Formatted File

```
// Create the CSV formatting for each individual routine
Set outstrD=##class(%GlobalCharacterStream).%New()
Set ret=colorer.Color(instr,outstrD,"COS","C",,,,lang,.coloringerrors)
//
// Write the CSV format to a local file
Do outstrD.Rewind()
Set Rnamed="C:\SftpFiles\"_Rname_"_D.php"
Open Rnamed:("NRW"):5
While 'outstrD.AtEnd
{
    Use Rnamed Write outstrD.ReadLine(),!
}
Close Rnamed
Kill outstrD,instr
```

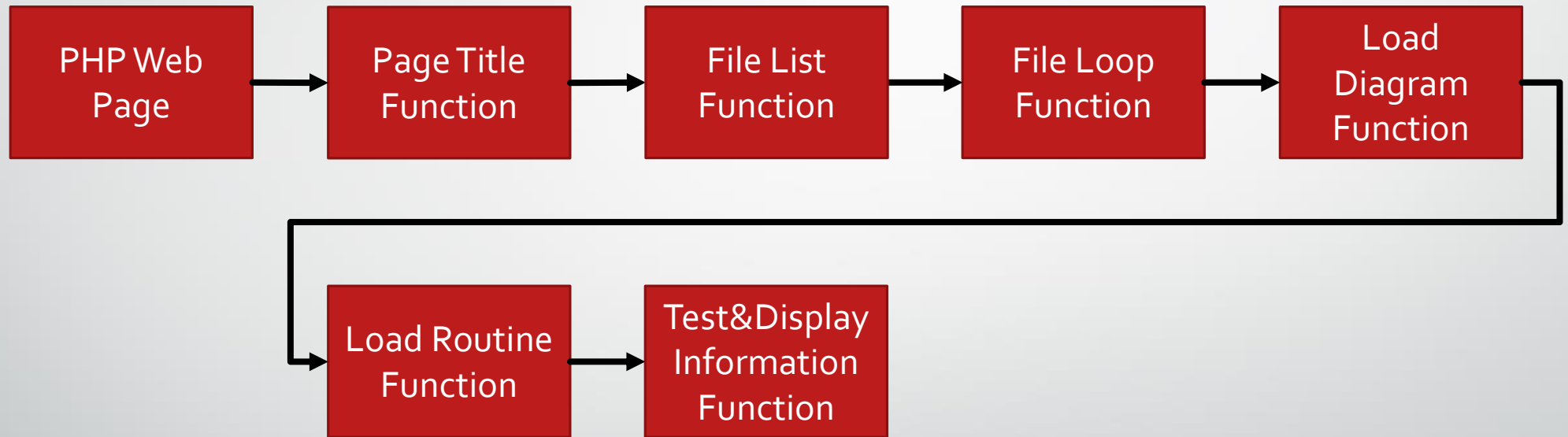
SFTP Files to Host

```
// Send Files to Host, Delete Files on the local machine.  
Do SftpAuto (RnameR, "/usr/home/biskate/public_html/bmirwin.com/vnode/routineR/"_ $C(asc) _ "/"_ Rname _ "_R.php", 1)  
Do SftpAuto (RnameD, "/usr/home/biskate/public_html/bmirwin.com/vnode/routineD/"_ $C(asc) _ "/"_ Rname _ "_D.php", 1)  
Do SftpAuto (RnameT, "/usr/home/biskate/public_html/bmirwin.com/vnode/routineT/"_ $C(asc) _ "/"_ Rname _ "_T.php", 1)
```


Automated SFTP Code

```
SftpAuto (INFILE,OUTFILE,DEL)
New host, port, username, password, ssh, SftpSession
Set DEL=$Get (DEL,0)
Set host="insert host IP address"
Set port=22
Set username="insert username"
Set password="insert password"
Set ssh = ##class (%Net.SSH.Session) .%New ()
Do ssh.Connect (host,port)
Do ssh.AuthenticateWithUsername (username,password)
Do ssh.OpenSFTP (.SftpSession)
Do SftpSession.Put (INFILE,OUTFILE,"0666")
Do SftpSession.CloseSFTP ()
If DEL { Open INFILE: ("NRW"):5 Close INFILE:"D" }
Quit
```

Process Overview - PHP



PHP Web Page

```
<!DOCTYPE html>
<html>
<?php require 'indexFunctions.php' ?>
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=9">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <?php pagetitle($_GET["ns"]); ?>
  <style>
    table, th, td
      { border: 1px solid black; border-collapse: collapse;
        font-family: "Courier New"; color: blue; padding: 10px; }
    td
      { vertical-align: top; }
  </style>
</head>
<body>
<table>
<tr><td>ROUTINE</td><td>TYPE</td><td>GLOBAL</td><td>LINE</td><td>CODE</td></tr>
  <?php filelist($_GET["ns"]); ?>
</table>
</body>
</html>
```

Page Title Function

```
function pagetitle($ns)
{ print " <title>Namespace: ". $ns . "</title>"; }
```

File List Function

```
function filelist($ns)
{
    $path = "../routineD/" . substr($ns, 0, 1) . "/";
    $dirArray = scandir($path);
    fileloop($dirArray, $ns);
}
```

File Loop Function

```
function fileloop($dirArray, $ns)
{
  for ($x = 0; $x < count($dirArray); $x++ )
  {
    if (($dirArray[$x] != ".")
        and ($dirArray[$x] != "..")
        and $ns == substr($dirArray[$x], 0, strlen($ns)) )
    {
      $RoutineName = substr($dirArray[$x], 0, strlen($dirArray[$x])-6);
      unset($diagram);
      unset($routine);
      loadDiagram($RoutineName, $diagram);
      loadRoutine($RoutineName, $routine);
      displayInformation($routine, $diagram, $RoutineName);
    }
  }
}
```

Load Diagram Function

```
function loadDiagram($RoutineName, &$diagram)
{
    // Identify and Open the Diagram File.
    $fullname = "../routined/" . substr($RoutineName,0,1) . "/" . $RoutineName . "_D.php";
    $myfile = fopen($fullname, "r");
    //
    // Load the Diagram elements into the diagram array.
    $codeline = 1; $element = 1;
    while (!feof($myfile))
    {
        $singleline = fgets($myfile);
        $diagram[$codeline][$element] = $singleline;
        $element = $element + 1;
        if ( substr($singleline, 0, 2) == '\n' )
            { $codeline = $codeline + 1; $element = 1; }
    }
    fclose($myfile);
}
```

Load Routine Function

```
function loadRoutine($RoutineName, &$routine)
{
    // Identify, Open, and Load the Routine file into the routine array.
    $fullname = "../routineT/" . substr($RoutineName,0,1) . "/" . $RoutineName . "_T.php";
    $myfile = fopen($fullname, "r") or die("Unable to open file!");
    $codeline = 1;
    while (!feof($myfile))
        { $routine[$codeline++] = fgets($myfile); }
    fclose($myfile);
}
```


Display Information Function

```
function displayInformation(&$routine, &$diagram, $RoutineName)
{
    // Display the Diagram elements from the diagram array.
    for($codeline = 1; $codeline < count($diagram); $codeline++ )
    {
        $globalHit = 0;
        for($element = 1; $element-1 < count($diagram[$codeline]); $element++ )
            { if ( strpos($diagram[$codeline][$element], "COS,12,^") !== FALSE ) { $globalHit = 1; } }
        .
        Additional Testing Functionality
        .
        if ( $globalHit === 1 )
        {
            print "<tr>";
            print "<td><a href=../RoutineDg/MainDisplay.php?routine=" . $RoutineName . ">" . $RoutineName . "</a></td>";
            print "<td>" . substr($comment, 0, strlen($comment)-2) . "</td>";
            print "<td>" . substr($globalName, 0, strlen($globalName)-2) . "</td>";
            print "<td>" . $codeline . "</td>";
            print "<td>" . $routine[$codeline] . "</td>";
            print "</tr>\n";
        }
    }
}
```

Globals to be Skipped

```
if ( strpos($diagram[$codeline][$element], "COS,12,^UTILITY") !== FALSE or
      strpos($diagram[$codeline][$element], "COS,12,^%ZOSF") !== FALSE or
      strpos($diagram[$codeline][$element], "COS,12,^XTMP") !== FALSE or
      strpos($diagram[$codeline][$element], "COS,12,^%ZIS") !== FALSE or
      strpos($diagram[$codeline][$element], "COS,12,^TMP") !== FALSE )
    {}
```

Test Set to be Performed

```
if ( strpos($diagram[$codeline][$element-2], 'COS,11,$D') !== FALSE ) { $comment = $comment . '$Data, `; }
if ( strpos($diagram[$codeline][$element-2], 'COS,11,$O') !== FALSE ) { $comment = $comment . '$Order, `; }
if ( strpos($diagram[$codeline][$element-2], 'COS,11,$P') !== FALSE ) { $comment = $comment . '$Piece, `; }
if ( strpos($diagram[$codeline][$element-2], 'COS,20,X') !== FALSE ) { $comment = $comment . 'Execute, `; }
if ( strpos($diagram[$codeline][$element-2], 'COS,20,K') !== FALSE ) { $comment = $comment . 'Kill, `; }
if ( strpos($diagram[$codeline][$element-2], 'COS,20,S') !== FALSE ) { $comment = $comment . 'Set Write, `; }
if ( strpos($diagram[$codeline][$element-4], 'COS,20,S') !== FALSE and
    strpos($diagram[$codeline][$element-1], 'COS,18,=' ) !== FALSE ) { $comment = $comment . 'Set Read, `; }
if ( strpos($diagram[$codeline][$element-3], 'COS,D,,') !== FALSE and
    strpos($diagram[$codeline][$element-1], 'COS,18,=' ) !== FALSE ) { $comment = $comment . 'Set Read, `; }
if ( strpos($diagram[$codeline][$element-5], 'COS,20,S') !== FALSE and
    strpos($diagram[$codeline][$element-2], 'COS,18,=' ) !== FALSE and
    strpos($diagram[$codeline][$element-1], 'COS,18,+') !== FALSE ) { $comment = $comment . 'Set Read, `; }
if ( $comment === "" ) { $comment = $comment . 'Unknown, `; }
```

Questions And Links

View of Routine and CSV Output:

<https://www.bmirwin.com/vnode/RoutineDg/index.php>

Example of Final Output:

<https://www.bmirwin.com/vnode/dg/index.php?ns=PSI>