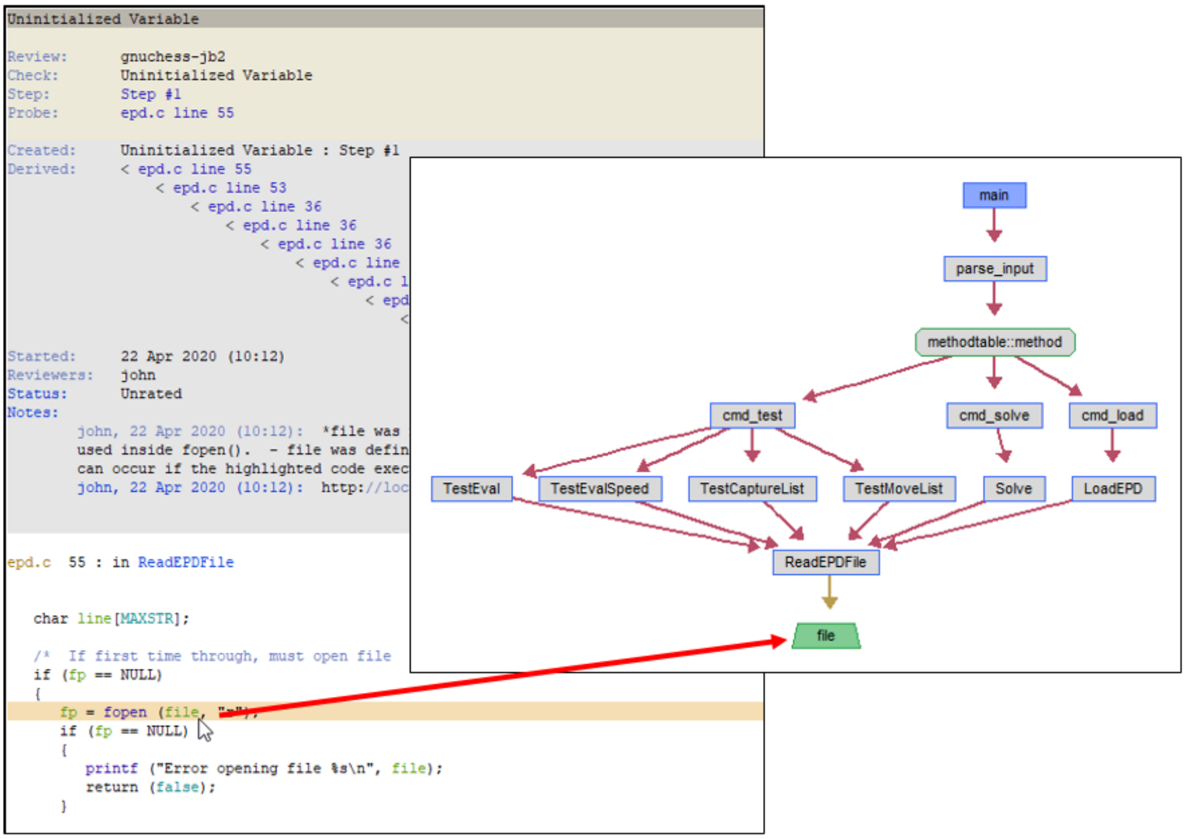
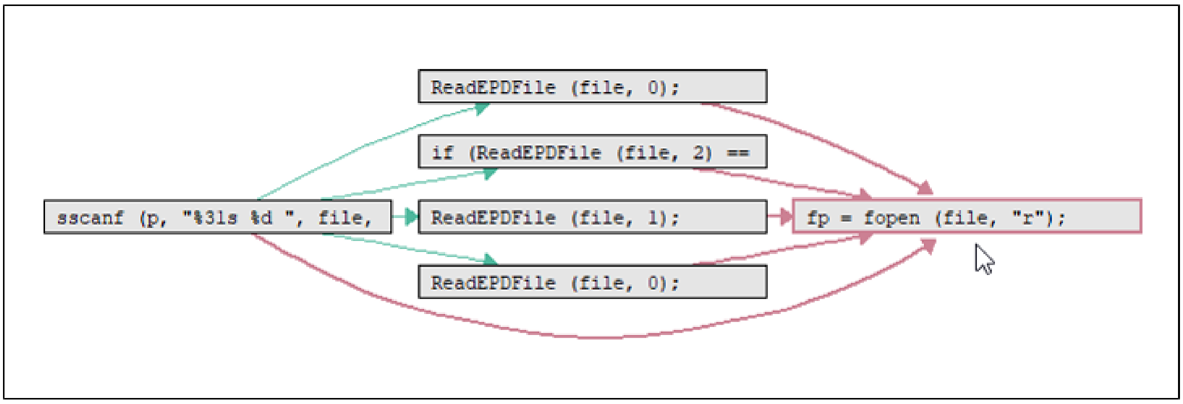
Imagix 4D Enhances CodeSonar Results

Imagix 4D provides now enhanced interpretations of GrammaTech CodeSonar results to help developers understand, debug and fix the reported issues. Imagix 4D imports results in SARIF format for each warning reported and provides enhanced, graphical visualization of each, overlaying CodeSonar results (via SARIF) on to code analysis Imagix 4D has done from the same source. For example, the warning trace information provided by CodeSonar is represented as a visual call graph in Imagix 4D:

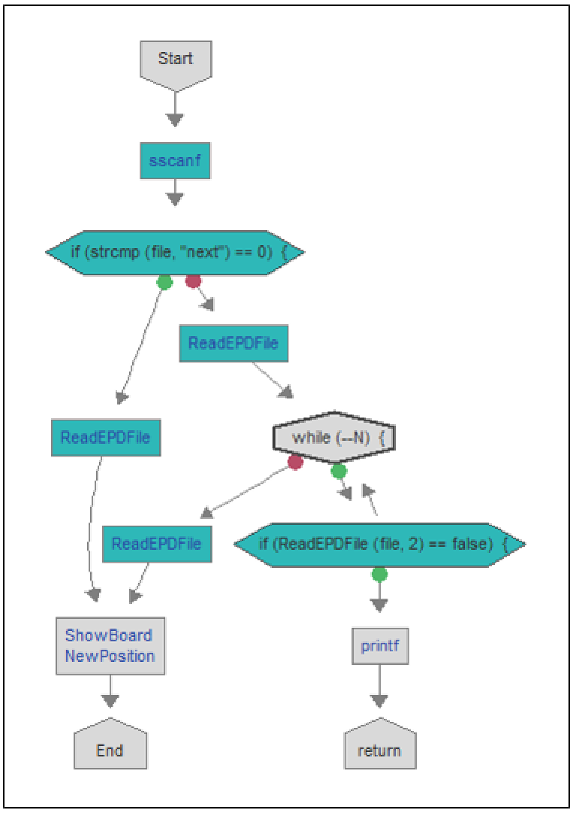


```
Uninitialized Variable

Review:      gnuchess-jb2
Check:       Uninitialized Variable
Step:        Step #1
Probe:       epd.c line 55

Created:     Uninitialized Variable : Step #1
Derived:     < epd.c line 55
                < epd.c line 53
                   < epd.c line 36
                      < epd.c line 36
                         < epd.c line 36
                            < epd.c line
                               < epd.c l
                                  < epd
                                     <

Started:     22 Apr 2020 (10:12)
Reviewers:   john
Status:      Unrated
Notes:
      john, 22 Apr 2020 (10:12):  *file was
      used inside fopen().  - file was defin
      can occur if the highlighted code exec
      john, 22 Apr 2020 (10:12):  http://loc

epd.c  55 : in ReadEPDFile


   char line[MAXSTR];

   /*  If first time through, must open file
   if (fp == NULL)
   {
      fp = fopen (file, "r");
      if (fp == NULL)
      {
         printf ("Error opening file %s\n", file);
         return (false);
      }
```

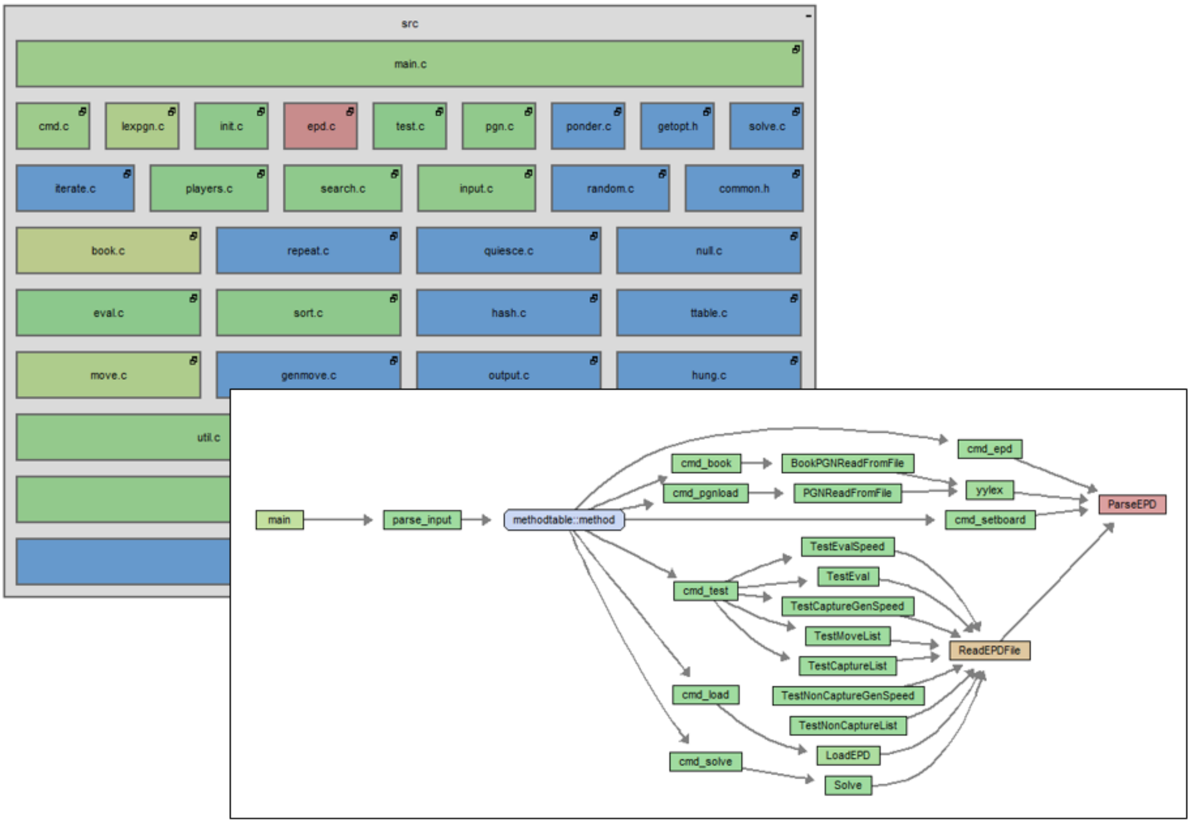
Data flow is also important to help understand the root cause of a warning. In Imagix 4D, data flows are displayed graphically to help discover error sources since data flow often passes from function to function. The following is an example data flow diagram from Imagix 4D:



In this example, the data flow shows the scanf() function that is the root cause of the uninitialized variable warning. Imagix 4D goes further by providing a flow chart representation:



Imagix 4D allows developers to record the results of their investigation such as indicating the validity, priority and severity of the error. This progress is recorded for each defect and over time, an architectural distribution of the warnings is possible. Imagix 4D creates a distribution diagram of warning “hot spots” based on the analysis results:



Imagix 4D illustrates an important use case for SARIF in extending static analysis results to enhance the understanding of warning results. Using visualization, developers can diagnose and fix bugs more efficiently.