TPF Toolkit

TPF Users Group - Fall 2009
Migrating to z/TPF Using TPF Toolkit

Mohammed Ajmal
Education Session

AIM Enterprise Platform Software
IBM z/Transaction Processing Facility Enterprise Edition 1.1.0

Any reference to future plans are for planning purposes only. IBM reserves the right to change those plans at its discretion. Any reliance on such a disclosure is solely at your own risk. IBM makes no commitment to provide additional information in the future.

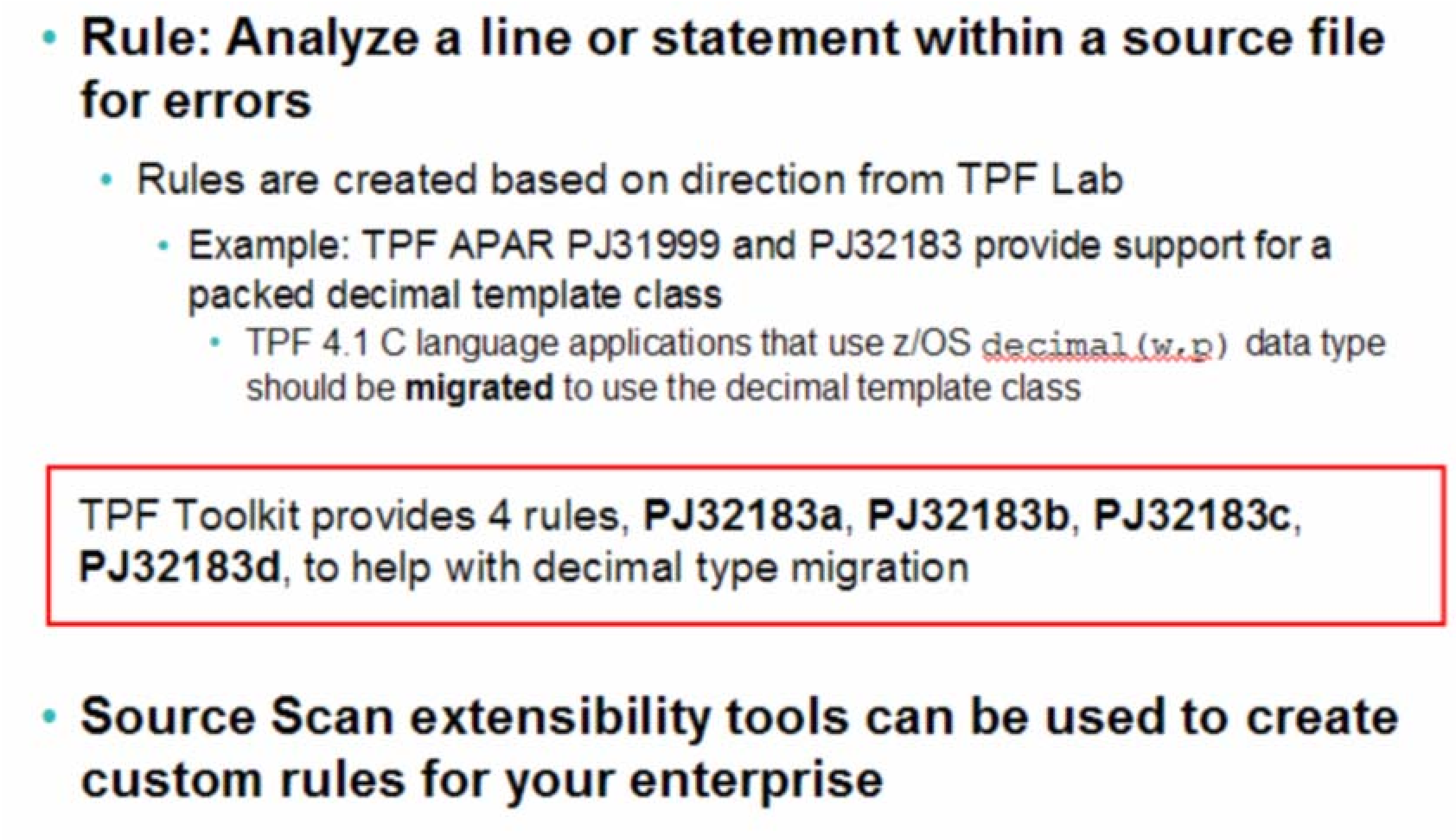
Agenda

- **Overview**
- **Source Scan Concepts**
- **Migration Process Overview**
  - *Analysis* – Planning & Estimation
  - *Implementation*
  - *Validation* – Maintaining Single Source Compliance
- **Source Scan Extensibility Tools**
- **Questions**

Overview – Source Scan

- **Source Scan – Code analysis tool in TPF Toolkit**
  - First introduced in TPF Toolkit v3.0 (2005)
- **What does Source Scan provide?**
  - **Single source**: Modify your source code so that it can be compiled on both TPF 4.1 and z/TPF
    - IBM provides many **rules** to detect migration problems
  - **Extensibility:** Add your own rules to TPF Toolkit to detect and fix migration problems specific to your environment
    - Can also be used for governance, to enforce coding standards
    - Set-up by administrators if required for migration purposes

Source Scan Concept – Rule



- **Rule: Analyze a line or statement within a source file for errors**
  - Rules are created based on direction from TPF Lab
    - Example: TPF APAR PJ31999 and PJ32183 provide support for a packed decimal template class
      - TPF 4.1 C language applications that use z/OS `decimal(w,p)` data type should be **migrated** to use the decimal template class

TPF Toolkit provides 4 rules, **PJ32183a**, **PJ32183b**, **PJ32183c**, **PJ32183d**, to help with decimal type migration

- **Source Scan extensibility tools can be used to create custom rules for your enterprise**

Types of Errors

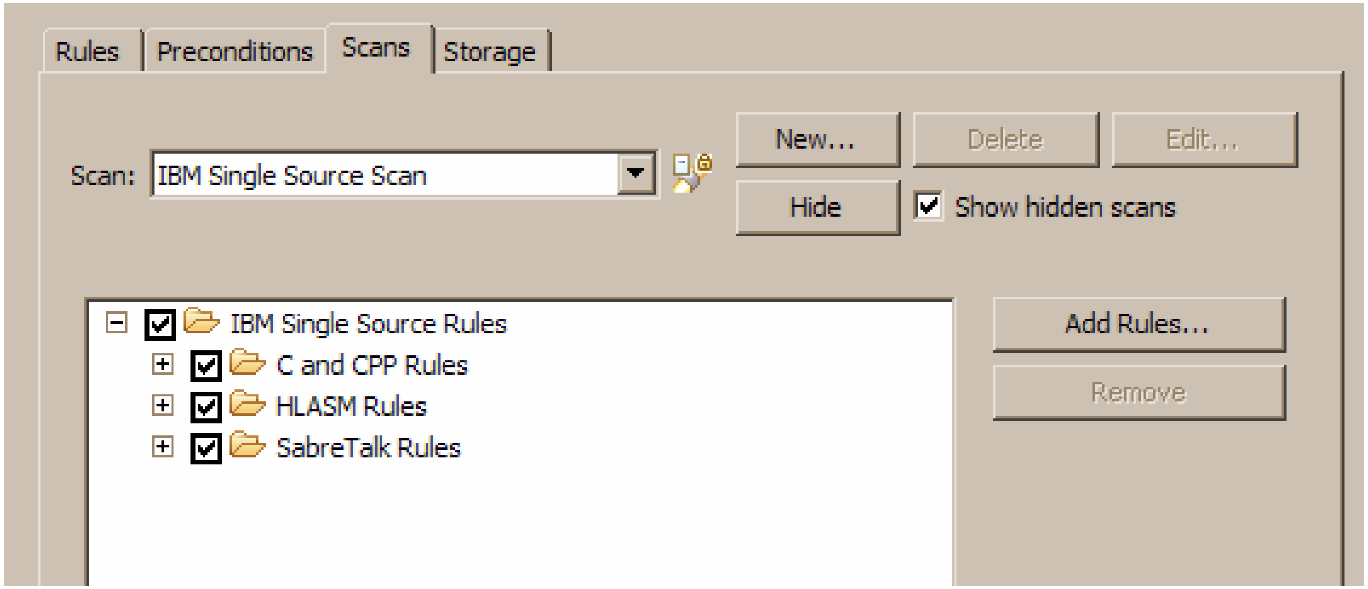
- **Rules flag either a *definite* or *potential* error**
  - *Definite error* – Source Scan has enough information to determine that there is a **definite** migration problem
  - *Potential error* – Source scan does not have enough information but there might be a **potential** migration problem
    - For example, lines in the source code might be a problem if certain runtime conditions are satisfied
    - Potential errors require further investigation by a developer

Types of Errors

- **Rules provide either an *automatic* fix or require a *manual* fix**
  - *Automatic fix* – Source Scan **can make the code change** for you to migrate your source code
  - *Manual fix* – Source scan **cannot make the code change** because additional information is required
    - Manual errors require further investigation by a developer

Source Scan Concept – Scan

- **Scan: A collection of rules used to analyze a file**
  - IBM provides the **IBM Single Source Scan**
    - Contains a set of IBM-supplied C/C++, HLAsm and SabreTalk rules
    - Can be used immediately for migration scans



Scan: IBM Single Source Scan

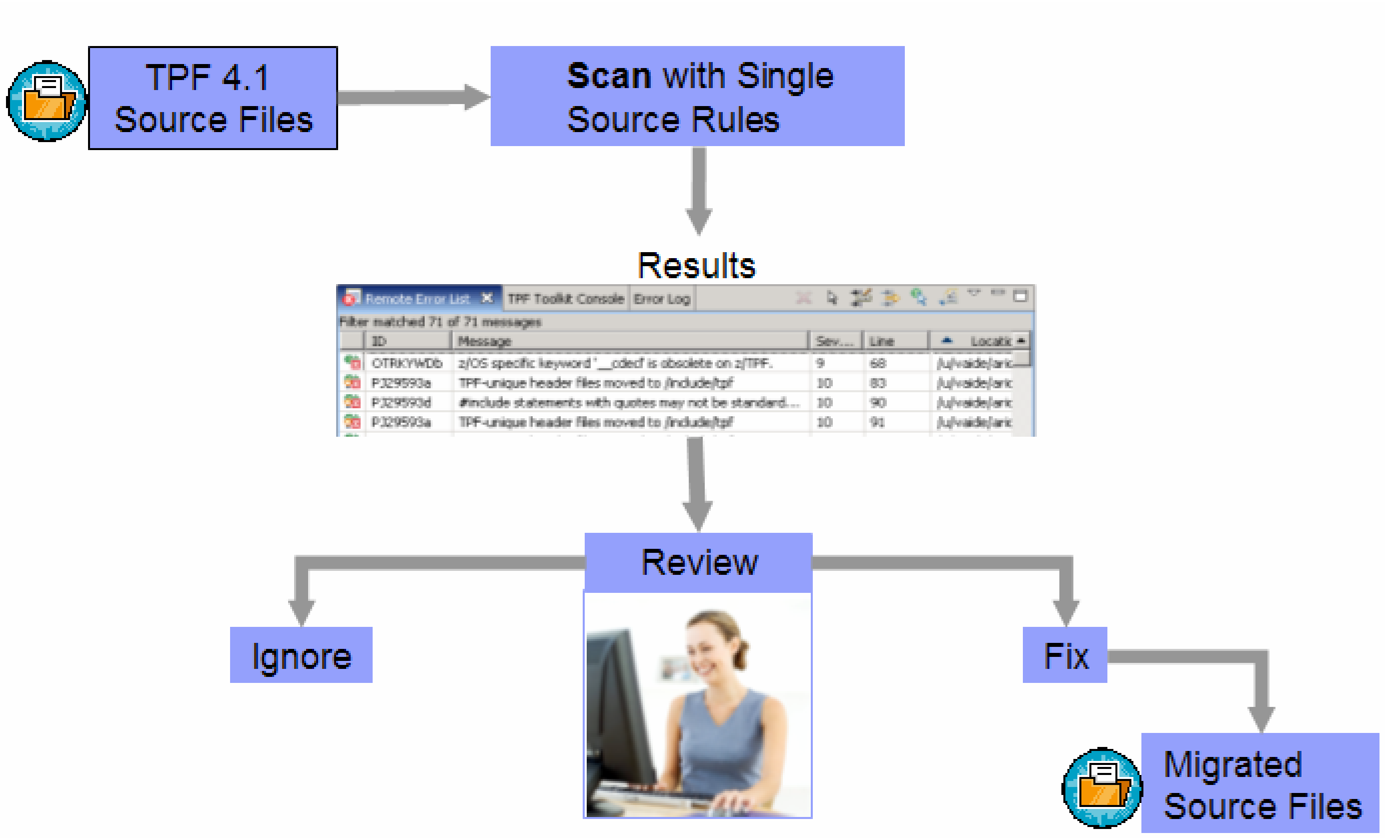
☑ Show hidden scans

- ☑ IBM Single Source Rules
  - ☑ C and CPP Rules
  - ☑ HLASM Rules
  - ☑ SabreTalk Rules

Source Scan Concept – Scan

- **IBM Single Source Scan can be modified to include custom enterprise rules**
- **Single Source extensibility tools can be used to create custom scans for your enterprise**
  - Custom scans can include IBM-supplied rules and custom rules

Migration Process Overview



TPF 4.1 Source Files

Scan with Single Source Rules

Results

Filter matched 71 of 71 messages

ID	Message	Sev...	Line	Locatio
OTRKYWDb	z/OS specific keyword '__cdecl' is obsolete on z/TPF.	9	68	/u/vaide/aric
PJ29593a	TPF-unique header files moved to /include/tpf	10	83	/u/vaide/aric
PJ29593d	#include statements with quotes may not be standard...	10	90	/u/vaide/aric
PJ29593a	TPF-unique header files moved to /include/tpf	10	91	/u/vaide/aric

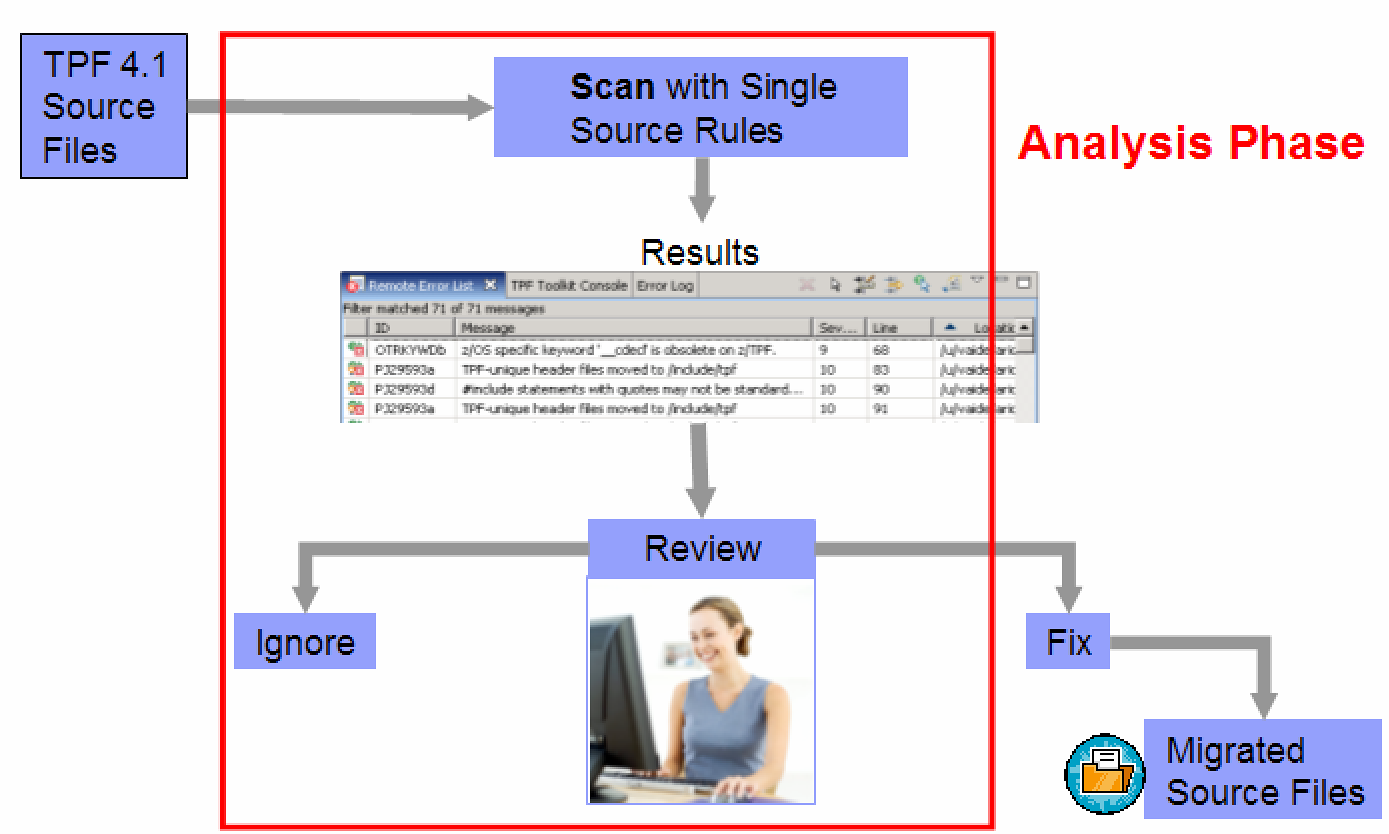
Review

Ignore

Fix

Migrated Source Files

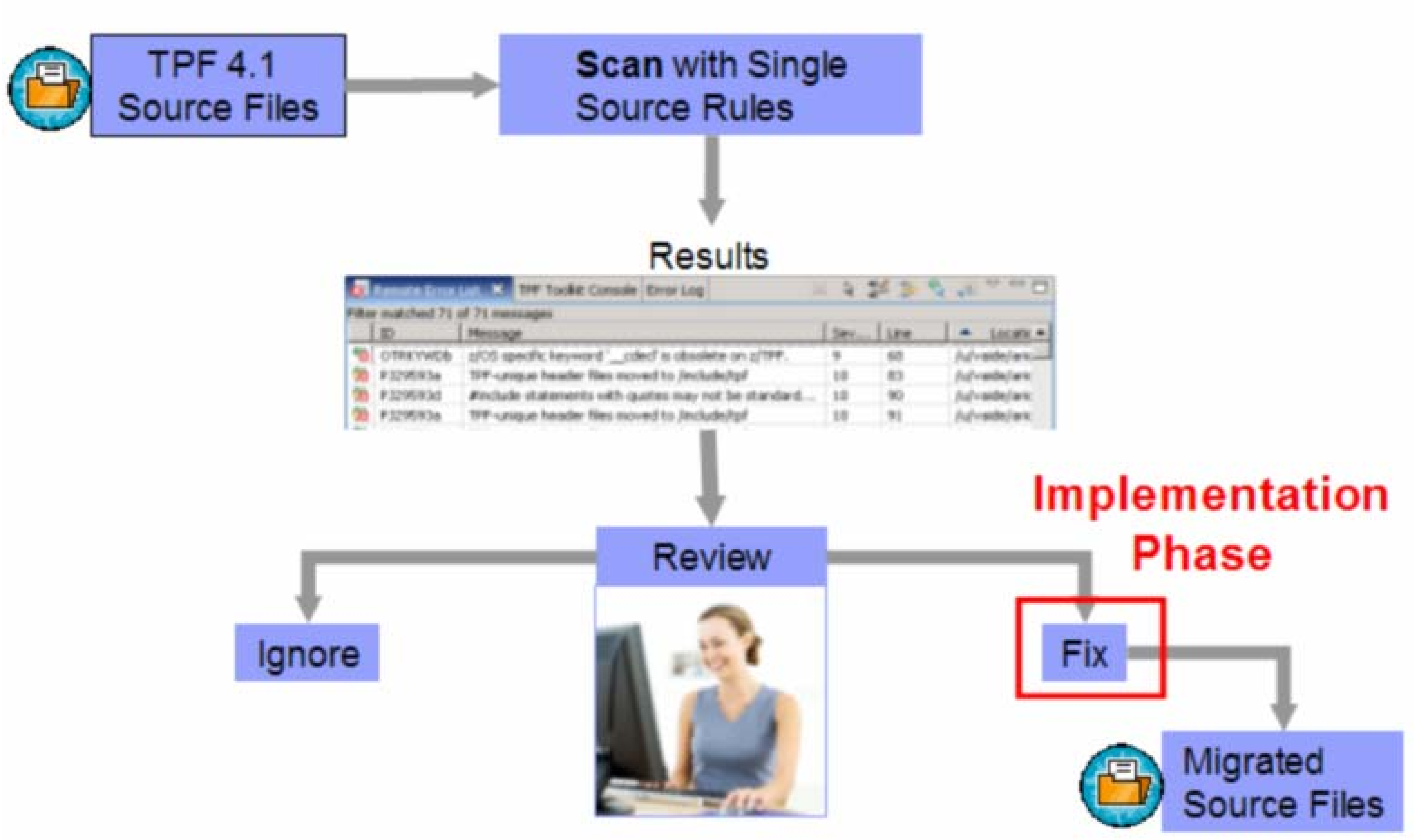
Migration Process Overview



Filter matched 71 of 71 messages

ID	Message	Sev...	Line	Locatic
OTRKYWDb	z/OS specific keyword '__cdecl' is obsolete on z/TPF.	9	68	/u/vaidearic
P329593a	TPF-unique header files moved to /include/tpf	10	83	/u/vaidearic
P329593d	#include statements with quotes may not be standard...	10	90	/u/vaidearic
P329593a	TPF-unique header files moved to /include/tpf	10	91	/u/vaidearic

Migration Process Overview

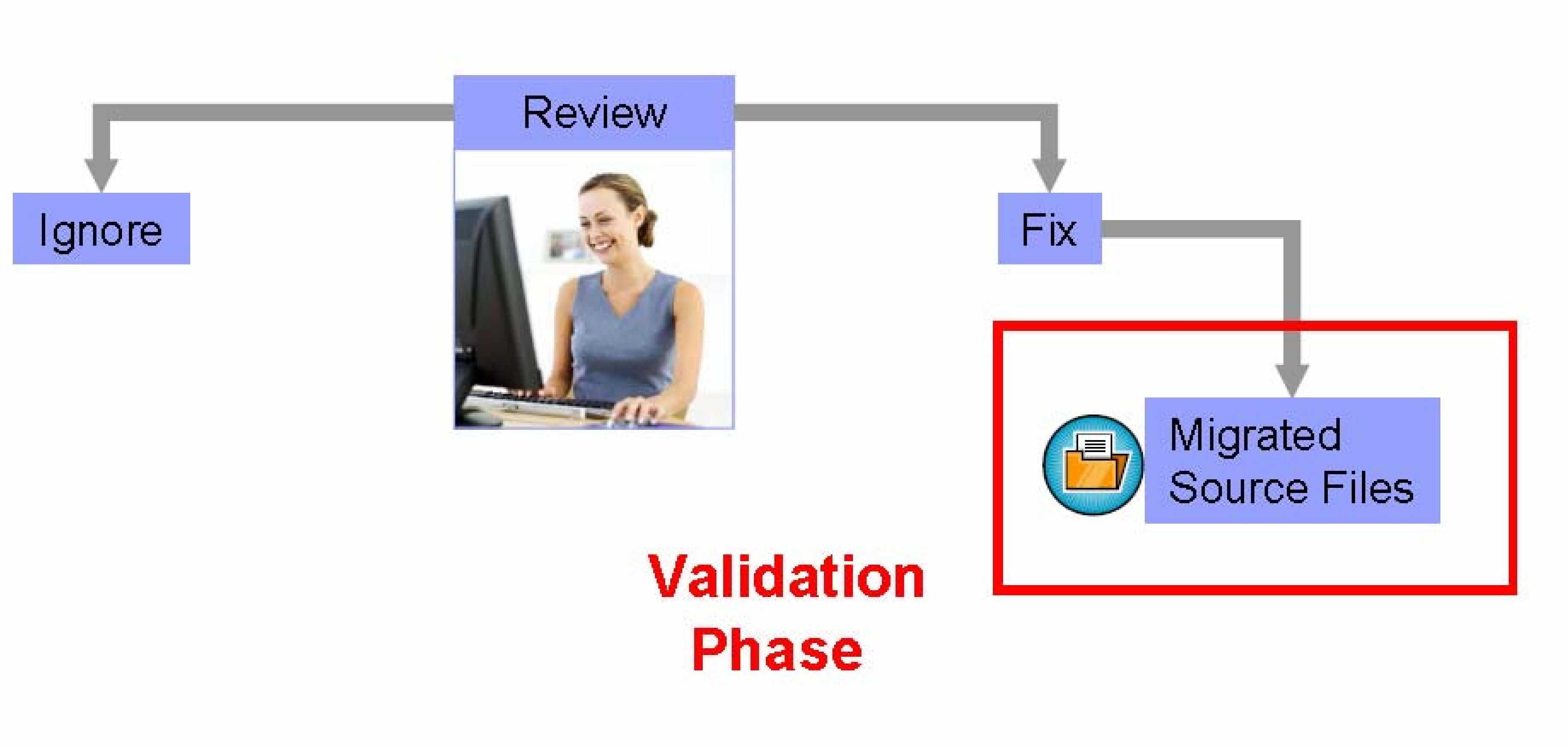


Remote Error List | TPF Toolkit Console | Error Log

Filter matched 71 of 71 messages

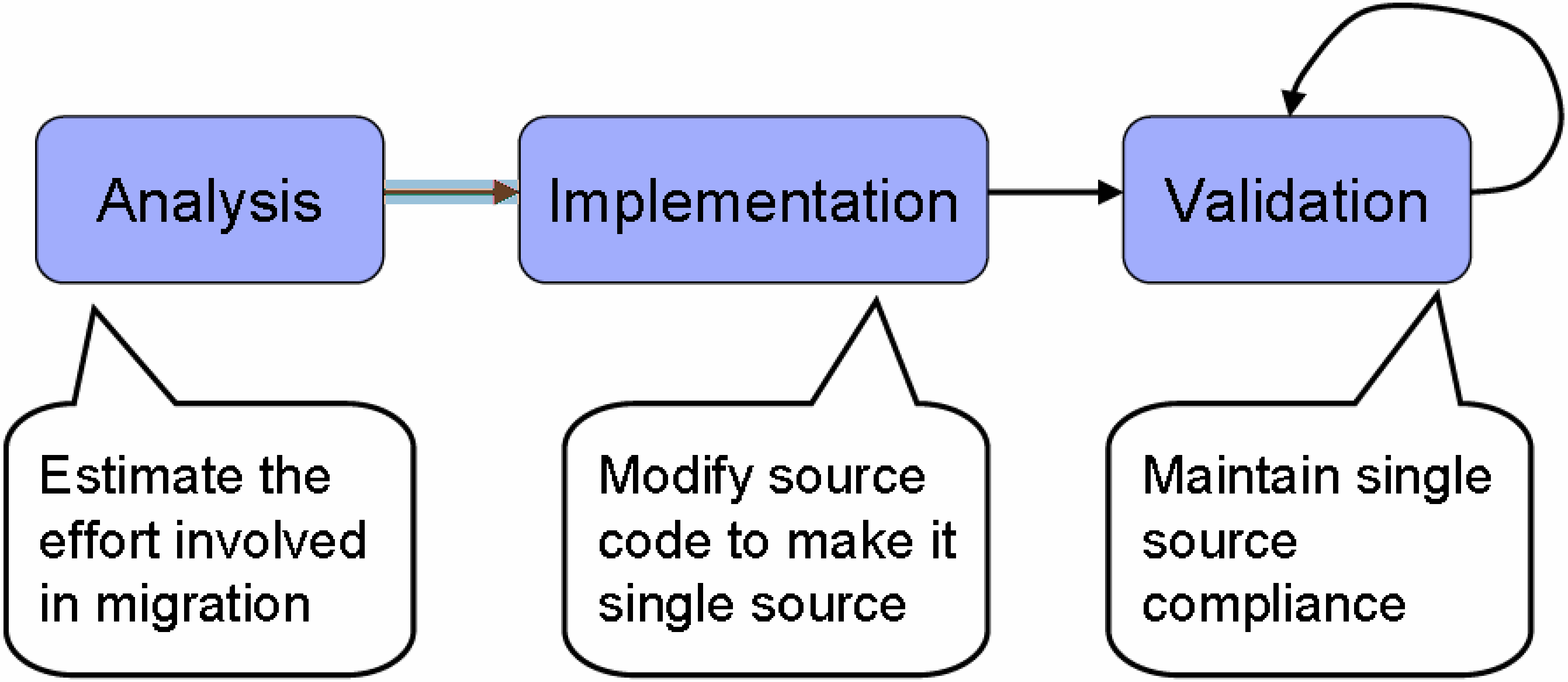
ID	Message	Sev...	Line	Locati...
OTRKYWDB	z/OS specific keyword '__cdecl' is obsolete on z/TPF.	9	60	/u/vaide/arc
P329593a	TPF-unique header files moved to /include/tpf	10	83	/u/vaide/arc
P329593d	#include statements with quotes may not be standard...	10	90	/u/vaide/arc
P329593a	TPF-unique header files moved to /include/tpf	10	91	/u/vaide/arc

Migration Process Overview



Migration Process Overview

- **Three typical stages of a migration effort:**



Analysis

Implementation

Validation

Estimate the effort involved in migration

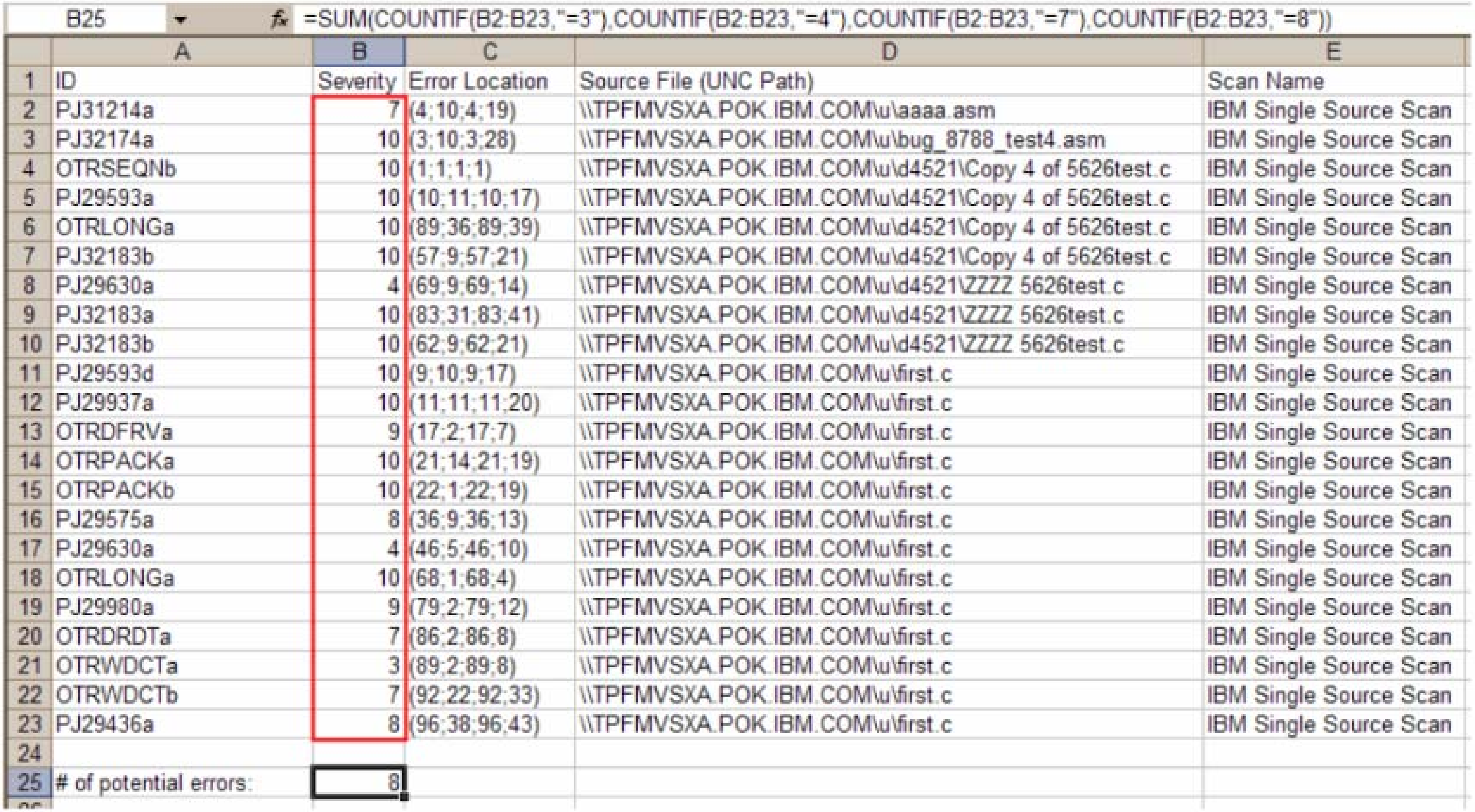
Modify source code to make it single source

Maintain single source compliance

Analysis Phase

- **TPF Toolkit provides ability to scan entire directories and generate a CSV errors file**
  - Errors file can be analyzed to produce metrics
  - For example, CSV file can be imported into Microsoft Excel
- **Estimate migration effort using metrics**
  - # of potential errors – Will require further investigation
  - # of errors with manual fixes – Will require modifications to source code by a developer

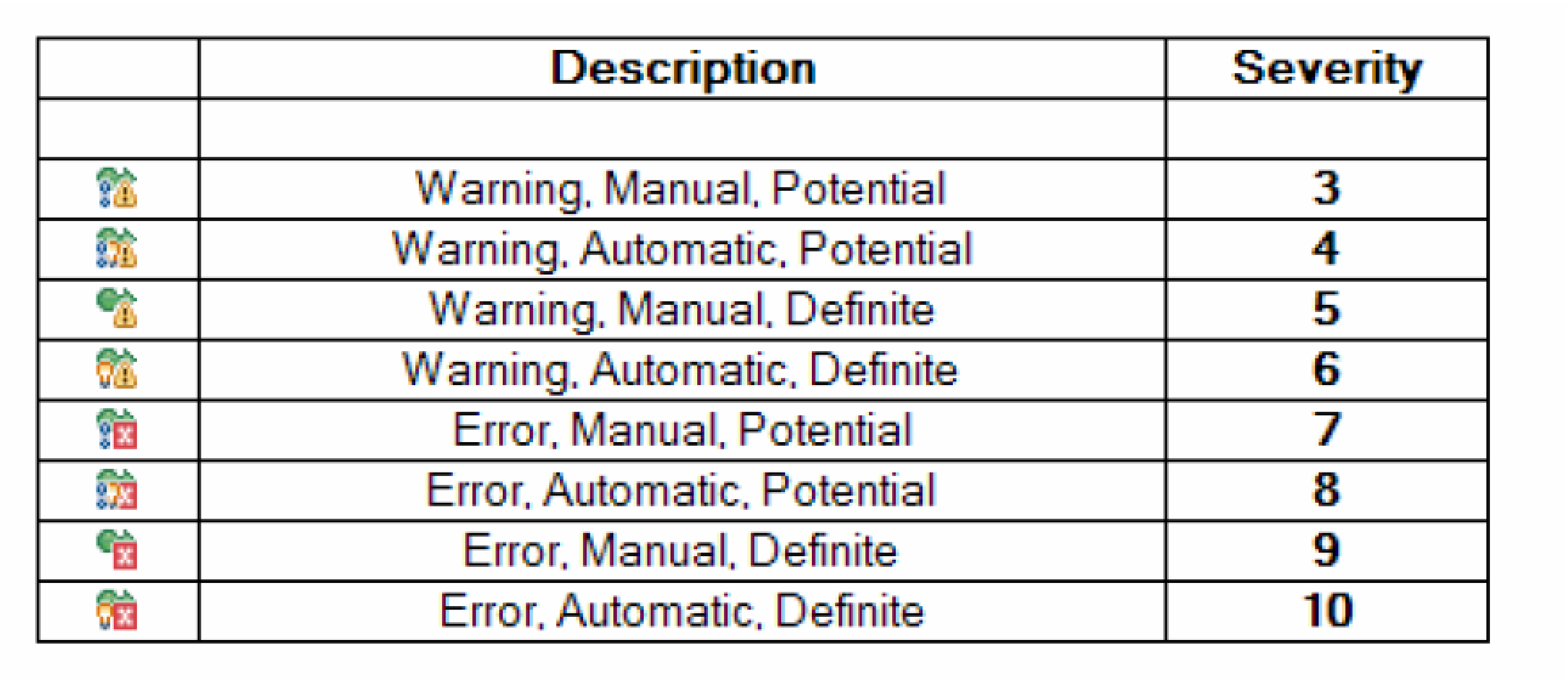
Analysis Phase



B25 fx =SUM(COUNTIF(B2:B23,"=3"),COUNTIF(B2:B23,"=4"),COUNTIF(B2:B23,"=7"),COUNTIF(B2:B23,"=8"))

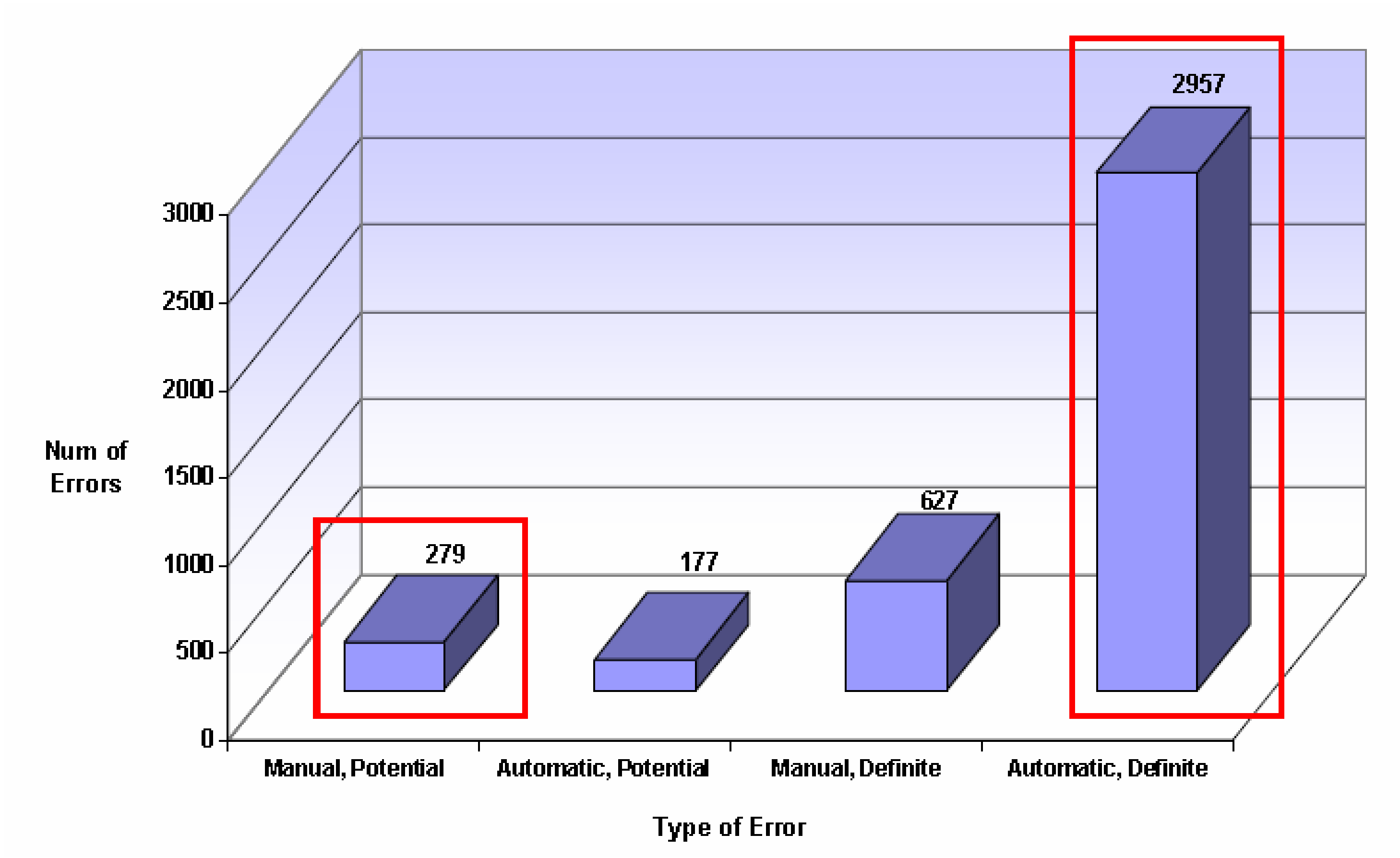
	A	B	C	D	E
1	ID	Severity	Error Location	Source File (UNC Path)	Scan Name
2	PJ31214a	7	(4;10;4;19)	\\TPFMVSXA.POK.IBM.COM\u\aaaa.asm	IBM Single Source Scan
3	PJ32174a	10	(3;10;3;28)	\\TPFMVSXA.POK.IBM.COM\u\bug_8788_test4.asm	IBM Single Source Scan
4	OTRSEQNb	10	(1;1;1;1)	\\TPFMVSXA.POK.IBM.COM\u\d4521\Copy 4 of 5626test.c	IBM Single Source Scan
5	PJ29593a	10	(10;11;10;17)	\\TPFMVSXA.POK.IBM.COM\u\d4521\Copy 4 of 5626test.c	IBM Single Source Scan
6	OTRLONGa	10	(89;36;89;39)	\\TPFMVSXA.POK.IBM.COM\u\d4521\Copy 4 of 5626test.c	IBM Single Source Scan
7	PJ32183b	10	(57;9;57;21)	\\TPFMVSXA.POK.IBM.COM\u\d4521\Copy 4 of 5626test.c	IBM Single Source Scan
8	PJ29630a	4	(69;9;69;14)	\\TPFMVSXA.POK.IBM.COM\u\d4521\ZZZZ 5626test.c	IBM Single Source Scan
9	PJ32183a	10	(83;31;83;41)	\\TPFMVSXA.POK.IBM.COM\u\d4521\ZZZZ 5626test.c	IBM Single Source Scan
10	PJ32183b	10	(62;9;62;21)	\\TPFMVSXA.POK.IBM.COM\u\d4521\ZZZZ 5626test.c	IBM Single Source Scan
11	PJ29593d	10	(9;10;9;17)	\\TPFMVSXA.POK.IBM.COM\u\first.c	IBM Single Source Scan
12	PJ29937a	10	(11;11;11;20)	\\TPFMVSXA.POK.IBM.COM\u\first.c	IBM Single Source Scan
13	OTRDFRVa	9	(17;2;17;7)	\\TPFMVSXA.POK.IBM.COM\u\first.c	IBM Single Source Scan
14	OTRPACKa	10	(21;14;21;19)	\\TPFMVSXA.POK.IBM.COM\u\first.c	IBM Single Source Scan
15	OTRPACKb	10	(22;1;22;19)	\\TPFMVSXA.POK.IBM.COM\u\first.c	IBM Single Source Scan
16	PJ29575a	8	(36;9;36;13)	\\TPFMVSXA.POK.IBM.COM\u\first.c	IBM Single Source Scan
17	PJ29630a	4	(46;5;46;10)	\\TPFMVSXA.POK.IBM.COM\u\first.c	IBM Single Source Scan
18	OTRLONGa	10	(68;1;68;4)	\\TPFMVSXA.POK.IBM.COM\u\first.c	IBM Single Source Scan
19	PJ29980a	9	(79;2;79;12)	\\TPFMVSXA.POK.IBM.COM\u\first.c	IBM Single Source Scan
20	OTRDRDTa	7	(86;2;86;8)	\\TPFMVSXA.POK.IBM.COM\u\first.c	IBM Single Source Scan
21	OTRWDCTa	3	(89;2;89;8)	\\TPFMVSXA.POK.IBM.COM\u\first.c	IBM Single Source Scan
22	OTRWDCTb	7	(92;22;92;33)	\\TPFMVSXA.POK.IBM.COM\u\first.c	IBM Single Source Scan
23	PJ29436a	8	(96;38;96;43)	\\TPFMVSXA.POK.IBM.COM\u\first.c	IBM Single Source Scan
24					
25	# of potential errors:	8			

Types of Errors



	Description	Severity
	Warning, Manual, Potential	3
	Warning, Automatic, Potential	4
	Warning, Manual, Definite	5
	Warning, Automatic, Definite	6
	Error, Manual, Potential	7
	Error, Automatic, Potential	8
	Error, Manual, Definite	9
	Error, Automatic, Definite	10

Analysis Phase



Demo – Generating Metrics in Analysis Phase

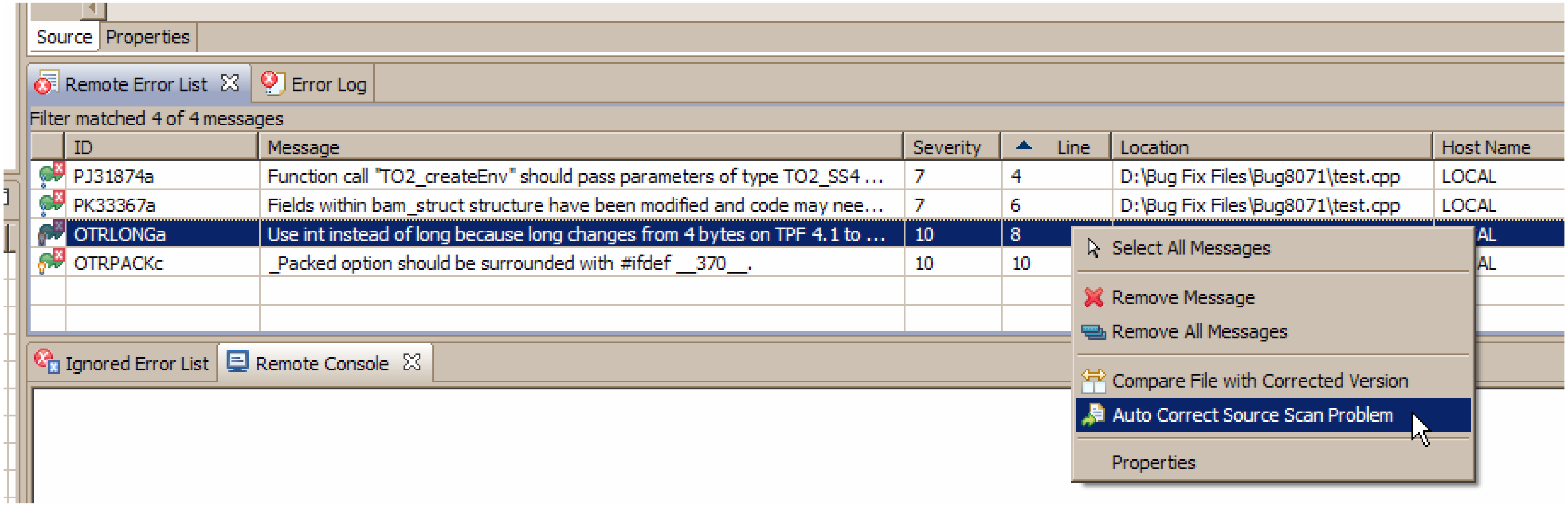
Analysis Phase

- **Potential errors can be ignored through TPF Toolkit**
  - Solution for a multi-user environment
  - Ignored errors are not flagged in subsequent scans by any user

Demo – Ignoring Errors through TPF Toolkit

Implementation Phase: Automatic Fixes

- **Fix errors using TPF Toolkit**
  - *Auto Correct action* – Fix multiple errors automatically through the Remote Error List

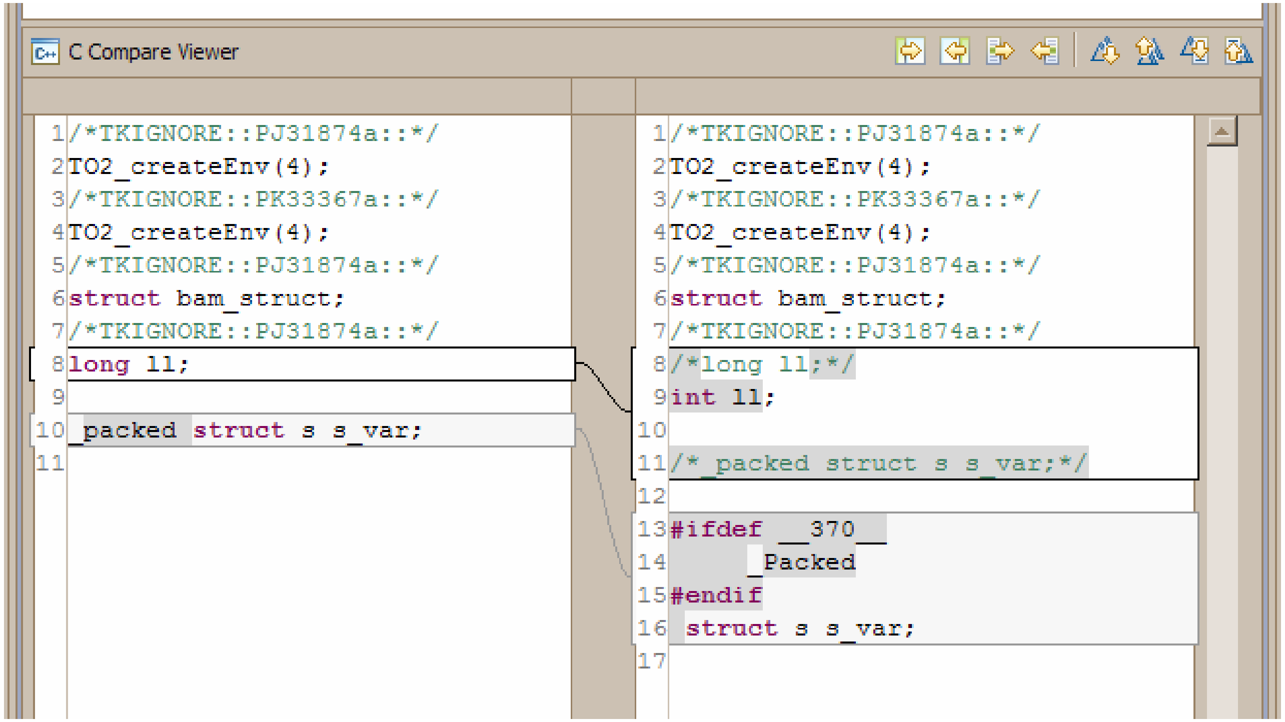


Filter matched 4 of 4 messages

ID	Message	Severity	▲ Line	Location	Host Name
PJ31874a	Function call "TO2_createEnv" should pass parameters of type TO2_SS4 …	7	4	D:\Bug Fix Files\Bug8071\test.cpp	LOCAL
PK33367a	Fields within bam_struct structure have been modified and code may nee…	7	6	D:\Bug Fix Files\Bug8071\test.cpp	LOCAL
OTRLONGa	Use int instead of long because long changes from 4 bytes on TPF 4.1 to …	10	8		AL
OTRPACKc	_Packed option should be surrounded with #ifdef __370__.	10	10		AL

Implementation Phase: Automatic Fixes

- **Fix errors using TPF Toolkit**
  - *Compare editor* – Fix errors by comparing original source file with a corrected version

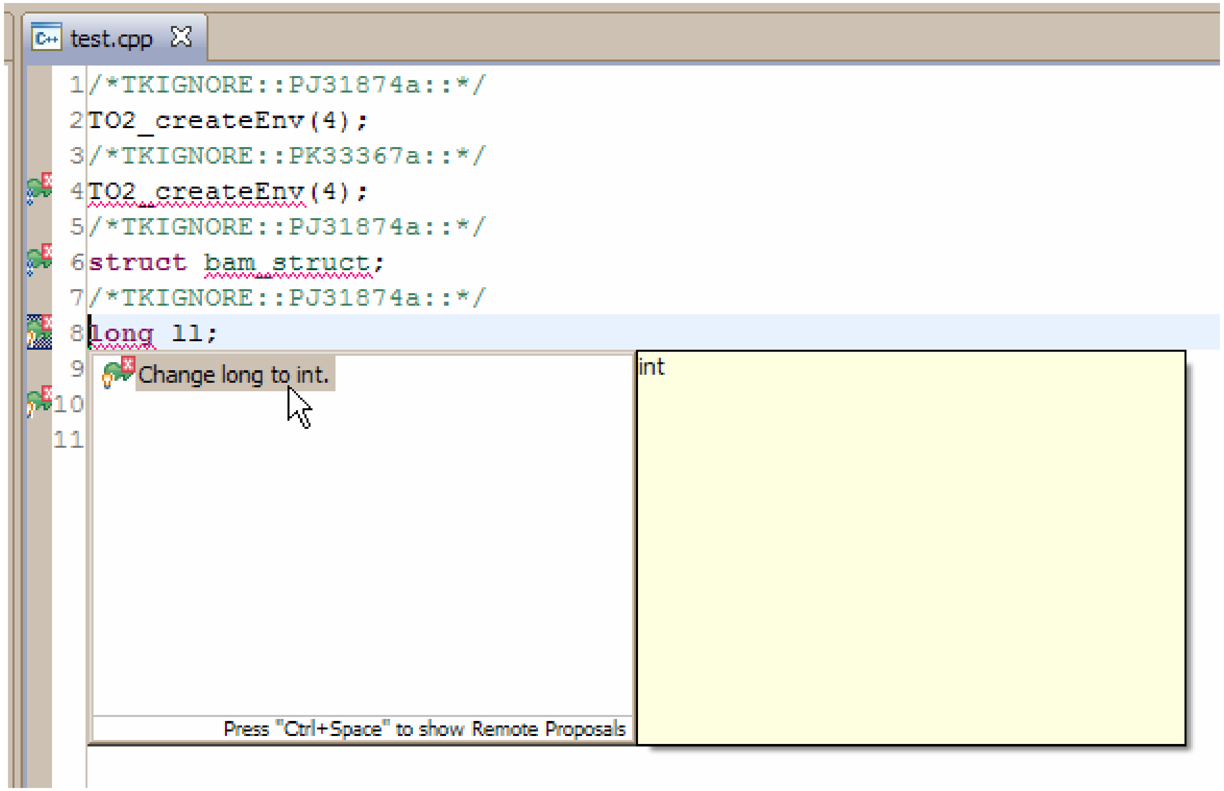


```
/*TKIGNORE::PJ31874a::*/
TO2_createEnv(4);
/*TKIGNORE::PK33367a::*/
TO2_createEnv(4);
/*TKIGNORE::PJ31874a::*/
struct bam_struct;
/*TKIGNORE::PJ31874a::*/
long ll;

_packed struct s s_var;

```

```
/*TKIGNORE::PJ31874a::*/
TO2_createEnv(4);
/*TKIGNORE::PK33367a::*/
TO2_createEnv(4);
/*TKIGNORE::PJ31874a::*/
struct bam_struct;
/*TKIGNORE::PJ31874a::*/
/*long ll;*/
int ll;

/*_packed struct s s_var;*/

#ifdef __370__
      _Packed
#endif
 struct s s_var;

```

Implementation Phase: Automatic Fixes

- **Fix errors using TPF Toolkit**
  - *Quick-fix action* – Fix individual errors through an editor



```
/*TKIGNORE::PJ31874a::*/
TO2_createEnv(4);
/*TKIGNORE::PK33367a::*/
TO2_createEnv(4);
/*TKIGNORE::PJ31874a::*/
struct bam_struct;
/*TKIGNORE::PJ31874a::*/
long ll;
```

Implementation Phase: Manual Fixes

- **Developers need to investigate these errors**
- **A number of resources are available to assist:**
  - TPF Toolkit documentation
  - Information in the TPF Single Source APARs
  - z/TPF Migration Guide

  http://www-306.ibm.com/software/htp/tpf/pages/gtpm1mst.pdf

Demo – Fixing Errors through TPF Toolkit

Validation Phase

- **Ensure that migrated code remains single source compliant**

- **Validation when files are edited by users**
  - Errors are automatically flagged when file is saved

Validation Phase

- **Command-line tool to scan source directories**
  - Schedule nightly scans of source code
- **Validation using custom enterprise actions**
  - For example, create a source check-in action which only proceeds if no single source errors are present in file

Demo – Validation in TPF Toolkit

Source Scan Extensibility Tools

- **Extensibility tools allow you to create:**
  - Custom rules – Detect errors specific to your enterprise
  - Custom scans – Group rules according to your needs
    - For example, create a validation scan containing custom governance rules
  - Custom categories – Used to organize rules
    - For example, organize rules by language

Rule Creation Summary

- **Step 1 – Choose or Create a Category**
- **Step 2 – Invoke the New Rule Wizard**
- **Step 3 – Supply general rule information**
- **Step 4 – Choose a language and Detection Template**
- **Step 5 – Provide required template data**
- **Step 6 – Choose and customize a fix**
- **Step 7 – Add the new rule to a scan**
- **Step 8 – Test the new rule**

Templates

- **Detection Templates**
  - Used to describe what the rule will look for
  - C/C++ detection templates
  - ASM detection template

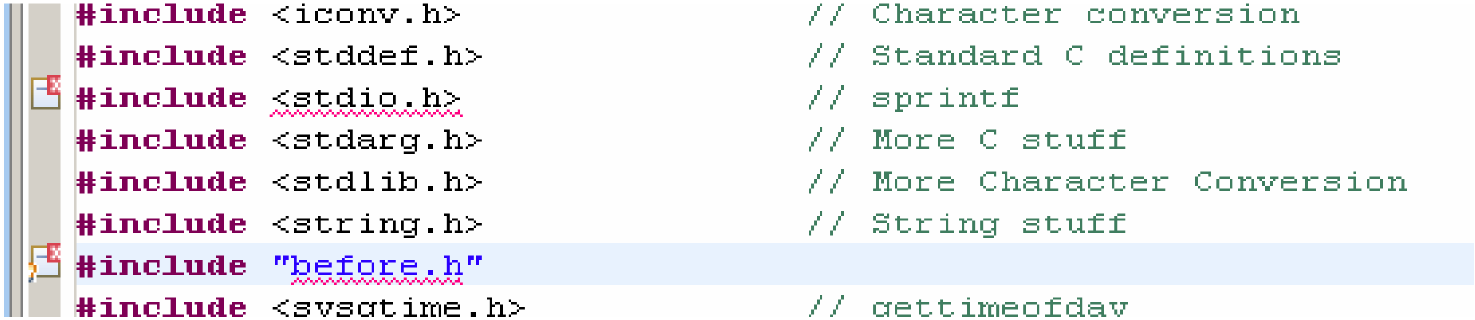
- **Fix Templates**
  - Used to describe how a fix can be made if an error is found
  - C/C++ fix templates
  - ASM fix template

C/CPP Templates

- **CPP Include Statement Rule Template**
- **CPP Plain Text Matching Rule Template**
- **CPP Type Usage Rule Template**
- **CPP Directive Rule Template**
- **CPP Function Call Rule Template**
- **CPP Constant Rule Template**

CPP Include Statement Rule Template

- **Examines only #include directives**
- **Examples:**
  - Flag before.h
  - Flag stdio.h when it is included using angled brackets

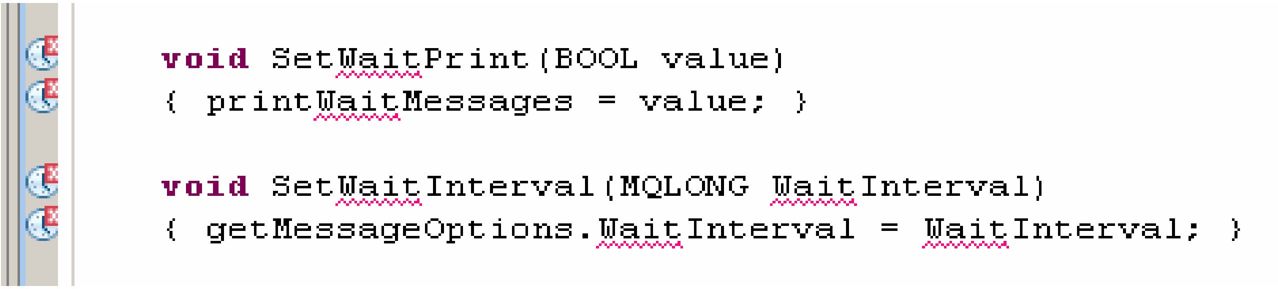


```
#include <iconv.h>                 // Character conversion
#include <stddef.h>                // Standard C definitions
#include <stdio.h>                 // sprintf
#include <stdarg.h>                // More C stuff
#include <stdlib.h>                // More Character Conversion
#include <string.h>                // String stuff
#include "before.h"
#include <sysgtime.h>              // gettimeofday
```

- **Single Source example: PJ29593b - Remove $ symbol from all header file names**

CPP Plain Text Matching Rule

- **Examines all text in the file (option to exclude comment lines)**
- **Most generic rule - use this template when other more specific templates don't apply**
- **Example: Flag all usage of the word 'Wait'**

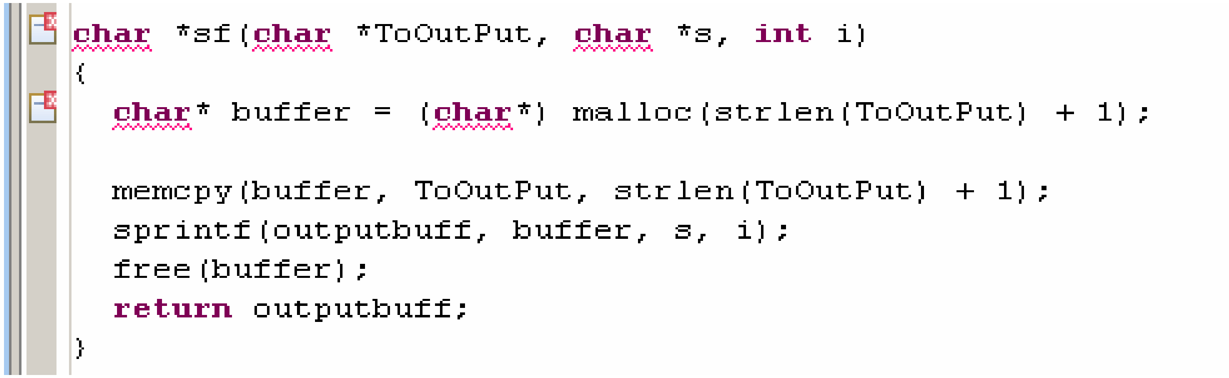


```
void SetWaitPrint(BOOL value)
{ printWaitMessages = value; }

void SetWaitInterval(MQLONG WaitInterval)
{ getMessageOptions.WaitInterval = WaitInterval; }
```

- **Single Source example: PJ29436a - OTREXTCa - Upper case "C" required in linkage specification**

CPP Type Usage Rule

- **Examines text only when it is used as a type name**
- **Example: Flag all char* as return type, parameter type, declaration type, cast type**

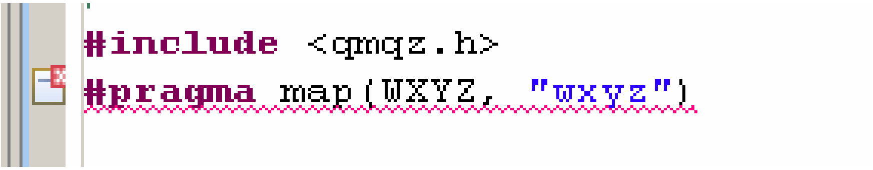


```
char *sf(char *ToOutPut, char *s, int i)
{
   char* buffer = (char*) malloc(strlen(ToOutPut) + 1);

   memcpy(buffer, ToOutPut, strlen(ToOutPut) + 1);
   sprintf(outputbuff, buffer, s, i);
   free(buffer);
   return outputbuff;
}
```

- **Single Source example: OTRWDCTa - Use wide characters (wchar_t) for in-memory processing only**

CPP Directive Rule Template

- **Examines directives only**
- **Example: Flag all usage of #pragma directive**

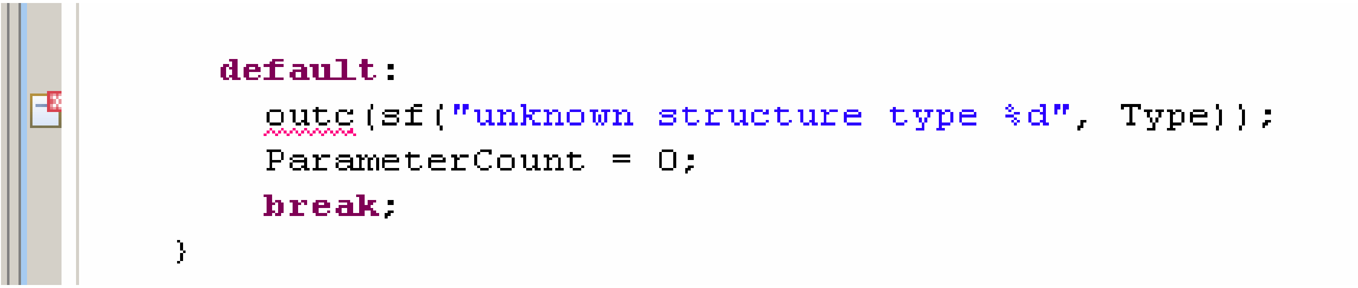


```
#include <qmqz.h>
#pragma map(WXYZ, "wxyz")
```

- **Single Source example: OTRPRAGa - Replace or remove #pragma directives**

CPP Function Call Rule Template

- **Examines function calls only**
- **Example: Calls to function named outc**

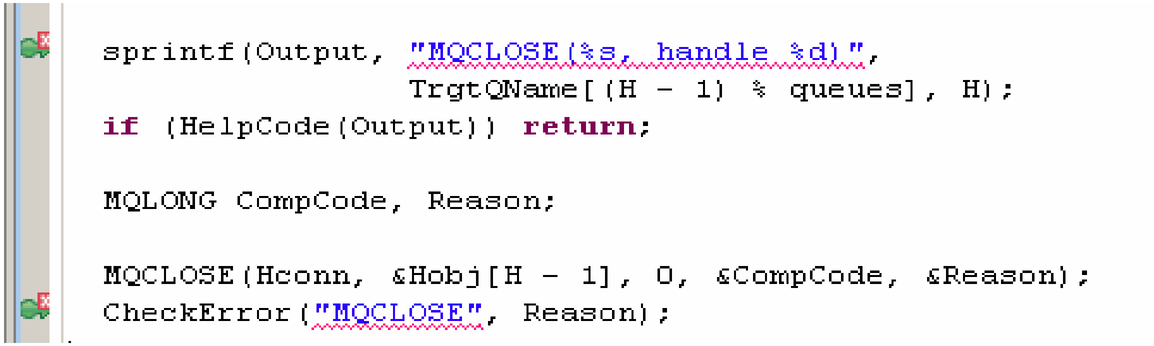


```
    default:
      outc(sf("unknown structure type %d", Type));
      ParameterCount = 0;
      break;
  }
```

- **Single Source example: PJ29957a - setlocale function changed for obsolete category LC_TOD**

CPP Constant Rule Template

- **Examines only constants (i.e. string literals and numbers)**
- **Example: Find MQCLOSE in constants**



```
sprintf(Output, "MQCLOSE(%s, handle %d)",
                TrgtQName[(H - 1) % queues], H);
if (HelpCode(Output)) return;

MQLONG CompCode, Reason;

MQCLOSE(Hconn, &Hobj[H - 1], 0, &CompCode, &Reason);
CheckError("MQCLOSE", Reason);
```

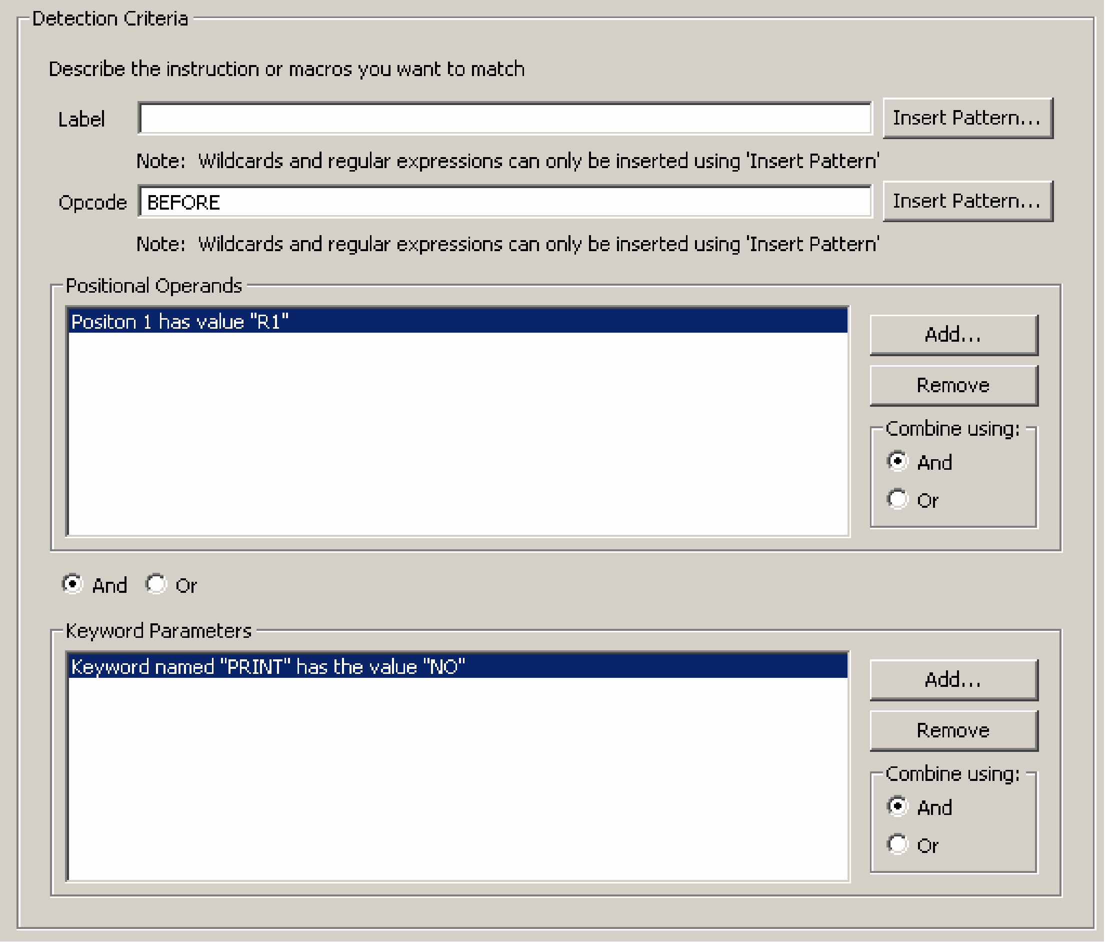
- **Single Source example: OTRWDCTb - Wide characters coded using hex values break single source**

CPP Fix Templates

- **Replace matched text**
  - Replaces all text matched at the error location with the specified replacement text
  - Example: Replace one data type name with another data type name.
- **Insert additional line**
  - Inserts a line into the code
  - Example: Add an include statement

HLASM Detection Template

- **One template matches instructions based on**
  - Label
  - Opcode
  - Positional Operands
  - Keyword Operands



Detection Criteria

Describe the instruction or macros you want to match

Label: (empty)

Note: Wildcards and regular expressions can only be inserted using 'Insert Pattern'

Opcode: BEFORE

Note: Wildcards and regular expressions can only be inserted using 'Insert Pattern'

Positional Operands

Positon 1 has value "R1"

Combine using: ● And ○ Or

● And ○ Or

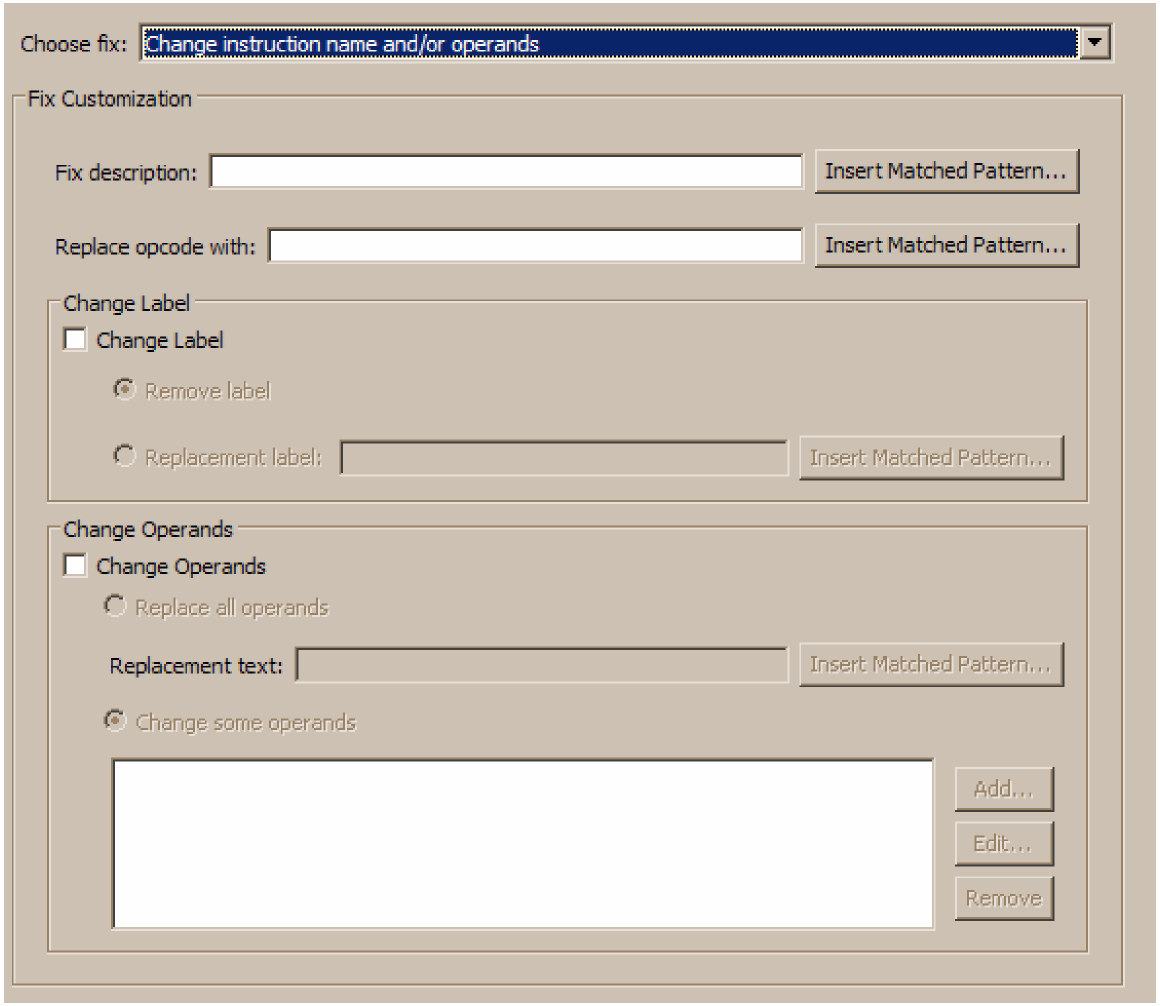
Keyword Parameters

Keyword named "PRINT" has the value "NO"

Combine using: ● And ○ Or

HLASM Fix Template

- **Modify the matched opcode, operands or label**



Choose fix: Change instruction name and/or operands

Fix Customization

Fix description: [Insert Matched Pattern...]

Replace opcode with: [Insert Matched Pattern...]

Change Label

□ Change Label

● Remove label

○ Replacement label: [Insert Matched Pattern...]

Change Operands

□ Change Operands

○ Replace all operands

Replacement text: [Insert Matched Pattern...]

● Change some operands

Demo – Source Scan Extensibility Tools

Questions

- **???**

Trademarks

- IBM is a trademark of International Business Machines Corporation in the United States, other countries, or both.
- Other company, product, or service names may be trademarks or service marks of others.

- Notes
- Performance is in Internal Throughput Rate (ITR) ratio based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput improvements equivalent to the performance ratios stated here.
- All customer examples cited or described in this presentation are presented as illustrations of the manner in which some customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics will vary depending on individual customer configurations and conditions.
- This publication was produced in the United States. IBM may not offer the products, services or features discussed in this document in other countries, and the information may be subject to change without notice. Consult your local IBM business contact for information on the product or services available in your area.

- All statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only.

- Information about non-IBM products is obtained from the manufacturers of those products or their published announcements. IBM has not tested those products and cannot confirm the performance, compatibility, or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

- Prices subject to change without notice. Contact your IBM representative or Business Partner for the most current pricing in your geography.

- This presentation and the claims outlined in it were reviewed for compliance with US law. Adaptations of these claims for use in other geographies must be reviewed by the local country counsel for compliance with local laws.