

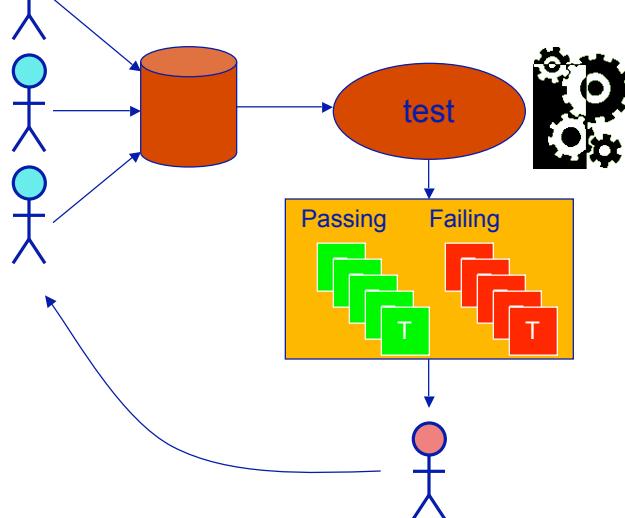
Automation to Assist Software Debugging

James A. Jones
College of Computing
Georgia Institute of Technology

Supported by Tata Consultancy Services, NSF, NASA
Boeing Commercial Airplanes



Overview



CERCS IAB Workshop April 2007



Debugging Phases

- Make changes to the program
- Test
- Observe failure
- Locate bugs
- Identify possible fixes
- Choose best fix
- Implement fix



CERCS IAB Workshop April 2007



Debugging Phases

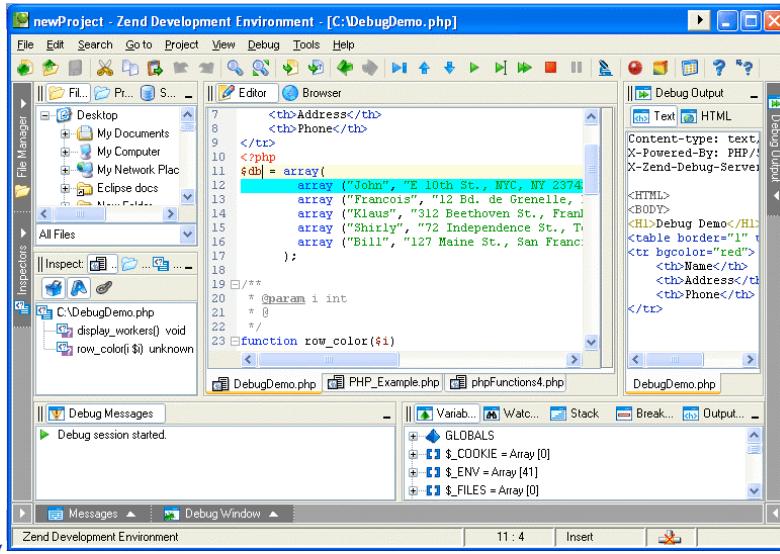
- Make changes to the program
- Test
- Observe failure
- Locate bugs – Fault Localization
- Identify possible fixes
- Choose best fix
- Implement fix



CERCS IAB Workshop April 2007



Current Practice



A

ristotle
Research Group

CERCs IAB Workshop

April 2007



Tarantula: Fault Localization

```
mid() {
    int x,y,z,m;
1:read('Enter 3 integers:');
2:m = z;
3:if (y<z)
4:    if (x<y)
5:        m = y;
6:    else if (x>z)
7:        m = y;
8:else
9:    if (x>y)
10:       m = y;
11:    else if (x>z)
12:       m = x;
13:print("Middle number is:", m,
      )
```

**Technique uses
Dynamic information**

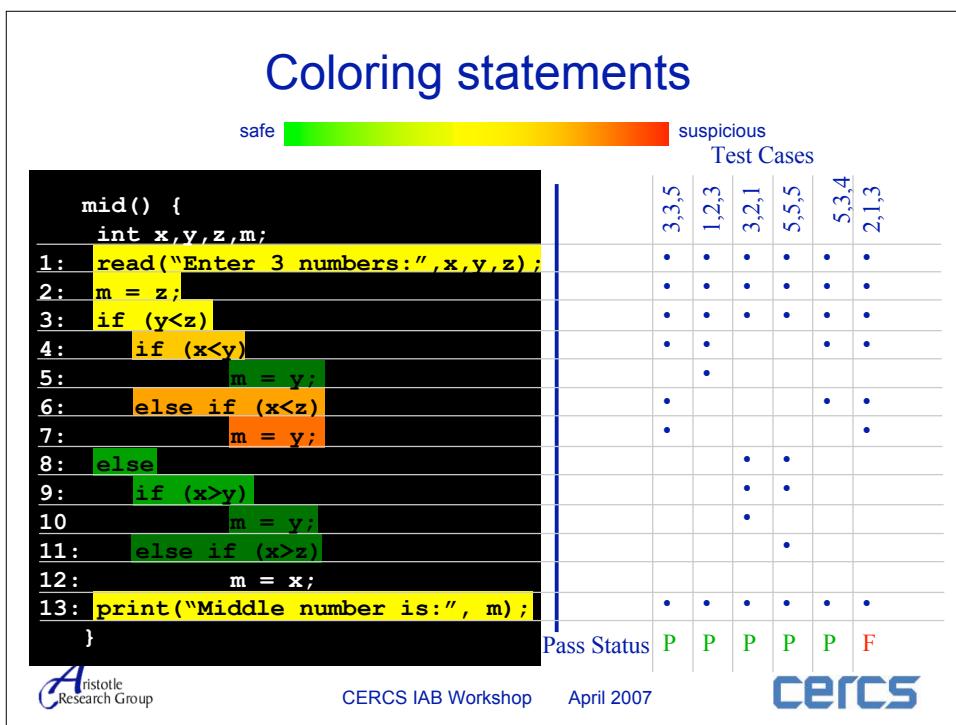
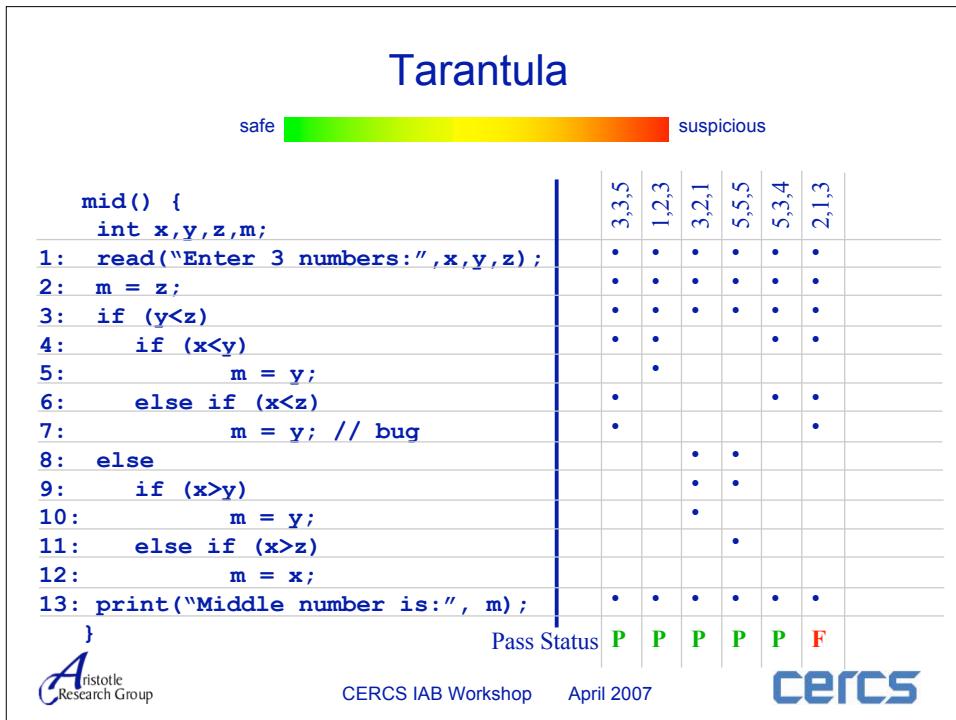
- statements executed
- outcome (pass/fail)

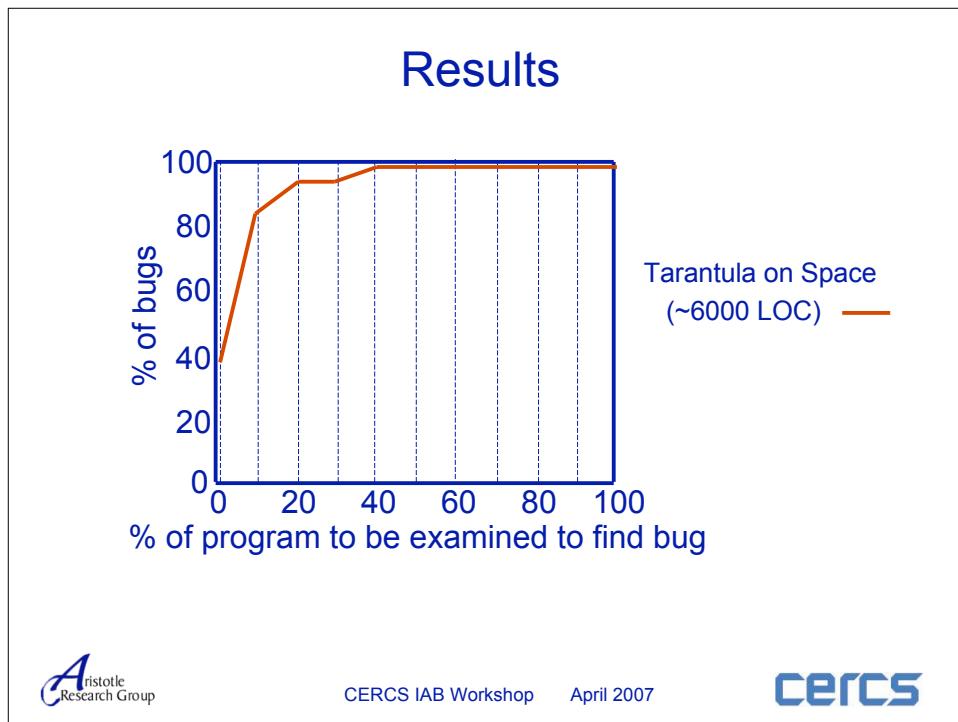
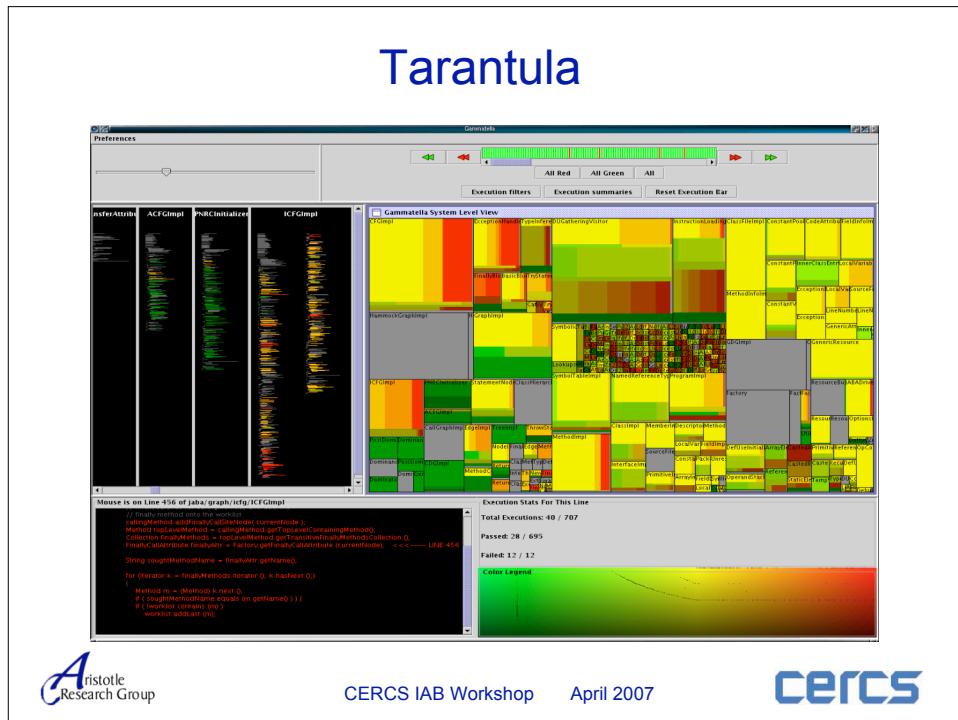
Intuition: Statements that are primarily executed by failed test cases are more suspicious than statements that are primarily executed by passed test cases

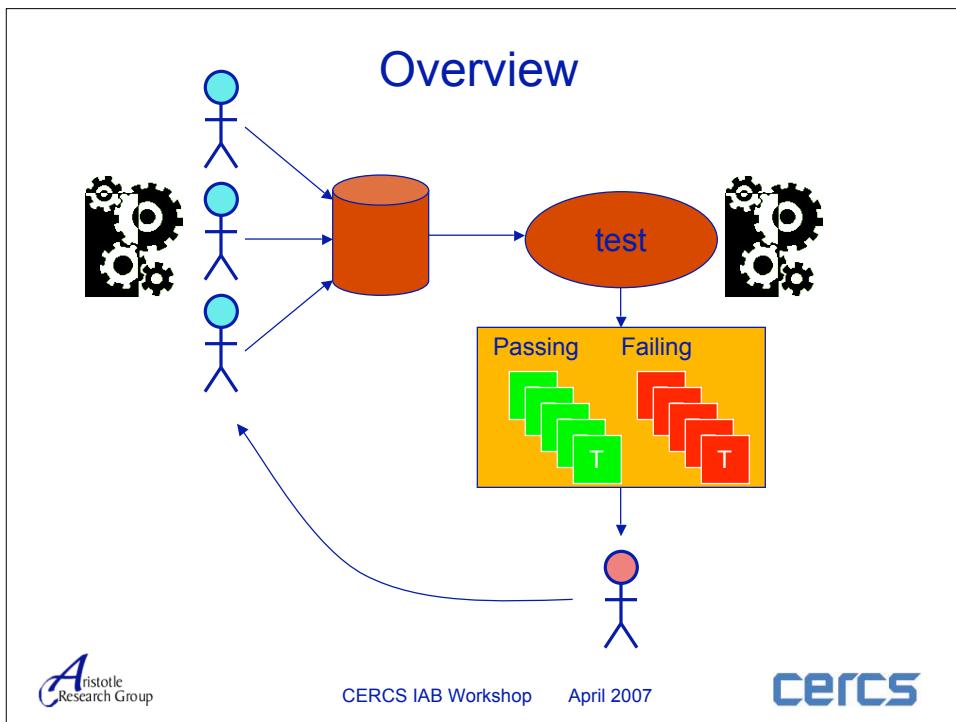
A
ristotle
Research Group

CERCs IAB Workshop April 2007









Tarantula: Fault Localization

	t1	t2	t3	t4	t5	t6	t7	t8	t9	t10
mid() {	3,3,5									
int x,y,z,m;	1,2,3									
1:read("Enter 3 integers:",x,y,z);	*	*	*	*	*	*	*	*	*	*
2:m = z;	*	*	*	*	*	*	*	*	*	*
3:if (y<z)	*	*	*	*	*	*	*	*	*	*
4: if (x<y)	*	*	*	*	*	*	*	*	*	*
5: m = y;	*									
6: else if (x<z)	*		*	*	*	*	*	*	*	*
7: m = y;	*		*		*		*		*	*
8:else	*	*	*	*	*	*	*	*	*	*
9: if (x>y)	*	*	*	*	*	*	*	*	*	*
10: m = z; //bug;correct m=y	*									
11: else if (x>z)	*									
12: m = x;										
13:print("Middle number is:", m);	*	*	*	*	*	*	*	*	*	*
}										
Pass/fail Status	P	P	P	P	P	F	F	F	F	F

Aristotle
Research Group

CERCS IAB Workshop April 2007

cercs

Tarantula: Fault Localization

	t1	t2	t3	t4	t5	t6	t7	t8	t9	t10
mid()	3,3,5									
int x,y,z,m;		1,2,3								
1:read("Enter 3 integers:",x,y,z);	*	*	*	*	*	*	*	*	*	*
2:m = z;	*	*	*	*	*	*	*	*	*	*
3:if (y<z)	*	*	*	*	*	*	*	*	*	*
4: if (x<y)	*	*		*	*	*	*	*	*	*
5: m = y;	*									
6: else if (x<z)	*			*	*	*	*	*	*	*
7: m = y;	*			*		*	*	*	*	*
8:else		*	*			*	*		*	*
9: if (x>y)		*	*			*	*		*	*
10: m = y; //fixed		*				*	*		*	*
11: else if (x>z)				*						
12: m = x;										
13:print("Middle number is:", m);	*	*	*	*	*	*	*	*	*	*
}										

Pass/fail Status P P P P P P P F P F

Aristotle
Research Group

CERCs IAB Workshop April 2007

cercs

Tarantula: Fault Localization

	t1	t2	t3	t4	t5	t6	t7	t8	t9	t10
mid()	3,3,5									
int x,y,z,m;		1,2,3								
1:read("Enter 3 integers:",x,y,z);	*	*	*	*	*	*	*	*	*	*
2:m = z;	*	*	*	*	*	*	*	*	*	*
3:if (y<z)	*	*	*	*	*	*	*	*	*	*
4: if (x<y)	*	*		*	*	*	*	*	*	*
5: m = y;	*									
6: else if (x<z)	*			*	*	*	*	*	*	*
7: m = y; //bug;correct:m=x;	*			*		*	*	*	*	*
8:else		*	*			*	*		*	*
9: if (x>y)		*	*			*	*		*	*
10: m = y; //fixed		*				*	*		*	*
11: else if (x>z)				*						
12: m = x;										
13:print("Middle number is:", m);	*	*	*	*	*	*	*	*	*	*
}										

Pass/fail Status P P P P P P P F P F

Aristotle
Research Group

CERCs IAB Workshop April 2007

cercs

Tarantula: Fault Localization

Aristotle
Research Group

CERCS IAB Workshop

April 2007

cercs

Debugging Process



- Are all failing tests caused by the same fault?
 - Can we associate groups of tests with different faults?
 - Can we search for different faults simultaneously?

Aristotle
Research Group

CERCS IAB Workshop April 2007



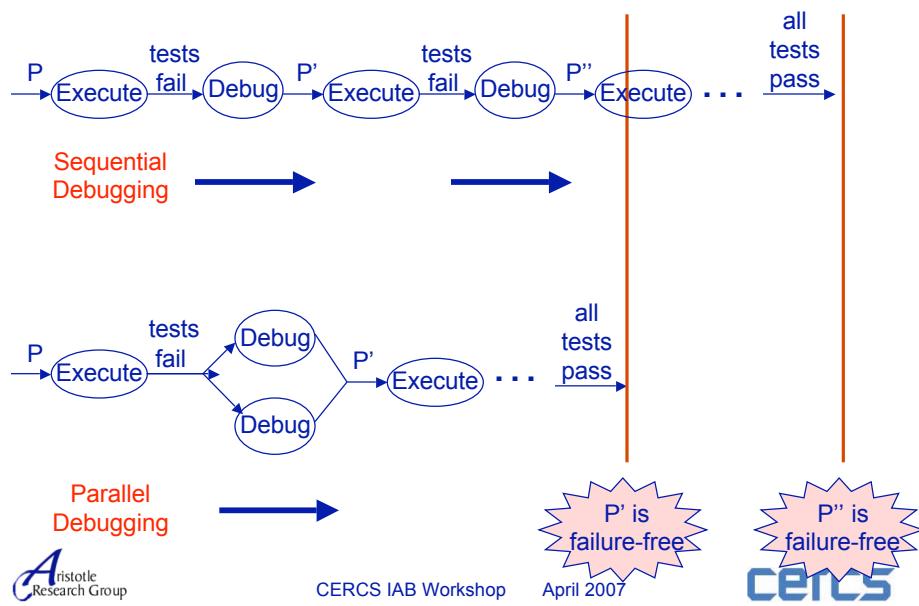
Tarantula: Fault Localization

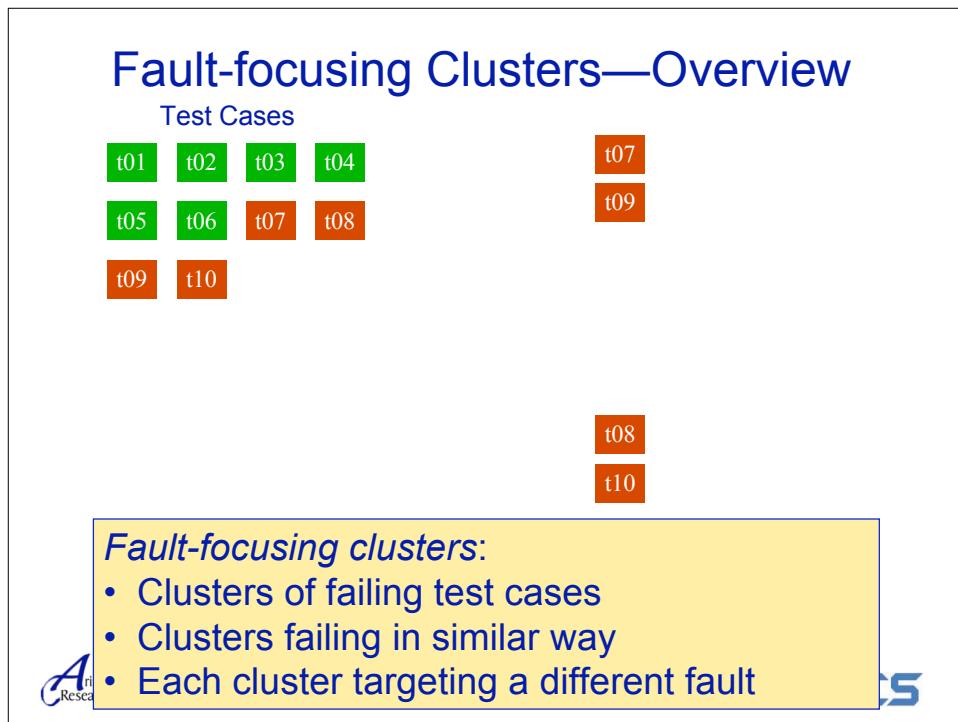
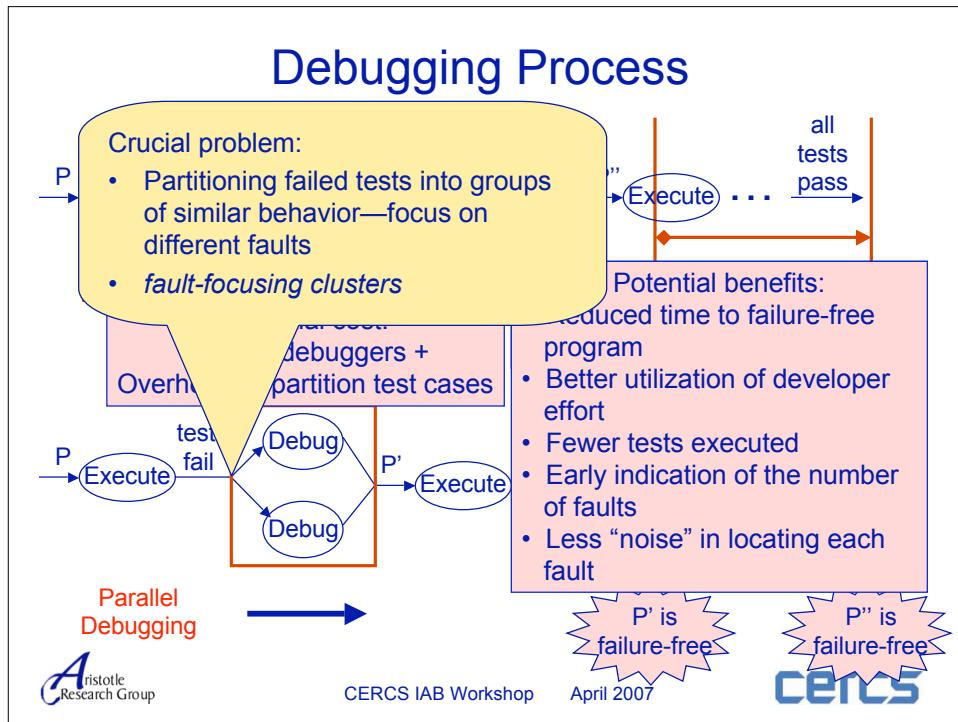
	t1	t2	t3	t4	t5	t6	t7	t8	t9	t10
mid() {	3,3,5	1,2,3	3,2,2	5,5,5	1,1,4	5,3,4	3,2,1	2,1,3	5,4,2	5,2,6
int x,y,z,m;
1:read("Enter 3 integers:",x,y,z);	*	*	*	*	*	*	*	*	*	*
2:m = z;	*	*	*	*	*	*	*	*	*	*
3:if (y<z)	*	*	*	*	*	*	*	*	*	*
4: if (x<y)	*	*	*	*	*	*	*	*	*	*
5: m = y;	*	*	*	*	*	*	*	*	*	*
6: else if (x<z)	*	*	*	*	*	*	*	*	*	*
7: m = y;	*	*	*	*	*	*	*	*	*	*
8:else	*	*	*	*	*	*	*	*	*	*
9: if (x>y)	*	*	*	*	*	*	*	*	*	*
10: m = z;	*	*	*	*	*	*	*	*	*	*
11: else if (x>z)	*	*	*	*	*	*	*	*	*	*
12: m = x;	*	*	*	*	*	*	*	*	*	*
13:print("Middle number is:", m);	*	*	*	*	*	*	*	*	*	*
}	*	*	*	*	*	*	*	*	*	*

Pass/fail Status P P P P P F F F F

Aristotle Research Group CERCS IAB Workshop April 2007 CERCS

Debugging Process





Specialized Test Suites

Specialized test suites:
Fault-focusing clusters
combined with
passing test cases

Give each specialized
test suite to different
developer (debugger)

t07
t09

Developer 1



t08
t10

Developer 2



CERCS IAB Workshop April 2007



Visualization of Each Result

```
mid() {  
    int x,y,z,m;  
1:read("Enter 3 integers:");  
2:m = z;  
3:if (y<z)  
4:    if (x<y)  
5:        m = y;  
6:    else if (x<z)  
7:        m = v; //bug  
8:else  
9:    if (x>y)  
10:       m = z; //bug  
11:    else if (x>z)  
12:       m = x;  
13:print("Middle number is:"...);  
}
```

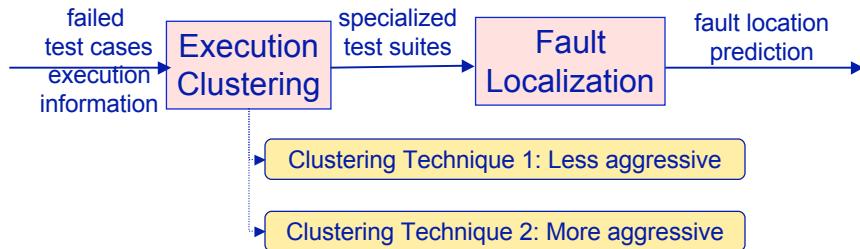
```
mid() {  
    int x,y,z,m;  
1:read("Enter 3 integers:");  
2:m = z;  
3:if (y<z)  
4:    if (x<y)  
5:        m = y;  
6:    else if (x<z)  
7:        m = y; //bug  
8:else  
9:    if (x>y)  
10:       m = z; //bug  
11:    else if (x>z)  
12:       m = x;  
13:print("Middle number is:"...);  
}
```



CERCS IAB Workshop April 2007



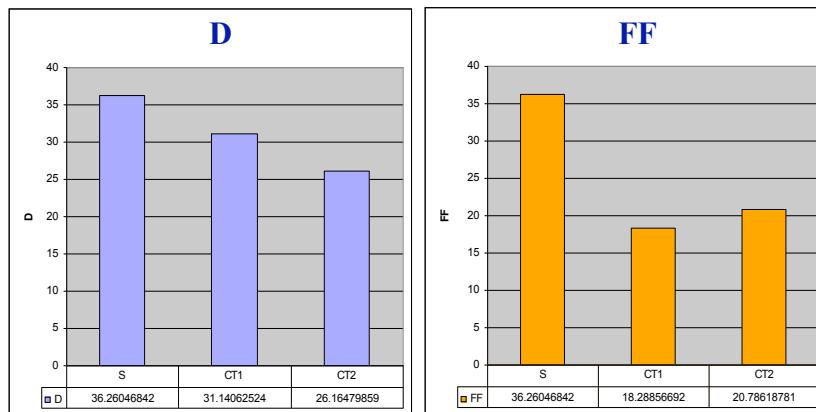
Fault-focusing Clusters



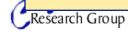
CERCs IAB Workshop April 2007



Summary of Results



- Parallel less expensive even if debugging sequential
- CT1 if time to delivery is most important (low salaries, high urgency)
- CT2 if developer cost is most important (high salaries, low urgency)



CERCs IAB Workshop April 2007



Conclusion

- Technique and visualization to provide support for finding faults
- New approach to debugging—parallelized debugging
- Technique to get fault-focused clusters to create specialized test suites
- Benefits, supported by studies
 - less time to failure-free program
 - better utilization of developer effort
 - less running of the test suite
 - indication of the number of faults
 - less “noise” in locating each fault



CERCs IAB Workshop April 2007



Tarantula: Fault Localization

- Jones, Harrold, Stasko, Workshop on Software Visualization 2001
- Jones, Harrold, Stasko, International Conference on Software Engineering 2002
- Jones, Orso, Harrold, Software Visualization 2003
- Jones, ICSE 2004 Doctoral Workshop
- Jones, Orso, Harrold, Journal of Information Visualization 2004
- Jones, Harrold, Automated Software Engineering 2005
- Jones, Bowring, Harrold, International Symposium on Software Testing and Analysis, 2007



CERCs IAB Workshop April 2007

