

All severity 3 bugs in *Testing Object-Oriented Systems: Models, Patterns and Tools* (as of December 20, 2002) are listed in this document.

The bug reports on this list include items found by the author and those submitted by Rana Ahmed, Antti Auer, Bill Bently, Prof. Lionel Briand, Dr. Arthur Carlson, Jeff Dever, Dion Dock, Ron Lusk, Patrick T. McGuigan, Norbert Mueller, Thomas Mueller, Prof. Arthur Reyes, Patrick Schoenbach, Peter Seibel, Serge Tanguay, Simon Watts, Michael Wells, and Russ Williams. Prof. Zdzislaw Ploski sent in an extensive list produced by painstaking effort. Thanks to all for your contributions.

The “sev” (severity) column designates the kind of bug:

- 1: Incorrect statement, concept, code syntax error, or other blunder.
- 2: Misspelling, misconstruction, or inconsistent usage.
- 3: Minor typo not likely to cause confusion or misunderstanding.

The “rev” (revision) column indicates the press run in which this bug is corrected. If blank, the bug correction hasn’t been published yet. The press run of your copy is on the bottom page iv (not numbered by convention.) If your copy is the 3<sup>rd</sup> or higher press run, items with rev=3 are corrected in your copy. Corrections for all other items will be published as soon as possible.

<b>Pg.</b>	<b>S e v</b>	<b>Version 1.0</b>	<b>Correction</b>	<b>R e v</b>
xxviii	3	Titles out of order 12.2 Kaynine policy 12.1 Dogpatch policy	12.1 Dogpatch policy 12.2 Kaynine policy	
xxx	3	Titles out of order 18.3. Simulation Oracle 18.2. Solved Example Oracle	18.2. Solved Example Oracle 18.3. Simulation Oracle	
49	3	Figure 3.3: Surprise	Surprises	
51	3	[Agrawal+89]	[Agarwal+89]	
52	3	[Morel+92]	[Morell+92]	
53	3	[Kaner+92]	[Kaner+93]	
62	3	[Kaner+92]	[Kaner+93]	
65	3	[Wilke 92]	[Wilke 93]	

<b>Pg.</b>	<b>S e v</b>	<b>Version 1.0</b>	<b>Correction</b>	<b>R e v</b>
70	3	[Jones 97]	[Jones 97a]	
71	3	[Stroustrup 92]	[Stroustrup 92b]	
71	3	add( <i>ix</i> ) is that <i>ix</i>	add( <i>x</i> ) is that <i>x</i>	
71	3	add(1) was	add(1) was	
72	3	[Stroustrup 88]	[Stroustrup 88]	
79	3	[Meyer 92] ... [Meyer 92]	[Meyer 92a] ... [Meyer 92a]	
84	3	[Meyer 92]	[Meyer 92a]	
97	3	[Harrold 92]	[Harrold+92]	
97	3	[Murphy 92]	[Murphy+92]	
97	3	[Skubics+96]	[Skubics+96]	
98	3	[D'Souza 94]	[D'Souza+94]	
101	3	[Harrold+92, 78-79]	[Harrold+92, 78-79]	
110	3	[D'Souza+98]	[D'Souza+99]	
113	3	[Weide 91]	[Weide+91]	
114	3	Bezier	Beizer	
151	3	[Devedas 97]	[Devadas 97]	
153	3	[Fujiwara+85, Abramovici+90, Devedas 97]	[Fujiwara 85, Abramovici+90, Devedas 97]	
156	3	Table 6.9 header: Action <i>S</i> ?	Action <i>S</i>	
157	3	Footnote: ... <i>i</i> = 1, <i>m</i> , and	<i>i</i> = (1, <i>m</i> ) and	
181	3	folllowing	following	
181	3	[Ward 85, Hatley 87]	[Ward+85, Hatley+87]	

<b>Pg.</b>	<b>S e v</b>	<b>Version 1.0</b>	<b>Correction</b>	<b>R e v</b>
184	3	[Harel 97]	[Harel+97]	
185	3	[Allen 95]	[Allen+95]	
186	3	[Fujiwara 91]	[Fujiwara+91]	
186	3	[Ward 85]	[Ward+85]	
186	3	[Sidhu 89]	[Sidhu+89]	
186	3	[Hopcroft 79]	[Hopcroft+79]	
186	3	[Devedas+94]	[Devadas+94]	
193	3	[Hopcroft 79]	[Hopcroft+79]	
201	3	[Allen 95]	[Allen+95]	
201	3	[Hong 95b]	[Hong+95b]	
201	3	[Harel 97]	[Harel+97]	
202	3	[Harel 97]	[Harel+97]	
203	3	[Coplien 95] ... [Coplien 95]	[Coplien+95] ... [Coplien+95]	
204	3	[Schuman 87]	[Schuman+87]	
205	3	In footnote: [Harel 97]	[Harel+97]	
212	3	(Footnote) relying on the externally visible actions or a Moore machine	relying on the externally visible actions of a Moore machine	3
222	3	[Lewis 81]	[Lewis+81]	
222	3	[Hopcroft 79]	[Hopcroft+79]	
228	3	[Weide 91]	[Weide+91]	
229	3	with only 5 states, 2 events and 2 states, there 10 billion	with only 5 states, 2 events and 2 states, there are 10 billion	3

<b>Pg.</b>	<b>S e v</b>	<b>Version 1.0</b>	<b>Correction</b>	<b>R e v</b>
238	3	[Hopcroft 79]	[Hopcroft+79]	
238	3	Table 7.5 State Model Checklist, Part 1: Structure	Table 7.5 State Model Checklist: Structure	
239	3	Table 7.6 State Model Checklist, Part 3: State Names	Table 7.5 State Model Checklist: State Names	
239	3	Table 7.6, items in checklist two start with 1	All items in checklist two should start with number 2	3
240	3	Table 7.7, item 3.4: that is, i.e.,	that is,	
241	3	Table 7.9, Items in checklist five start with 4. Note refers to incorrect numbers.	All items in checklist five should start with number 5. Note should refer to items "5.6 through 5.9"	3
246	3	Upper right cell of table: Started ?	? (delete Started)	3
247	3	Figure 7.37: key missing.	Repeat key given in Figure 7.26	
258	3	Confidence that no incorrect	Determining that no incorrect	3
264	3	Table 7.40, row 8: resultant State	resultant state	
266	3	[Järvinen 90]	[Järvinen+90]	
267	3	[Harel 97]	[Harel+97]	
267	3	[Fowler 97]	[Fowler+97]	
267	3	[Wasserman+85]	[Wasserman 85]	
268	3	[Drusinsky 94]	[Drusinsky+94]	
268	3	[Devadas 94]	[Devadas+94]	
276	3	[Jacobson 95a]	[Jacobson 95]	
290	3	Figure 8.6: Retain	retain	
291	3	Table 8.6: Narrow Rectangle	Narrow rectangle	
305	3	Table 8.12: thst	that	

<b>Pg.</b>	<b>S e v</b>	<b>Version 1.0</b>	<b>Correction</b>	<b>R e v</b>
309	3	directed arrows	arrows	
322	3	[Wirfs-Brock 90]	[Wirfs-Brock+90]	
336	3	[Horgan 96]	[Horgan+96]	
338	3	Table 9.1, incorrect entry: Transitive Operations Design a test suite for methods that are transitive.	Delete this entry.	
340	3	... review the entry criteria	... review the Entry Criteria	
342	3	Figure 9.4, Name-Type-Value entries under Returned Objects not in same order as others.	Revise order.	
346	3	[Juttner+94]	[Jüttner+94]	
349	3	Table 10.1: Stratgey	Strategy	
349	3	Table 10.1, last row, integration coverage model: Flattened class free graph	Flattened class FREE graph	
351	3	4. Select test design patterns for each method: Category-Partition, Combinatorial	4. Select test design patterns for each method: Category-Partition, Combinatorial	3
353	3	[Skublics+96]	[Skubics+96]	
353	3	... illustrate the importance of adequate class scope testing. <end page 354> TRACS is a network management system used to monitor traffic and out ages on telephone central office systems connected by trunk lines [Murphy+94].	... illustrate the importance of adequate class scope testing [Murphy+94]. <end page 354> TRACS is a network management system used to monitor traffic and out ages on telephone central office systems connected by trunk lines.	
377	3	[Li 93] ... [Lejter 92, Li 93, Wilde 93, Lorenz 94]	[Li+93] ... [Lejter+92, Li+93, Wilde+93, Lorenz+94]	
377	3	[Gill 91]	[Gill+91]	
377	3	[Bertolino 97]	[Bertolino+97]	
377	3	[Prather 87]	[Prather+87]	

<b>Pg.</b>	<b>S e v</b>	<b>Version 1.0</b>	<b>Correction</b>	<b>R e v</b>
378	3	[Bertolino 97]	[Bertolino+97]	
381	3	[Frankl 88]	[Frankl+88]	
384	3	[Ashcroft 71]	[Ashcroft+71]	
390	3	Procedure 5: For each D action	For each <b>D</b> action	
399	3	[Prather 87]	[Prather+87]	
399	3	<pre>{     // ...     case RAISE_LANDING_GEAR:         landing_gear_motor(lgm_turn_on_until_raised);         break;     // ... }</pre>	<pre>{     // ...     case RAISE_LANDING_GEAR:         landing_gear_motor(lgm_turn_on_until_raised);         break;     // ... }</pre>	
429	3	Test Hamess	test harness	
443	3	[McCabe 94]	[McCabe+94]	
460	3	unshuffledList	UnshuffledList	
465	3	[Smith 92]	[Smith+92]	
465	3	[Zweben 92]	[Zweben+92]	
467	3	Intset indentation incorrect.	Align all identations.	
472	3	Figure 10.29, some guard operators not surrounded with spaces.	Place leading and trailing space on each guard operator.	
479	3	(2) missing or incorrect action	(2) missing or incorrect actions	
480	3	[Hoffman 95]	[Hoffman+95]	
481	3	[Jacobsen+92]	[Jacobson+92]	
488	3	Table 10.30, incorrect header: Nest State	Next State	3

<b>Pg.</b>	<b>S e v</b>	<b>Version 1.0</b>	<b>Correction</b>	<b>R e v</b>
488	3	Table 10.30, last entry in Message column inconsistent with Figure 10.33: Closed	close	
490	3	Figure 10.34, some guard operators not surrounded with spaces.	Place leading and trailing space on each guard operator.	
491	3	done by the server	done by servers	
496	3	[Hoffman 95]	[Hoffman+95]	
496	3	[Kiriani 94]	[Kiriani+94]	
504	3	[Lieberherr 92]	[Lieberherr+92]	
505	3	[Perry 90]	[Perry+90]	
508	3	Figure 10.40, upward arrows used to indicate inheritance.	Use UML triangles to indicate inheritance.	
511	3	the subclass class test suite	the subclass test suite	
522	3	[Adrion 82]	[Adrion+82]	
522	3	[Harrold 91]	[Harrold+91]	
523	3	[Chang 96]	[Chang+96]	
532	3	[Cox 91]	[Cox+91]	
544	3	For example;	For example:	
568	3	Hw	hardware	
569	3	Relationships occur in many forms, including association, aggregation, whole-part, is-a, subtype, aggregation, and recursive inclusion.	Relationships occur in many forms, including association, aggregation, whole-part, is-a, subtype, and recursive inclusion.	3
570	3	The person-owns-dog requirement can be implemented with at least four different collaborations.	The person-owns-dog requirement can be implemented with at least four different collaborations [Page-Jones 95].	3
582	3	[Lange 97]	[Lange+97]	

Pg.	S e v	Version 1.0	Correction	R e v
611	3	Figure 12.20, inconsistent spaces in guard operators	All operators have leading and trailing space.	
623	3	[Selic 94]	[Selic+94]	
624	3	[Lange 97]	[Lange+97]	
658	3	[Cheatham 90]	[Cheatham+90]	
663	3	represetned	represented	
666	3	Because, many stubs	Because many stubs	
666	3	bnoteen	not been	3
676	3	In response, some sequences of messages are generated by objects.	Objects generate a message sequence in response.	
679	3	stragety	strategy	
714	3	Marick argues that testing should be done at (small) subsystem scope, implying subsystem big bang integration followed by subsystem integration [Marick 95].	Marick argues that testing should be done at (small) subsystem scope, implying big bang integration of subsystem components [Marick 95].	3
762	3	generates an exception, or causes B to violate the contract of another baseline component C.	generates an exception, or (2) causes B to violate the contract of another baseline component C.	
775	3	Retest All has ...	<i>Retest All</i> has ...	
778	3	Retest Risky Use Cases has ...	<i>Retest Risky Use Cases</i> has ...	
784	3	Table 15.4, extra comma: b8,	b8	
788	3	Retest Changed Code has ...	<i>Retest Changed Code</i> has ...	
795	3	See Table 15.2 (page 775) Retest Within Firewall	See Table 15.2 (page 775). <i>Retest Within Firewall</i>	
807	3	class/scope	class scope	
814	3	assertions are a practical	Assertions are a practical	

Pg.	S e v	Version 1.0	Correction	R e v
827	3	Jezequel presents ... [Jezequel 96].	Jézéquel presents ... [Jézéquel 96].	
828	3	for all i	for all i	
829	3	assert (	assert(	
830	3	float sqrt(float x);	float sqrt(float x)	
840	3	Check superclass sequential constraints using Percolation.	Check superclass sequential constraints using <i>Percolation</i> .	
875	3	<i>Protocols</i> are a kind of abstract method (interface only)	A <code>protocol</code> is a kind of abstract method (interface only)	
875	3	A <i>category</i> is a named collection of method definitions that is similar to aspects of multiple inheritance and the C++ friend.	A <code>category</code> is a named collection of method definitions combining capabilities similar to multiple inheritance and the C++ friend class.	
880	3	An error message is displayed in Window.	An error message is displayed.	
895	3	[Leavens, Cheon+94]	[Cheon+94]	
895	3	and Objective-C: Upon	and Objective-C: upon	
895	3	Percolation would be supported in the C++ extensions proposed by Porat and Fertig	Percolation is supported in the C++ extensions developed by Porat and Fertig	3
914	3	[Schulman+87]	[Schuman+87]	
914	3	[Walden 95]	[Walden+95]	
915	3	[Campione 96]	[Campione+96]	
935	3	[Karani 94]	[Karani+94]	
936	3	data value	data structure	
936	3	individual numbers of the identifier	individual digits of the identifier	
939	3	[Korel 96]	[Korel+96]	
953	3	in this expression	in these expressions	

Pg.	S e v	Version 1.0	Correction	R e v
954	3	[Howden 78]	[Howden+78]	
955	3	[Richardson 92]	[Richardson+92]	
963	3	exclusively controlled by the OUT.	exclusively controlled by the IUT.	
964	3	sucessor tests	successor tests	3
975	3	test case cases	test cases	
992	3	Table 19.4: Envy	ENVY	
994	3	TestDriver version CPU ... TestDriver version CPU	TestDriver version, CPU ... TestDriver version, CPU	
1042	3	// Test code follows--e.g., and instance	// Test code follows- e.g., an instance	
1058	3	recursively the state set	recursively set the state	3
1058	3	global and class variable	global and class variables	
1068	3	line item	line-item	
1086	3	(2) In many GUI environments	(3) In many GUI environments	
1106	3	<b>requirement.</b> Any capability, or feature, function	<b>requirement.</b> Any capability, feature, or function	
1107	3	<b>runtime stack.</b> ... <b>runtime.</b> ...	<b>runtime.</b> ... <b>runtime stack.</b> ...	
1113	3	<b>test harness.</b> A system of test drivers and other tools to support test execution (e.g., ...).	<b>test harness.</b> A system of test drivers and other tools that support test execution.	
1131	3	[Kan+95]	[Kan 95]	