David Binkley

List of Publications by Year
in descending order

73 papers	3,327 citations	331670 21 h-index	243625 44 g-index
75 all docs	75 docs citations	75 times ranked	1061 citing authors

#	Article	IF	Citations
1	Featherweight assisted vulnerability discovery. Information and Software Technology, 2022, 146, 106844.	4.4	0
2	Web Service Slicing: Intra and Inter-Operational Analysis to Test Changes. IEEE Transactions on Services Computing, 2021, 14, 930-943.	4.6	7
3	Service Evolution Analytics: Change and Evolution Mining of a Distributed System. IEEE Transactions on Engineering Management, 2021, 68, 137-148.	3.5	7
4	From Neuron Coverage to Steering Angle: Testing Autonomous Vehicles Effectively. Computer, 2021, 54, 77-85.	1.1	5
5	Observation-based approximate dependency modeling and its use for program slicing. Journal of Systems and Software, 2021, 179, 110988.	4.5	3
6	QSES: Quasi-Static Executable Slices. , 2021, , .		3
7	Evaluating lexical approximation of program dependence. Journal of Systems and Software, 2020, 160, 110459.	4.5	3
8	On Adaptive Change Recommendation. Journal of Systems and Software, 2020, 164, 110550.	4.5	3
9	An Exploratory Study of the Relationship Between Software Test Smells and Fault-Proneness. IEEE Access, 2019, 7, 139526-139536.	4.2	16
10	A comparison of tree- and line-oriented observational slicing. Empirical Software Engineering, 2019, 24, 3077-3113.	3.9	13
11	MOAD: Modeling Observation-Based Approximate Dependency. , 2019, , .		3
12	What are the effects of history length and age on mining software change impact?. Empirical Software Engineering, 2018, 23, 2362-2397.	3.9	7
13	The need for software specific natural language techniques. Empirical Software Engineering, 2018, 23, 2398-2425.	3.9	13
14	Aggregating Association Rules to Improve Change Recommendation. Empirical Software Engineering, 2018, 23, 987-1035.	3.9	21
15	[Research Paper] The Case for Adaptive Change Recommendation. , 2018, , .		0
16	Observational slicing based on visual semantics. Journal of Systems and Software, 2017, 129, 60-78.	4.5	6
17	Entropy as a lens into LDA model understanding. , 2017, , .		0
18	Tree-Oriented vs. Line-Oriented Observation-Based Slicing. , 2017, , .		10

#	Article	IF	Citations
19	Generalized observational slicing for tree-represented modelling languages. , 2017, , .		14
20	A Case for Software Specific Natural Language Techniques. , 2016, , .		1
21	Practical guidelines for change recommendation using association rule mining. , 2016, , .		14
22	PORBS: A parallel observation-based slicer. , 2016, , .		7
23	Source code analysis with LDA. Journal of Software: Evolution and Process, 2016, 28, 893-920.	1.6	4
24	Improving change recommendation using aggregated association rules. , 2016, , .		7
25	ORBS and the limits of static slicing. , 2015, , .		25
26	The impact of vocabulary normalization. Journal of Software: Evolution and Process, 2015, 27, 255-273.	1.6	4
27	Enabling improved IR-based feature location. Journal of Systems and Software, 2015, 101, 30-42.	4.5	17
28	Are test smells really harmful? An empirical study. Empirical Software Engineering, 2015, 20, 1052-1094.	3.9	135
29	Seeing Is Slicing: Observation Based Slicing of Picture Description Languages. , 2014, , .		10
30	ORBS: language-independent program slicing. , 2014, , .		57
31	An empirical study of identifier splitting techniques. Empirical Software Engineering, 2014, 19, 1754-1780.	3.9	41
32	Recovering test-to-code traceability using slicing and textual analysis. Journal of Systems and Software, 2014, 88, 147-168.	4.5	47
33	Coherent clusters in source code. Journal of Systems and Software, 2014, 88, 1-24.	4.5	10
34	The impact of identifier style on effort and comprehension. Empirical Software Engineering, 2013, 18, 219-276.	3.9	106
35	Vocabulary normalization improves IR-based concept location. , 2012, , .		7
36	An empirical analysis of the distribution of unit test smells and their impact on software maintenance. , 2012, , .		104

#	Article	IF	Citations
37	SCOTCH: Test-to-code traceability using slicing and conceptual coupling. , 2011, , .		28
38	Expanding identifiers to normalize source code vocabulary. , 2011, , .		54
39	Assessing the impact of global variables on program dependence and dependence clusters. Journal of Systems and Software, 2010, 83, 96-107.	4.5	20
40	A trajectory-based strict semantics for program slicing. Theoretical Computer Science, 2010, 411, 1372-1386.	0.9	19
41	Normalizing Source Code Vocabulary. , 2010, , .		60
42	To camelcase or under_score. , 2009, , .		75
43	Dependence clusters in source code. ACM Transactions on Programming Languages and Systems, 2009, 32, 1-33.	2.1	39
44	Increasing diversity: Natural language measures for software fault prediction. Journal of Systems and Software, 2009, 82, 1793-1803.	4.5	22
45	Identifier length and limited programmer memory. Science of Computer Programming, 2009, 74, 430-445.	1.9	29
46	Using peer-led team learning to increase participation and success of under-represented groups in introductory computer science. SIGCSE Bulletin, 2009, 41, 163-167.	0.1	37
47	An empirical study of the relationship between the concepts expressed in source code and dependence. Journal of Systems and Software, 2008, 81, 2287-2298.	4.5	10
48	Impact of Limited Memory Resources. , 2008, , .		6
49	Program slicing. , 2008, , .		29
50	Dependence Anti Patterns. , 2008, , .		16
51	An empirical study of static program slice size. ACM Transactions on Software Engineering and Methodology, 2007, 16, 8.	6.0	71
52	Guest Editors' Introduction to the Special Section from the International Conference on Software Maintenance and Evolution. IEEE Transactions on Software Engineering, 2007, 33, 797-798.	5.6	0
53	Source Code Analysis: A Road Map. , 2007, , .		122
54	An empirical study of rules for well-formed identifiers. Journal of Software: Evolution and Process, 2007, 19, 205-229.	1.1	21

#	Article	IF	Citations
55	Quantifying identifier quality: an analysis of trends. Empirical Software Engineering, 2007, 12, 359-388.	3.9	48
56	Effective identifier names for comprehension and memory. Innovations in Systems and Software Engineering, 2007, 3, 303-318.	2.1	83
57	Software Fault Prediction using Language Processing. , 2007, , .		1
58	Characterising, Explaining, and Exploiting the Approximate Nature of Static Analysis through Animation. , 2006, , .		4
59	Stop-List Slicing. , 2006, , .		6
60	Syntactic Identifier Conciseness and Consistency. , 2006, , .		40
61	Theory and algorithms for slicing unstructured programs. Information and Software Technology, 2006, 48, 549-565.	4.4	17
62	A formalisation of the relationship between forms of program slicing. Science of Computer Programming, 2006, 62, 228-252.	1.9	32
63	Theoretical foundations of dynamic program slicing. Theoretical Computer Science, 2006, 360, 23-41.	0.9	35
64	A Survey of Empirical Results on Program Slicing. Advances in Computers, 2004, 62, 105-178.	1.6	100
65	Syntax-Directed Amorphous Slicing. Automated Software Engineering, 2004, 11, 27-61.	2.9	15
66	Interprocedural slicing using dependence graphs. ACM SIGPLAN Notices, 2004, 39, 229-243.	0.2	77
67	Evolutionary testing in the presence of loop-assigned flags. Software Engineering Notes: an Informal Newsletter of the Special Interest Committee on Software Engineering / ACM, 2004, 29, 108-118.	0.7	17
68	Amorphous program slicing. Journal of Systems and Software, 2003, 68, 45-64.	4.5	83
69	Flow insensitive points-to sets. Information and Software Technology, 2002, 44, 743-754.	4.4	6
70	The application of program slicing to regression testing. Information and Software Technology, 1998, 40, 583-594.	4.4	91
71	Program integration for languages with procedure calls. ACM Transactions on Software Engineering and Methodology, 1995, 4, 3-35.	6.0	100
72	Precise executable interprocedural slices. ACM Transactions on Programming Languages and Systems, 1993, 2, 31-45.	1.5	50

#	Article	IF	Citations
73	Interprocedural slicing using dependence graphs. ACM Transactions on Programming Languages and Systems, 1990, 12, 26-60.	2.1	1,134