

Nagappan: Mining metrics x

https://scholar.google.com/scholar?cites=15531...

Web Images More... tim.menzies@gmail.com

Google

Scholar About 527 results (0.02 sec) Any time 8

Mining metrics to predict component failures

☐ Search within citing articles

Predicting defects for eclipse queensu.ca [PDF]
T Zimmermann, R Premraj... - Predictor Models in ..., 2007 - ieeexplore.ieee.org
Abstract We have mapped defects from the bug database of Eclipse (one of the largest open-source projects) to source code locations. The resulting data set lists the number of pre-and post-release defects for every package and file in the Eclipse releases 2.0, 2.1, and 3.0. ...
Cited by 398 Related articles All 26 versions Cite Save

Predicting defects using network analysis on dependency graphs thomas-zimmermann.com [PDF]
T Zimmermann, N Nagappan - ... of the 30th international conference on ..., 2008 - dl.acm.org
Abstract In software development, resources for quality assurance are limited by time and by cost. In order to allocate resources effectively, managers need to rely on their experience backed by code complexity metrics. But often dependencies exist between various pieces ...
Cited by 299 Related articles All 9 versions Cite Save

A survey and taxonomy of approaches for mining software repositories in the context of software evolution researchgate.net [PDF]
H Kagdi, ML Collard, JI Maletic - Journal of Software ..., 2007 - Wiley Online Library
Abstract A comprehensive literature survey on approaches for mining software repositories (MSR) in the context of software evolution is presented. In particular, this survey deals with those investigations that examine multiple versions of software artifacts or other temporal ...
Cited by 299 Related articles All 9 versions Cite Save

Predicting faults from cached history ust.hk [PDF]
S Kim, T Zimmermann, EJ Whitehead Jr... - Proceedings of the 29th ..., 2007 - dl.acm.org
Abstract We analyze the version history of 7 software systems to predict the most fault prone entities and files. The basic assumption is that faults do not occur in isolation, but rather in bursts of several related faults. Therefore, we cache locations that are likely to have faults: ...
Cited by 252 Related articles All 26 versions Cite Save

[BOOK] Introduction and roadmap: History and challenges of software evolution ieee.org [PDF]
T Mens - 2008 - Springer
Abstract The ability to evolve software rapidly and reliably is a major challenge for software engineering. In this introductory chapter we start with a historic overview of the research domain of software evolution. Next, we briefly introduce the important research themes in ...
Cited by 245 Related articles All 22 versions Cite Save More