Patents by Examiner Adam Conkey

Management of template versions

Patent number: 8689176

Abstract: A computer implemented method, system, and/or computer program product modifies a master template for an executable document. A version architecture comprises a master template for a master document, a new variation document that describes what changes are to be made to the master template for a specific context, and a resulting modified template based on the master template and rule-based instructions found in the new variation document. A processor defines an executable master document from the master template that, when executed, produces a final product. Specific rules set out conditions for modifying the master template for the specific context. These specific rules are used to define a new variation document, which is applied against the master template to generate a context-specific modified template. The context-specific modified template is then used to generate a context-specific executable document that, when executed, generates a final context-specific product.

Type: Grant

Filed: May 11, 2011

Date of Patent: April 1, 2014

Assignee: International Business Machines Corporation

Inventors: Saeed Bagheri, Jarir K. Chaar, Yi-Min Chee, Feng Liu, Daniel V. Oppenheim, Krishna C. Ratakonda

Natural language assertion processor

Patent number: 8689192

Abstract: A method of processing natural language assertions (NLAs) can include identifying an NLA and then translating that NLA into a verification language assertion (VLA) using a natural language parser (NLP) and synthesis techniques. This VLA can be translated into an interpreted NLA (NLA*) using a VLA parser and pattern matching techniques. At this point, the process can allow user review of the NLA* and the NLA. When the user determines that the NLA* and the NLA are the same or have insignificant difference, then verification can be performed using the VLA. The results of the verification can then be back annotated on the NLA. In one fully-automatic embodiment, in addition to comparing the NLA and the NLA*, the VLA and a VLA* (generated from the NLA*) can be compared, thereby providing yet another test of accuracy for the user during verification.

Type: Grant

Filed: January 12, 2009

Date of Patent: April 1, 2014

Assignee: Synopsys, Inc.

Inventor: Alain M. Dargelas

Dynamic interface reduction for software model checking

Patent number: 8671396

Abstract: Architecture employs an iterative process that incrementally discovers inter-component interactions and explores local state spaces within each component. Thus, the architecture lazily constructs the behavior of the environment of a component in the target software system, and integrates the construction of the inter-component interactions with the model checking process itself, and hence, does not need to eagerly construct the interface process. Component-based state space reduction is applied during the exploration of the whole system. The architecture decomposes a target software system into a set of loosely coupled components where interactions between the components tend to be significantly simpler than interactions within each component. An iterative algorithm facilitates the component-based state space reduction, which is exponential, on the real large-scale software systems.

Type: Grant

Filed: May 30, 2011

Date of Patent: March 11, 2014

Assignee: Microsoft Corporation

Inventors: Ming Wu, Huayang Guo, Yi Yang, Gang Hu, Lintao Zhang, Lidong Zhou, Tisheng Chen

Mobile specialized software code updated

Patent number: 8667479

Abstract: A first host (60), computer readable medium and method for providing a specialized software package (24) to a second host (70).

Type: Grant

Filed: January 19, 2009

Date of Patent: March 4, 2014

Assignee: Telefonaktiebolaget L M Ericsson (Publ)

Inventors: Andreas Johnsson, Ayodele Damola

Managed enterprise software components as dynamic services

Patent number: 8661410

Abstract: A computer-implemented method, system, and medium is disclosed for implementing a hybrid application server. The server is configured to determine that a bundle deployed onto a dynamic module system defines a component that is manageable by an application server component container. In response to the determining, the application server deploys the component onto the component container and creates an instance of the component in the container so that the container can manage the component. Finally, the application server registers a proxy handle to the component as a service in the dynamic module system. Thereafter, the proxy handle is discoverable and invocable by another component.

Type: Grant

Filed: March 11, 2011

Date of Patent: February 25, 2014

Assignee: Oracle International Corporation

Inventors: Sanjeeb Sahoo, Sivakumar Melapannai Thyagarajan

Method and system for creating OWL ontology from java

Patent number: 8656356

Abstract: A method, system, and computer program product are disclosed for creating an OWL ontology from a Java source code, wherein the Java source code includes a plurality of Java classes. Each of these Java classes includes one or more Java members and one or more Java objects. The Java objects are used to instantiate each of the Java classes. An OWL class is created for each of the Java classes, and an OWL concept is created for each of the Java members. Further, an OWL instance is created for each of the Java objects. Each of the OWL classes, the OWL concepts, and the OWL instances are assigned data type properties and object properties. Subsequently, each of the OWL classes and relations between them, each of the OWL concepts, and each of the OWL instances are included to form the OWL ontology.

Type: Grant

Filed: March 14, 2011

Date of Patent: February 18, 2014

Assignee: Infosys Limited

Inventors: Suman Roy, Ming Fei Yan

Lossless path reduction for efficient symbolic execution and automatic test generation

Patent number: 8645924

Abstract: In one embodiment, symbolically executing a software module having a number of execution paths; and losslessly reducing the number of execution paths during the symbolic execution of the software module.

Type: Grant

Filed: June 6, 2011

Date of Patent: February 4, 2014

Assignee: Fujitsu Limited

Inventors: Guodong Li, Sreeranga P. Rajan, Indradeep Ghosh

System and method for remotely compiling multi-platform native applications for mobile devices

Patent number: 8612947

Abstract: A computer readable medium comprises executable instructions to: provide an SDK to a client computer comprising executable instructions for communicating with a build server, receive an HTML/Javascript source application and a configuration

file referencing one or more source application files over a computer network from a client computer to the build server, transmit the HTML/Javascript source application and configuration file to multiple compile servers corresponding to each of multiple mobile device platforms, combine the HTML/Javascript source application with a mobile device platform specific framework source code for each mobile device platform on each compile server, compile the HTML/Javascript source application and framework source code on the compile server to output an executable native application for each mobile device platform, and transmit each executable native application from the compile server to the client computer over a computer network.

Type: Grant
Filed: July 26, 2010
Date of Patent: December 17, 2013
Assignee: Adobe Systems Canada Inc.
Inventors: Brian LeRoux, James Douma, Andre Charland, Dave Johnson, Brock Whitten, Robert Ellis, Joey Bowser

Tracking stale comments in source code listings

Patent number: 8607193
Abstract: Managing comments within computer software source code by detecting a change in a portion of computer software source code, identifying a comment that is associated with the portion, and providing an indication that the comment was not changed subsequent to the portion being changed.

Type: Grant
Filed: January 16, 2012
Date of Patent: December 10, 2013
Assignee: International Business Machines Corporation
Inventors: Shlomit Avrahami, Joshua Fox, Yuri Kosharovsky, Michael Pelts

Profile driven multicore background compilation

Patent number: 8543991
Abstract: Idle processor cores can be used to compile methods that are likely to be executed by a program based on profile data that is captured during one or more previous executions. Methods that are determined by the profile data to be likely to be used can be compiled eagerly on one or more background threads. Transparency can be achieved by ensuring that module load order is not altered because of the background threads by recording the state of loaded modules after each profiled compilation, persisting that data, and waiting to eagerly compile a method until the method to be compiled and all its dependencies has been loaded by the executing program.

Type: Grant
Filed: March 10, 2011
Date of Patent: September 24, 2013
Assignee: Microsoft Corporation
Inventors: Subramanian Ramaswamy, David Hiniker-Roosa, Feng Yuan, Sedar Gokbulut, Ashok C. Kamath, Jan Kotas, Vance P. Morrison

Processing apparatus, trace unit and diagnostic apparatus

Patent number: 8533685
Abstract: A processing circuit 4 is responsive to at least one conditional instruction to perform a conditional operation in dependence on a current value of a subset of at least one condition flag 22. A trace circuit 6 is provided for generating trace data elements indicative of operations performed by the processing circuit 4. When the processing circuit 4 processes at least one selected instruction, then the trace circuit 6 generates a trace data element including a traced condition value indicating at least the subset of condition flags 22 required to determine the outcome of the conditional instruction. A corresponding diagnostic apparatus 12 uses the traced condition value to determine a processing outcome of the at least one conditional instruction.

Type: Grant
Filed: October 26, 2011
Date of Patent: September 10, 2013
Assignee: ARM Limited
Inventors: John Michael Horley, Simon John Craske, Michael John Gibbs, Paul Anthony Gilkerson

Context-sensitive analysis framework using value flows

Patent number: 8516443

Abstract: In general, in one aspect, the invention relates to a method for identifying program properties in source code. The method includes: identifying a set of program objects corresponding to a potential program property within the source code; identifying a function associated with the set of program objects in the source code; identifying an input and an output of the function in the source code; creating a value flow graph by: creating a function input node representing the input; creating a function output node representing the output; identifying value flow functions, where each value flow function defines a conditional dependency between a pair of graph nodes, and creating, based on the value flow functions, directed graph edges modeling a potential flow of data through the value flow graph; and identifying a feasible path in the value flow graph depicting an occurrence of the potential program property in the source code.
Type: Grant

Filed: May 26, 2011

Date of Patent: August 20, 2013

Assignee: Oracle International Corporation

Inventors: Lian Li, Cristina N. Cifuentes, Nathan Robert Albert Keynes

Execution time estimation method and device

Patent number: 8448140

Abstract: An execution time estimation device includes a program partitioning section that extracts partial programs partitioned by a conditional branch instruction or a function call instruction from a target program, a partial program execution time estimation calculating section that calculates the execution time of each of the partial programs to associate the leading instruction and the end instruction of each of the partial programs, and the calculated execution time with one another, a branch history information generating section that generates a branch history bit sequence which is a sequence of the true-false of the conditional branch instruction of when the target program is executed, an execution trace reproducing section that generates the execution sequences of the partial programs based on the branch history bit sequence, and an execution time estimation calculating section that adds the execution time of the partial programs based on the execution sequences of the partial programs.
Type: Grant

Filed: June 23, 2009

Date of Patent: May 21, 2013

Assignee: Tokyo Institute of Technology

Inventors: Tsuyoshi Isshiki, Hiroaki Kunieda, Naoto Kobayashi

Creating a composite program module in a computing ecosystem

Patent number: 8448136

Abstract: Methods are provided for creating a composite program module within a computing ecosystem. The ecosystem comprises physical and/or virtual computing resources of a company, department or other organization. When a user discovers a grouping of modules that is useful to him, the ecosystem or his workspace allows him to create a composite module that comprises the group of modules. He can then access their combined functionality by activating just the composite module, in which case all the modules are automatically opened, without having to activate each one individually. Further, the ecosystem may suggest complementary program modules to a user based on his pattern of activity with one or more other modules. A composite program module may be created to include a suggested complementary module, or just modules discovered independently by the user.
Type: Grant

Filed: June 25, 2009

Date of Patent: May 21, 2013

Assignee: Intuit Inc.

Inventors: Michael I. Mills, Justin M. Glaeser, Sharla K. Fullerton, Aline Baeck, Kumiko T. Toft, Noah D. Shaffer

Apparatus and method for network driver injection into target image

Patent number: 8407662

Abstract: A method provides network driver injection into a target image to transform the target image to be compatible with one or more source machines, for facilitating operating system streaming over a network. The method may include: facilitating access to a source system registry file of a source machine; facilitating access to a target system registry file of the target image, without copying the target image; determining whether source network interface cards of the source machine are compatible with the target image; and if the source network interface cards are not compatible with the target image, performing network interface driver injection into the target image. The target image may include an operating system. A machine-readable storage medium and apparatus are provided. A method is described for building a program for providing network driver injection into a target

image to transform the target image to be compatible with computing machines.

Type: Grant
Filed: June 25, 2010
Date of Patent: March 26, 2013
Assignee: Wyse Technology Inc.
Inventor: Mandy Sui Mei Lam

Ask a Lawyer

Question:

Please Ask Your Question Here. e.g., Do I need a Bankruptcy Lawyer?

Add details   120

Ask Question

Find a Lawyer

Patents

City, State

Search

Lawyers - Get Listed Now!

Get a free directory profile listing

Justia Legal Resources

FIND A LAWYER   ▼

INDIVIDUALS   ▼

BUSINESS   ▼

LAW STUDENTS   ▼

US FEDERAL LAW   ▼

US STATE LAW   ▼

OTHER DATABASES   ▼

LEGAL MARKETING   ▼