

# Venkatesh-Prasad Ranganath ([website](#))

---

## Education

- Ph.D. (2006) + MS (2002), Computer Science, Kansas State University, USA
- B.E (1997), Computer Science & Engineering, Bangalore University, India

## Professional Experience

- **Assistant Professor, Kansas State University, USA, Aug'15 – May'19**
  - Led efforts to
    - Create a repository of verifiable Android app vulnerability benchmarks [Ghera]
    - Evaluate efficacy of prevalent security analysis tools [Rekha] and vulnerability benchmark suites
  - Conducted user study to understand the pain points of users of HPC clusters
  - Contributed to the development of architecture and process to build integrated medical systems
  - Supervised efforts to
    - Create mobile and desktop apps for veterinarians to collect and report data about cattle
    - Create mobile apps to advise farmers about cattle medication
    - Create and evaluate a model-based design methodology to incrementally specify, reason, and bake in security properties into mobile apps
  - Led a [research group focused on software engineering and security](#)
  - Discovered and reported two security bugs in Android
  - Developed an undergraduate level course on software testing and a graduate-level course on verification/validation
- **Visiting Assistant Professor, Kansas State University, USA, Feb'14 – Aug'15**
  - Developed communication patterns that enable QoS-aware design and reasoning of integrated medical systems [SCP]
  - Explored the use of machine learning to vet Android apps
  - Developed module about code-level specifications for a graduate-level course on software specification
- **Researcher, Microsoft Research, India, Aug'07 – Sep'13**
  - Developed techniques to synthesize and enforce concurrency control in programs
  - Developed techniques to mine of quantified structural and temporal patterns from event streams [Tark]
  - Led efforts in collaboration with product teams to use quantified structural and temporal patterns to
    - Test backward compatibility of USB3 driver stack with USB2 driver stack in Windows 8
    - Minimize test suites to test Windows USB driver stacks in Device Compatibility Lab
    - Optimize testing of changes to algorithms that enabled Microsoft Ad platform
  - Supervised and mentored junior developers and interns
- **Software Engineer, Agitar Software, USA, Sep'06 – May'07**
  - Maintained the reasoning engine used to automate unit test generation in Agitator tool
- **Research/Teaching Assistant, Kansas State University, Aug'99 – Aug'06**
  - Developed scalable approaches to program analyses and transformations to help optimize verification of concurrent programs via model checking [Indus]
  - Developed techniques and tooling to enable model-driven development of avionics systems
  - Developed caching strategies to optimize SOAP
  - Developed a set-based approach to iptables
- **Software Engineer, Wipro Technologies (prev. Wipro Global R&D), India, Sep'97 – Jul'99**
  - Developed video rendering component of DirectX based MPEG-1 and -2 video decoder

- Developed MPEG-1 and -2 audio stream parsing component of a MPEG audio decoder
- Developed an SEI compliant engineering and documentation process for medium-sized projects

## Selection of Projects/Software

- **[Rekha](#)**: Evaluation of Android Security Analysis Tools
  - Co-Creator / Groovy, Java, R, and Android
- **[Ghera](#)**: Repository of verifiable benchmarks of known Android app vulnerabilities
  - Co-Creator / Java and Android
- **[SCP](#)**: Communication patterns to enable QoS-aware composition of loosely-coupled systems
  - Creator / Java and Vert.x
- **[Tark](#)**: Highly concurrent toolkit to mine structural and temporal patterns from event streams
  - Co-Creator / F# and .NET
- **[Indus](#)**: Library to analyze and slice concurrent Java programs (+110K downloads)
  - Creator / Java and Eclipse
- **[Cadena](#)**: Toolkit to design, analyze, and synthesize component-based systems (+21K downloads)
  - Contributor / Java and Eclipse
- **[Bandera](#)**: Toolkit to verify concurrent Java programs (+24K downloads)
  - Contributor / Java and Eclipse

## Publications & Service

- Authored 35 peer-reviewed articles, five patents, two book chapters, and one booklet
- Served as reviewer for funding agencies and numerous publication venues
- Served as social media chair at FSE'16 and local co-chair at APLAS'08

## Code Repositories

[Bitbucket \(Personal\)](#) | [Bitbucket \(SecureIT Team\)](#) | [GitHub \(Personal\)](#) | [GitHub Gist](#)

# Venkatesh Prasad Ranganath

<http://rvprasad.com>

## Education

- Ph.D. (Computer Science), Kansas State University, USA. (2006)  
– *Scalable and Accurate Approaches to Program Dependence Analysis, Slicing, and Verification of Concurrent Object Oriented Programs*
- M.S. (Computer Science), Kansas State University, USA. (2002)  
– *Object-flow analysis for optimizing finite-state models of Java Software*
- B.E. (Computer Science and Engineering), Bangalore University, India. (1997)

## Employment

- Assistant Professor, Kansas State University, USA. (Aug 2015 – May 2019)
- Visiting Assistant Professor, Kansas State University, USA. (Feb 2014 – Aug 2015)
- Researcher, Microsoft Research, India. (Aug 2007 – Sep 2013)
- Software Engineer, Agitar Software, USA. (Sep 2006 – May 2007)
- Instructor, Kansas State University, USA. (Sep 2005 – Apr 2006)
- Graduate Research Assistant, Kansas State University, USA. (May 2002 – Aug 2005)
- Graduate Teaching Assistant, Kansas State University, USA. (Sep 1999 – Apr 2002)
- Software Engineer, Wipro Global R&D (Wipro Technologies), India. (Sep 1997 – Jul 1999)

## Peer-reviewed Journal Publications

1. *Are Free Android App Security Analysis Tools Effective in Detecting Known Vulnerabilities?* – Venkatesh-Prasad Ranganath and Joydeep Mitra. *Empirical Software Engineering*, 2019.
2. *Mining Quantified Temporal Rules: Formalism, Algorithms, and Evaluation* – David Lo, G. Ramalingam, Venkatesh Prasad Ranganath, and Kapil Vaswani. *Science of Computer Programming (SCP)*, Volume 77, Issue 6, 2012.
3. *Logical Concurrency Control from Sequential Proofs* – Jyotirmoy Deshmukh, G. Ramalingam, Venkatesh Prasad Ranganath, and Kapil Vaswani. *Logical Methods in Computer Science (LMCS)*, Volume 7, Issue 3, 2011.
4. *A New Foundation For Control-Dependence and Slicing for Modern Program Structures* – Venkatesh Prasad Ranganath, Torben Amtoft, Anindya Banerjee, Matthew B. Dwyer, and John Hatcliff. *ACM Transactions on Programming languages and Systems (TOPLAS) - Special Issue ESOP 2005*, Volume 20, Issue 5, 2007.
5. *Slicing Concurrent Java Programs using Indus and Kaveri* – Venkatesh Prasad Ranganath and John Hatcliff. Special section of the *International Journal on Software Tools for Technology Transfer (STTT)*, Volume 9, Issue 5-6, 2007.

6. *Exploiting Object Escape and Locking Information in Partial-Order Reductions for Concurrent Object-Oriented Programs* – Matthew B. Dwyer, John Hatcliff, Robby, and Venkatesh Prasad Ranganath. *International Journal on Formal Methods in System Design (FMSD)*, Volume 25, 2004.

### Peer-reviewed Conference/Workshop Publications

1. *SeMA: A Design Methodology for Building Secure Android Apps* – Joydeep Mitra and Venkatesh-Prasad Ranganath. *International Workshop on Advances in Mobile App Analysis, A-Mobile 2019*.
2. *BenchPress: Analyzing Android App Vulnerability Benchmark Suites* – Joydeep Mitra and Venkatesh-Prasad Ranganath. *International Workshop on Advances in Mobile App Analysis, A-Mobile 2019*.
3. *Why do Users Kill HPC Jobs?* – Venkatesh-Prasad Ranganath and Daniel Andresen. *International Conference on High Performance Computing, Data, and Analytics, HiPC 2018*. Acceptance: 22%
4. *Ghera: A Repository of Android App Vulnerability Benchmarks* – Joydeep Mitra and Venkatesh-Prasad Ranganath. *International Conference on Predictive Modeling and Data Analytics in Software Engineering, PROMISE 2017*.
5. *Experimental Study with Real-world Data for Android App Security Analysis using Machine Learning* by Sankardas Roy, Jordan DeLoach, Yuping Li, Doina Caragea, Xinming Ou, Nicolae Herndon, Venkatesh Prasad Ranganath, HongMin Li, and Nicolais Guevara. *Annual Computer Security Applications Conference, ACSAC 2015*. Acceptance: 24.4%
6. *Ecosphere Principles for Medical Application Platforms* by Yu Jin Kim, Venkatesh-Prasad Ranganath, John Hatcliff, Robby and Sam Procter. *International Conference on Healthcare Informatics, ICHI 2015*.
7. *Communication Patterns for Interconnecting and Composing Medical Systems* – Venkatesh-Prasad Ranganath, Yu Jin Kim, John Hatcliff, and Robby. *International Conference of the IEEE Engineering in Medicine and Biology Society, EMBC 2015*. (Reviewed Invited Paper)
8. *Integrated Clinical Environment Device Model: Stakeholders and High Level Requirements* – Yu Jin Kim, John Hatcliff, Venkatesh-Prasad Ranganath, Robby, and Sandy Weininger. *Medical Cyber Physical Systems Workshop, Medical CPS 2015*.
9. *Compatibility Testing using Patterns-based Trace Comparison* – Venkatesh-Prasad Ranganath, Pradip Vallathol, and Pankaj Gupta. *International Conference on Automated Software Engineering, ASE 2014*. Acceptance: 19.9%
10. *Extrinsic Influence Factors in Software Reliability: A Study of 200,000 Windows Machines* – Christian Bird, Venkatesh Prasad Ranganath, Thomas Zimmermann, Nachiappan Nagappan, and Andreas Zeller. *International Conference on Software Engineering (SEIP track), ICSE 2014*. Acceptance: 21%
11. *Structural and Temporal Patterns-based Features* – Venkatesh Prasad Ranganath and Jithin Thomas. *International Workshop on Data Analysis Patterns in Software Engineering, DAPSE 2013*.

12. *Logical Concurrency Control from Sequential Proofs* – Jyotirmoy Deshmukh, G. Ramalingam, Venkatesh Prasad Ranganath, and Kapil Vaswani. *European Symposium on Programming, ESOP 2010*. Acceptance: 25%  
WINNER OF ETAPS BEST PAPER AWARD.
13. *Mining Quantified Temporal Rules: Formalism, Algorithms, and Evaluation* – David Lo, G. Ramalingam, Venkatesh Prasad Ranganath, and Kapil Vaswani. *Working Conference on Reverse Engineering, WCRE 2009*. Acceptance: 25.3%
14. *Isolator: Dynamically Ensuring Isolation in Concurrent Programs* – G. Ramalingam, Sriram K. Rajamani, Venkatesh Prasad Ranganath, and Kapil Vaswani. *International Conference on Architectural Support for Programming Languages and Operating Systems, ASPLOS 2009*. Acceptance: 26%
15. *An Overview of the Indus Framework for Analysis and Slicing of Concurrent Java Software* – Venkatesh Prasad Ranganath and John Hatcliff. *International Workshop on Source Code Analysis and Manipulation, SCAM 2006*. (Keynote)
16. *Evaluating the Effectiveness of Slicing for Model Reduction of Concurrent Object-Oriented Programs* – Matthew B. Dwyer, John Hatcliff, Matthew Hoosier, Venkatesh Prasad Ranganath, Robby, and Todd Wallentine. *International Conference on Tools and Algorithms for the Construction and Analysis of Systems, TACAS 2006*. Acceptance: 27%
17. *Automatic code generation for LYE, a high-performance caching SOAP implementation* – Venkatesh Prasad Ranganath, Andrew King, and Daniel Andresen. *International Conference on Semantic Web and Web Services, SWWS 2006*. Acceptance: 32%
18. *Towards highly optimized real-time middleware for software product-line architectures* – Arvind S Krishna, Aniruddha Gokhale, Douglas C. Schmidt, Venkatesh Prasad Ranganath, and John Hatcliff. *ACM SIGBED Review - Special issue: The work-in-progress (WIP) session of the RTSS 2005*.
19. *A New Foundation For Control-Dependence and Slicing for Modern Program Structures* – Venkatesh Prasad Ranganath, Torben Amtoft, Anindya Banerjee, Matthew B. Dwyer, and John Hatcliff. *European Symposium on Programming, ESOP 2005*. Acceptance: 24.5%
20. *Kaveri: Delivering Indus Java Program Slicer to Eclipse* – Ganeshan Jayaraman, Venkatesh Prasad Ranganath, and John Hatcliff. *International Conference on Fundamental Approaches to Software Engineering, FASE 2005*.
21. *LYE: high performance SOAP with multi-level caching* – Venkatesh Prasad Ranganath, David Saxton, and Daniel Andresen. *International Conference on Parallel and Distributed Computing and Systems, PDCS 2004*. Acceptance: 66%  
NOMINATED FOR BEST PAPER AWARD.
22. *LYE: a high performance caching SOAP implementation* – Daniel Andresen, David Saxton, Kiran Devaram, and Venkatesh Prasad Ranganath. *International Conference on Parallel Processing, ICPP 2004*. Acceptance: 34.2%
23. *Pruning Interference and Ready Dependence for Slicing Concurrent Java Programs* – Venkatesh Prasad Ranganath and John Hatcliff. *International Conference on Compiler Construction, CC 2004*. Acceptance: 32.7%

24. *A Correlation Framework for CORBA Component Model* – Georg Jung, John Hatcliff, and Venkatesh Prasad Ranganath. *International Conference on Fundamental Approaches to Software Engineering, FASE 2004*. Acceptance: 24.1%
25. *Cadena: An Integrated Development Environment for Analysis, Synthesis, and Verification of Component-based Systems* – Adams Child, Jesse Greenwald, Venkatesh Ranganath, Xi-anhua Deng, Matthew Dwyer, John Hatcliff, Georg Jung, Prashant Shanti, and Gurdip Singh. *International Conference on Fundamental Approaches to Software Engineering, FASE 2004*.
26. *A Set-based Approach to Packet Classification* – Venkatesh Prasad Ranganath and Daniel Andresen. *International Conference on Parallel and Distributed Computing and Systems, PDCS 2003*. Acceptance: 66%  
NOMINATED FOR BEST PAPER AWARD.
27. *CADENA: Enabling CCM-based Application Development in Eclipse* – Venkatesh Prasad Ranganath, Adam Childs, Jesse Greenwald, Matthew B. Dwyer, John Hatcliff and Gurdip Singh. *Workshop on eclipse technology exchange, eTX 2003*.
28. *Cadena: An Integrated Development, Analysis, and Verification Environment for Component-based Systems* – John Hatcliff, William Deng, Matthew B. Dwyer, Georg Jung, and Venkatesh Prasad Ranganath. *International Conference on Software Engineering, ICSE 2003*. Acceptance: 12.9%
29. *Slicing and Partial Evaluation of CORBA Component Model Designs for Avionics System* – John Hatcliff, William Deng, Matthew B. Dwyer, Georg Jung, Venkatesh Prasad Ranganath, and Robby. *Workshop on Partial Evaluation and Program Manipulation, PEPM 2003*.

## Book Content

1. *Logging in Python*, 2020. Published via Leanpub.
2. *Embrace Dynamic Artifacts in Perspectives on Data Science for Software Engineering*, 2016. Published by Morgan Kaufmann. ISBN: 978-0128042069.
3. *While Models are Good, Simple Explanations are Better in Perspectives on Data Science for Software Engineering*, 2016. Published by Morgan Kaufmann. ISBN: 978-0128042069.

## Miscellaneous Publications

1. *SeMA: Extending and Analyzing Storyboards to Develop Secure Android Apps* – Joydeep Mitra, Venkatesh-Prasad Ranganath; 2020.
2. *Localized Disaster Networks Platform* – Venkatesh-Prasad Ranganath, Hyung Jin Kim, and Daniel Andresen; 2017.
3. *Controlling Non-determinism for Semantic Guarantees* – Sriram Rajamani, G. Ramalingam, Venkatesh Prasad Ranganath, and Kapil Vaswani. *Exploiting Concurrency Efficiently and Correctly, (EC)<sup>2</sup>, a CAV 2008 workshop*.
4. *Component-Oriented Programming and Datacenter Applications* – Venkatesh Prasad Ranganath. *The Rise and Rise of the Declarative Datacentre (R2D2)*; 2008.

5. *Enabling Efficient Partial Order Reductions for Model Checking Object-Oriented Programs Using Static Calculation of Program Dependences* – Venkatesh Prasad Ranganath, John Hatcliff, and Robby; 2007.
6. *Notes on Interference Dependences* – Venkatesh Prasad Ranganath; 2007.
7. *Notes on Atomicity* – Venkatesh Prasad Ranganath; 2006.
8. *Enriching Component Interfaces with Checkable Dependence Specifications* – Venkatesh Prasad Ranganath, Georg Jung, John Hatcliff, and Matthew B. Dwyer; 2005.

## Patents

1. *Compatibility Testing Using Traces, Linear Temporal Rules, and Behavioral Models* – Randall Edward Aull, Pankaj Bharti Gupta, Robert Eugene Harris Jr, Jane Evguenia Lawrence, Venkatesh-Prasad Ranganath, and Pradip Harindran Vallathol; 2012. Patent US 8892493 B2. (Pending)
2. *Temporal Rule-Based Feature Definition and Extraction* – Venkatesh-Prasad Ranganath, Piyush Goyal, Pradip Harindran Vallathol, and Ganesan Ramalingam; 2010. Patent US 8538909 B2.
3. *Abstracting Events for Data Mining* – David Lo, Ganesan Ramalingam, Venkatesh-Prasad Ranganath, and Kapil Vaswani; 2009. Patent US 8280899 B2.
4. *Identifying Concurrency Control from a Sequential Proof* – Ganesan Ramalingam, Sriram Rajamani, Venkatesh-Prasad Ranganath, Kapil Vaswani, and Jyotirmoy Vinay Deshmukh; 2008. Patent US 20100169618 A1. (Pending)
5. *System to Reduce Interference in Concurrent Programs* – Sriram Rajamani, Ganesan Ramalingam, Venkatesh-Prasad Ranganath, and Kapil Vaswani; 2008. Patent US 7941616 B2.

## Courses

- Created and taught *CIS890 - SAT Solving* in Spring 2018.
- Created and taught *CIS640 - Software Testing Techniques* in Spring 2016 and 2017.
- Created and taught *CIS841 - Verification and Validation* in Fall 2015, 2016, and 2017.
- Taught *CIS771 - Software Specification* in Spring 2015.

## Software/Projects

**Rekha** Evaluation of Android security analysis tools and benchmarks.

*Technologies:* Java and Android

<https://bitbucket.org/secure-it-i/evaluate-representativeness>

<https://bitbucket.org/secure-it-i/may2018>

**Ghera** Repository of Android app vulnerability benchmarks.

*Technologies:* Java and Android

<https://bitbucket.org/secure-it-i/android-app-vulnerability-benchmarks/>

**SCP** Set of communication patterns to compose medical devices into medical systems.

*Technologies:* Java and Vert.x

<https://bitbucket.org/rvprasad/clinical-scenarios/src/master/simple-communication-patterns>

**Tark** Toolkit to mine structural and linear temporal patterns.

*Technologies:* F# and .NET

<http://research.microsoft.com/en-us/projects/tark> (1530 downloads)

**Indus** Library to analyze and slice concurrent Java programs.

*Technologies:* Java and Eclipse

<http://indus.projects.cis.ksu.edu> (110,000+ downloads until 2016)

**Cadena** Toolkit to design, analyze, and synthesize component-based systems.

*Technologies:* Java and Eclipse

<http://cadena.projects.cis.ksu.edu> (21,000+ downloads until 2016)

**Bandera** Toolkit to verify concurrent Java programs.

*Technologies:* Java and Eclipse

<http://bandera.projects.cis.ksu.edu> (24,000+ downloads until 2016)

**LYE** High-performance SOAP via caching.

<http://lye.projects.cis.ksu.edu>

*Technologies:* Java and Apache Axis

## Awards

Monetary rewards from Android Security Rewards program for reporting high priority security bugs (CVE-2018-9548, CVE-2019-9463).

## Service

### Organization

Social Media Chair of International Symposium of Foundations of Software Engineering (FSE) 2016.

Local co-chair of Asian Symposium on Programming Languages and Systems (APLAS) 2008.

### Program committee member at following venues

*ACSEAC'12* African Conference on Software Engineering & Applied Computing

*CSI'06* National Annual Convention, organized by the Computer Society of India

*DAPSE'13* International Workshop on Data Analysis Patterns in Software Engineering

*ICIIT'18* International Conference on Intelligent Information Technologies

*ISEC'13,'12,'11,'10,'09* India Software Engineering Conference

*ISSRE'12* International Symposium on Software Reliability Engineering

*PADTAD'12* Workshop on Parallel and Distributed Systems: Testing, Analysis, and Debugging



*SEDE '12* International Conference on Software Engineering and Data Engineering  
*RV '12* International Conference on Runtime Verification

**Reviewer for following journals**

*EMSE* Empirical Software Engineering  
*JCST* Journal of Computer Science and Technology  
*JOT* Journal of Object Technology  
*SCP* Science of Computer Programming  
*STTT* International Journal on Software Tools for Technology Transfer  
*TECS* ACM Transactions on Embedded Computing Systems  
*TOPLAS* ACM Transactions on Programming Languages and Systems  
*TOSEM* ACM Transactions on Software Engineering and Methodology  
*TSE* IEEE Transactions on Software Engineering

**External reviewer at following venues**

*APLAS'08* Asian Symposium on Program Languages and Systems  
*ATVA'13* International Symposium on Automated Technology for Verification and Analysis  
*CAV'09* Computer Aided Verification  
*DEBS'05* International Workshop on Distributed Event-Based Systems  
*EAIT'06* International Conference on Emerging Applications of IT  
*eTX'04* Eclipse Technology Exchange Workshop  
*FASE'04* Fundamental Approaches to Software Engineering  
*FSE'09* Foundations of Software Engineering  
*FTFJP'05* ECOOP Workshop on Formal Techniques for Java-like Programs  
*ICISS'09* International Conference on Information Systems Security  
*ICSE'10, '09, '08* International Conference on Software Engineering  
*ISSTA'08* International Symposium on Software Testing and Analysis  
*OOPSLA'09* International Conference on Object Oriented Programming, Systems, Languages and Applications  
*NFM'16* NASA Formal Methods Symposium  
*PASTE'05* Workshop on Program Analysis for Software Tools and Engineering  
*PLAS'06* Workshop on Programming Languages and Analysis for Security  
*PLDI'06, '04* International Conference on Programming Language Design and Implementation  
*POPL'09, '07, '06* Symposium on Principles of Programming Languages  
*PPOPP'11* Annual Symposium on Principles and Practice of Parallel Programming  
*SAS'06* International Static Analysis Symposium  
*SAVCBS'05* Specification and Verification of Component-Based Systems Workshop Engineering

*TACAS'08* International Conference on Tools and Algorithms for Construction and Analysis  
of Systems

**Member of Agency Review Panels**

*NSF* National Science Foundation 2016, 2018

*DOE* Department of Energy 2016