Formal Methods at Scale

Workshops & Opportunity Motivations

Brad Martin

Adapted in part from the workshop report out by:

Patrick Lincoln, Bill Scherlis, Katie Dey, and myself

FM@Scale Workshops – Purpose and Focus

Two meetings (one east coast, one west coast) on Formal Methods at Scale

- Under the auspices of National Science and Technology Council (NSTC) Special Cyber Operations Research and Engineering Subcommittee (SCORE)

Purpose:

- Identify successes, barriers, opportunities, and challenges regarding the use of formal methods in cyber systems
- Understand how the systems engineering and formal methods (FM) communities can achieve broader use of the technologies at increasing levels of scale

Technical focus:

- Formal methods logics, tools, and socio-technical ecosystems
- Experience with applications in practice
- Potential means to evaluate costs, risks, benefits – and formulate adoption cases

FM@Scale Workshops – Goals and Context

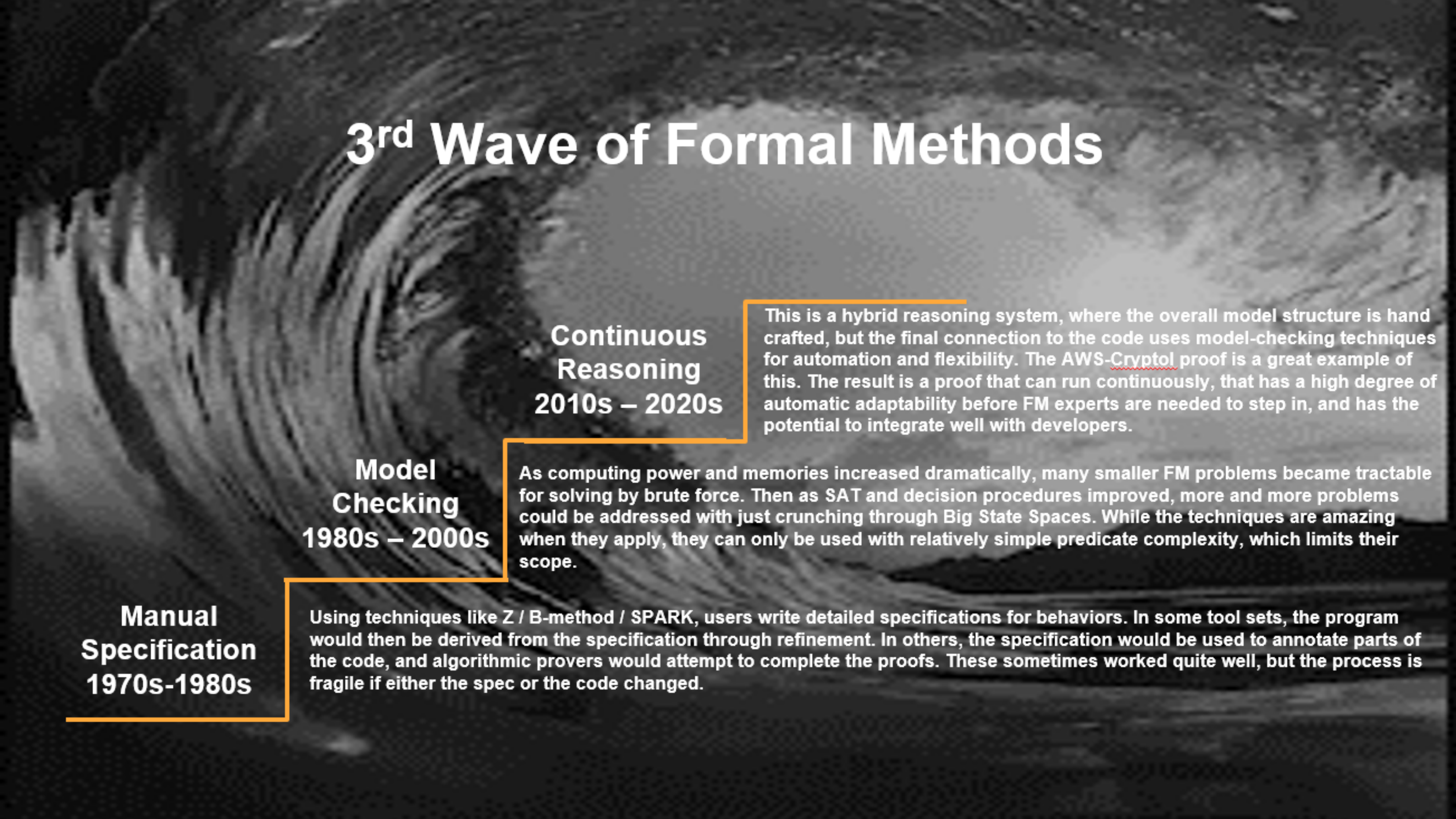
Goal:

- Improve understanding of how the formal methods (FM) community, in partnership with sponsors and users, might achieve broader use and at increasing levels of scale.

Context:

- In the half-century history of formal methods research and use, we have experienced both steps forward and also crises of expectations.
- This is analogous to the history of AI, which ultimately crossed a threshold of scale and adoptability around 2000.
- Some users and researchers believe we are at a similar inflection point with FM.

3rd Wave of Formal Methods



Continuous Reasoning 2010s – 2020s

This is a hybrid reasoning system, where the overall model structure is hand crafted, but the final connection to the code uses model-checking techniques for automation and flexibility. The AWS-Cryptol proof is a great example of this. The result is a proof that can run continuously, that has a high degree of automatic adaptability before FM experts are needed to step in, and has the potential to integrate well with developers.

Model Checking 1980s – 2000s

As computing power and memories increased dramatically, many smaller FM problems became tractable for solving by brute force. Then as SAT and decision procedures improved, more and more problems could be addressed with just crunching through Big State Spaces. While the techniques are amazing when they apply, they can only be used with relatively simple predicate complexity, which limits their scope.

Manual Specification 1970s-1980s

Using techniques like Z / B-method / SPARK, users write detailed specifications for behaviors. In some tool sets, the program would then be derived from the specification through refinement. In others, the specification would be used to annotate parts of the code, and algorithmic provers would attempt to complete the proofs. These sometimes worked quite well, but the process is fragile if either the spec or the code changed.

FM – Dimensions of *Scale*

1. **Scope**. The range of properties and qualities that are modeled and reasoned about, such as relating to security, safety, performance, fault tolerance, real-time, etc.
2. **Complexity**. Complexity and the size of systems and their supply chains, including issues related to composability
3. **Practice**. Efficiency of FM-related modeling, tooling, and engineering practices, including integration into mainstream tooling and practices
4. **Evolution**. Ability to rapidly co-evolve systems and associated evidence
5. **Adoptability**. Ease of use for non-expert developers and evaluators.

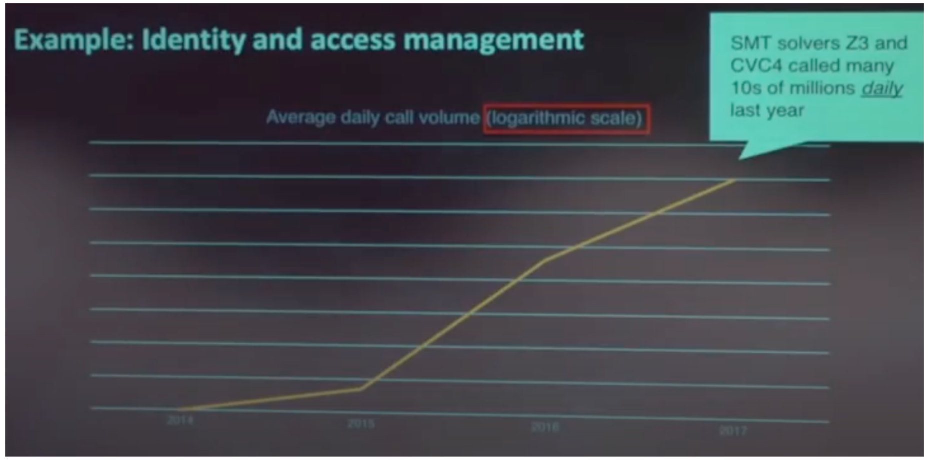
FM – Dimensions of *Experience*

1. **Major systems**. Applications to specific major systems in government and industry
2. **Big results**. Tour-de-force results, such as proofs of significant mathematical results or reasoning about modern processors
3. **Ecosystems**. The legacy, sustainment, and advancement of formal methods ecosystems surrounding the various provers and stacks
4. **Broad use**. Integration of more limited capabilities into broader communities of practice, such as has been happening in major tech firms.

Market Adoption *for Continuous Reasoning*

Recent formal methods successes at AWS, Facebook, Google, others:

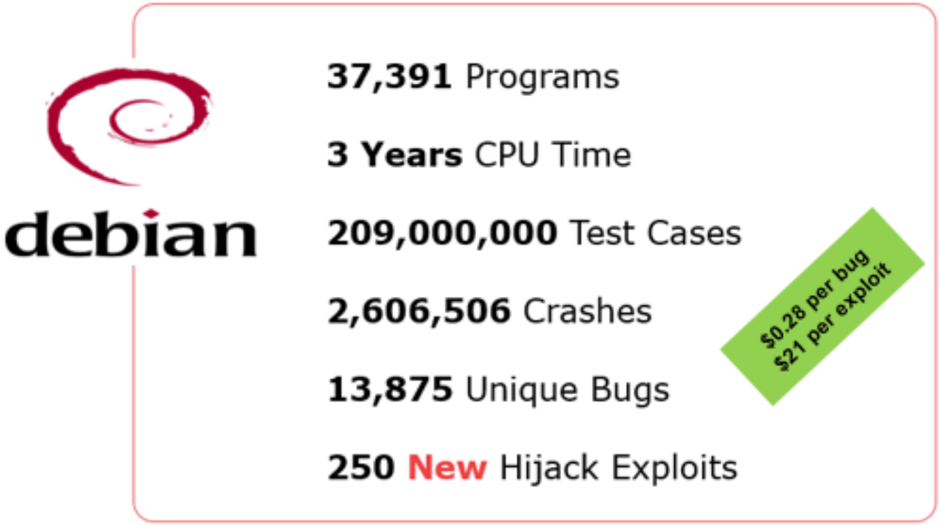
- **s2n TLS (AWS)** - cryptographic properties of code checked within seconds of developer commit
- **Identity Access Manager (AWS)** - cloud access controls validated in seconds for millions of Amazon network policies
- **Infer (Facebook)** - memory safety checked on developer commit for 100k+ loc apps
- **ErrorProne (Google)** - correctness checked in the IDE for millions of lines of code



Example: Identity and access management

SMT solvers Z3 and CVC4 called many 10s of millions *daily* last year

AWS Identity Access Manager tool adoption, 2014-17



37,391 Programs

3 Years CPU Time

209,000,000 Test Cases

2,606,506 Crashes

13,875 Unique Bugs

250 New Hijack Exploits

$0.28 per bug
$21 per exploit

FOR ALL SECURE Checking Debian Linux for Vulnerabilities

Market Adoption *for Continuous Reasoning*

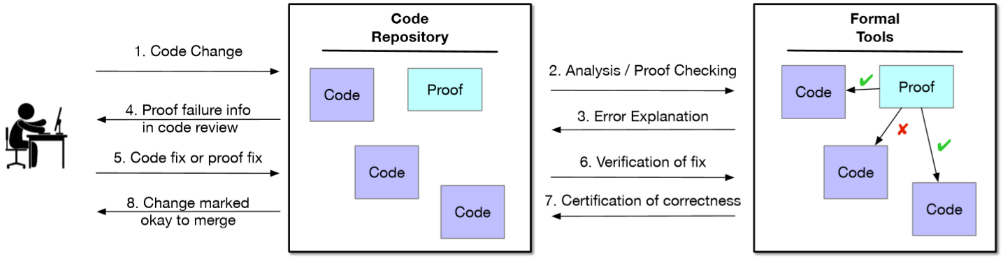
Tools are deployed in production environments:

- Properties delivered to domain-expert teams building and maintaining systems
- Assured systems range up to millions of lines of code / hundreds of millions of users
- Systems requirements rapidly change as use cases, security threat models, and developer intentions change
- Code rapidly changes, e.g. at FB / Google, hundreds of commits per day

> … a striking situation where the diff time deployment saw a 70% fix rate, where a more traditional “offline” or “batch” deployment (where bug lists are presented to engineers, outside their workflow) saw a 0% fix rate.

Key enablers of success:

- Integration into developer workflows
- Flexibility to deliver different assurance requirements
- Scalability in users, developers, lines of code



FM@Scale – Preliminary Findings

Experience. There are successful **major industrial use cases** of formal methods applied to complex, business-critical systems.

- Stakeholders report high return-on-investment for the businesses involved.
- Use cases in government are emerging from research and development projects
- Cross-institution academic use cases are building formally verified entire systems, including systems-of-systems, operating systems, and hardware.

Infrastructure. There are multiple **ecosystems** around several formal tool chains.

- These are maturing and stably evolving, with meaningful sustainment activities
- They are demonstrating increasing robustness and ease of use.
- Barriers are lowering for training new staff to become successful users of the toolchains

Scope. There is evident **opportunity to broaden the scope** of applicability.

- This is supported by explicit focus on increased usability, adoptability, “invisibility,” and integratability of multiple toolchains.

Critical systems*.* There are **increasing opportunities** for critical systems

- Formal methods are being linked with traditional safety cases, security cases, hazard analysis, test plan generation for critical systems.

Community Strategy

Link FM technology “push” with “pull” from potential applications and domains.

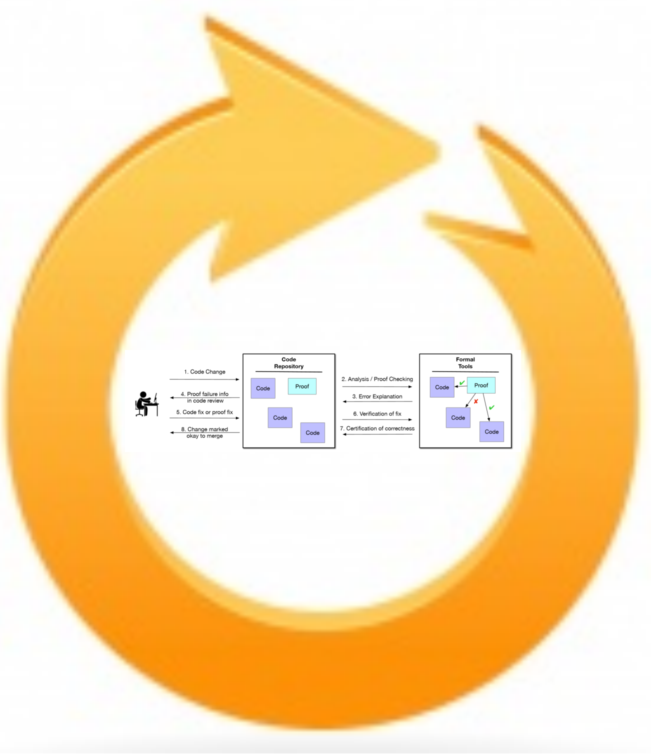
Tipping Point!

Critical Systems

Increasing opportunities

Infrastructure

Multiple ecosystems around several formal tool chains



Scope

Opportunity to broaden scope

Experience

Successful major industrial use cases

Emerging FM Capabilities

Invisible formal methods

- Provide the value of formal analysis, but without requiring users to learn new specification languages, instead delivering capabilities in ways that “hide the math,” such as currently evident in languages with modern type systems.

Small focused proofs

- Tools to facilitate little proofs about big programs can provide relatively easy on-ramps to a wider user base.
- Avoid “all or nothing” approaches, enabling increments of effort in modeling and proving to yield increments of benefit

Integration with model-based development

- Connect formal methods tools with traditional software- and systems-engineering models, enabling early verification and continuous value-add from formal reasoning tools

Formalized threat models

- Enable these to be developed and shared across a community, enabling wide agreement of what is meant by, for example, private information leakage.

Safer machine learning

- Use formal analyses (as in the DARPA Assured Autonomy program) to integrate reasoning about machine learning components into analyses.

Use of cloud infrastructure

- The emerging cloud computing infrastructure can be used to facilitate higher-scale formal methods, and can be the subject of formal analysis.

FM – Opportunities For R&D Investment

(*With the potential for disproportionate impact on quality and capability of systems*)

- **Legacy and hybrids.** Develop methods to provide assurance cases, building on both formal and informal evidence, for modifications and new integrations in existing platforms and systems-of-systems.
- **Security.** Expand existing and develop new methods to apply formal methods to problems in computer security and privacy.
- **Domains.** Address challenges specific to engineering critical domains, including cyber-physical systems, Internet of Things, AI-based systems, autonomous systems, and related.
- **Evidence.** Develop practical methods to ensure evidence including formal artifacts and toolchain information are brought along with components and systems as they are deployed, modified, and maintained.
- **Open source.** Integration with widely use open source components and libraries is an opportunity both to provide immediate assurance benefits and to visibly demonstrate FM engineering integration.
- **Engineering.** Develop methods of property specification, proof, and proof presentation, that ensure that flaws in formal evidence are obvious to domain-focused evaluators.

Conclusion

A revolution in the application of formal methods at scale has occurred over the last few years. There is a broadening range of areas of both commercial and government engineering where there is an existing or emerging mission-focused business case for use of FM.

Tools, practices, and ecosystems are already facilitating commercial, government, and academic application of formal tools across many application domains and types of systems, but work remains to advance the scope, capability, and usability of the key FM technologies, tools, and practices.

The momentum that is emerging regarding use of FM is now increasing, but the technologies are still at an early stage of development with regard to the potential benefits to security, quality, and other kinds of assurance – and also with regarding to the ancillary benefits to developing systems that are both readily adaptable and, on the basis of formal evidence, also readily re-certified.