

Enabling Applied Research and Technology Transfer with Models

Adam Porter

Professor: University of Maryland
Executive Director: Fraunhofer CESE
aporter@cs.umd.edu
aporter@cese.fraunhofer.org



UMD CS Department
The Brendan Iribe Center



2015

My Main Points

- ▶ Use-driven research is interesting, challenging and measurably valuable
- ▶ Funding realities are changing; clearer value demanded
- ▶ Examples of use-driven research (**models help**)



The Fraunhofer-Gesellschaft

- ▶ Europe's largest application-oriented research organization
- ▶ Geared to research that meets society's many needs: health, mobility, energy, communications, security, production, etc.



Joseph von Fraunhofer (1787 – 1826)



- ▶ Researcher - Discovered “Fraunhofer lines” in the solar spectrum
- ▶ Inventor - Developed new methods for processing lenses
- ▶ Entrepreneur - Director of a glassworks

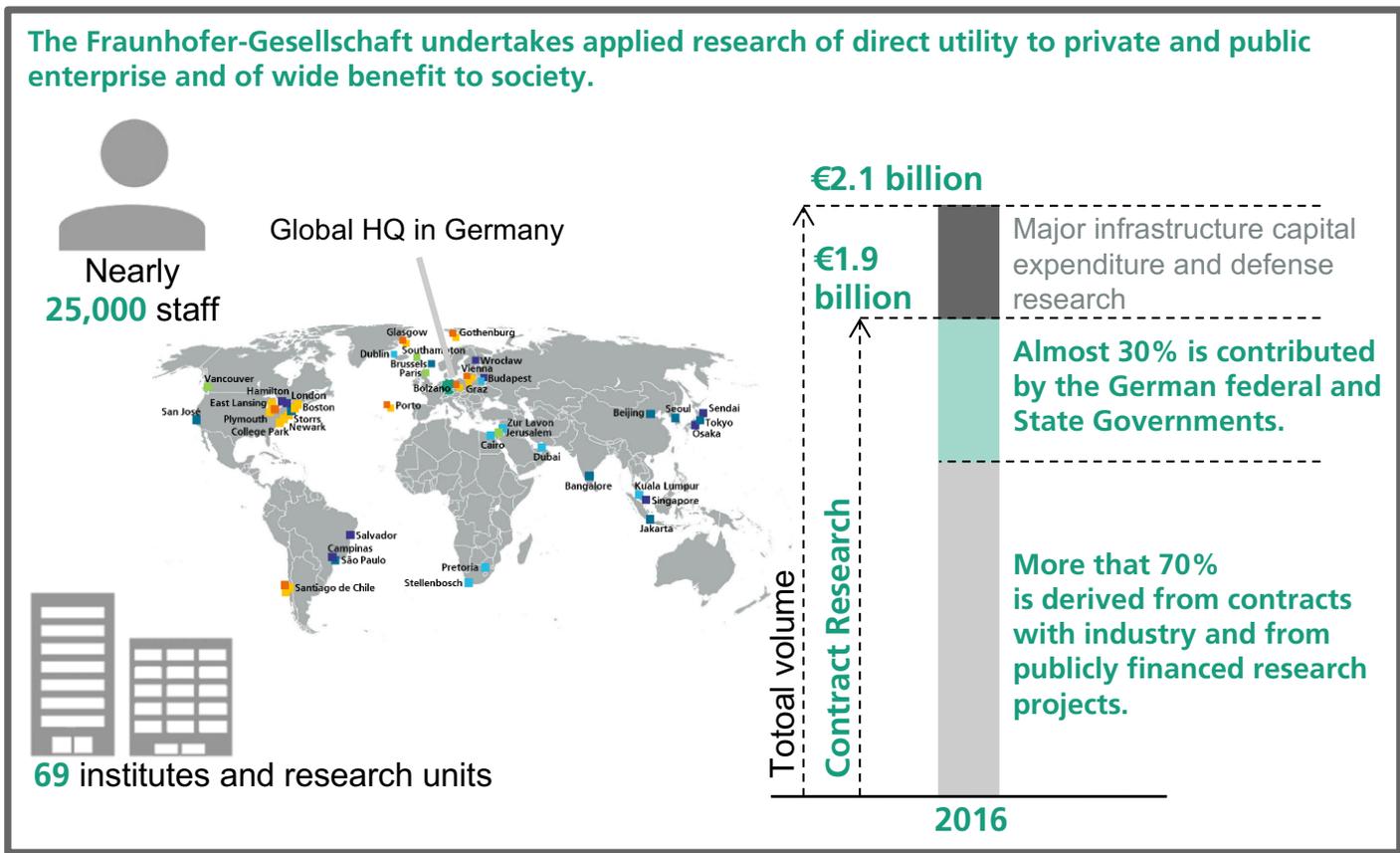


mp3 – A Fraunhofer Invention

- ▶ 1994: Fraunhofer IIS and Micronas jointly develop the first one-chip mp3 decoder.
 - ▷ Fraunhofer IIS also creates the first prototype mp3 player
- ▶ 1995: Fraunhofer and Thomson introduce a joint licensing pool for mp3-related software and patterns
- ▶ Today: mp3 and its successors in use in over 7 billion devices worldwide



The Fraunhofer-Gesellschaft at a Glance



The Fraunhofer Model

- ▶ Fraunhofer's funding model is designed to keep research centers engaged with industry
- ▶ Ideally, annual center revenues should have the following makeup:
 - ▷ 1/3 Institutional (base) funding
 - ▷ 1/3 Competitive public sector funding
 - ▷ 1/3 Industry contracts

Fraunhofer USA locations

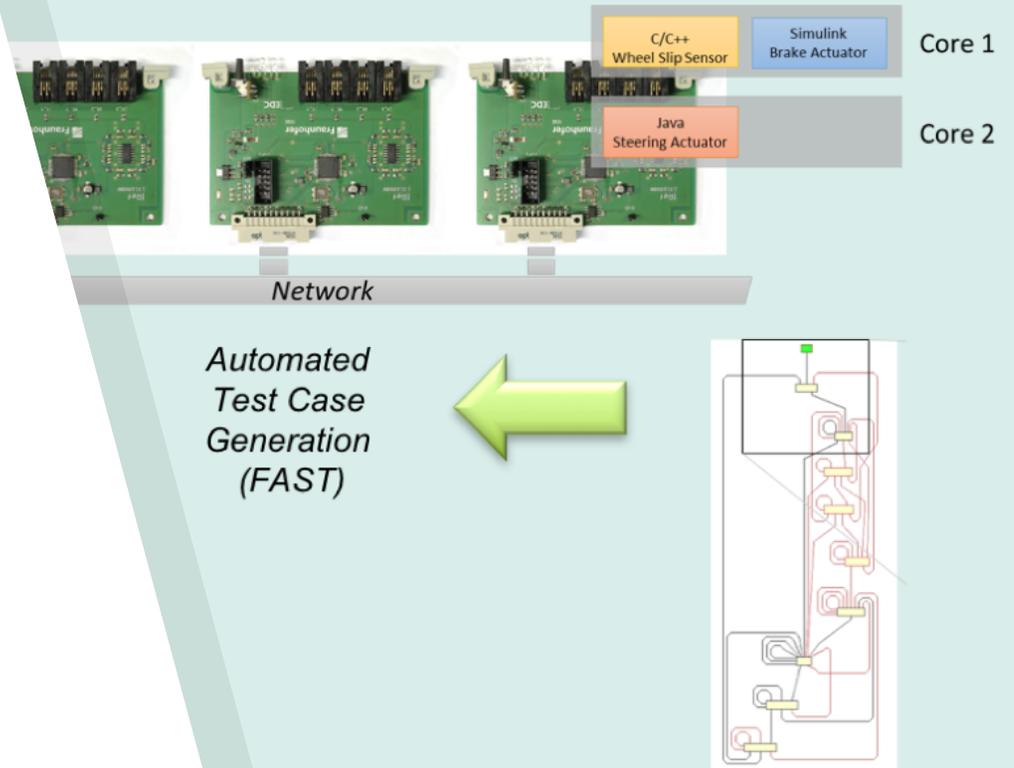


Our Vision

- ▶ Accelerating economic and industrial development for its customers by using innovative model-based methods for developing and assuring complex software-intensive systems.

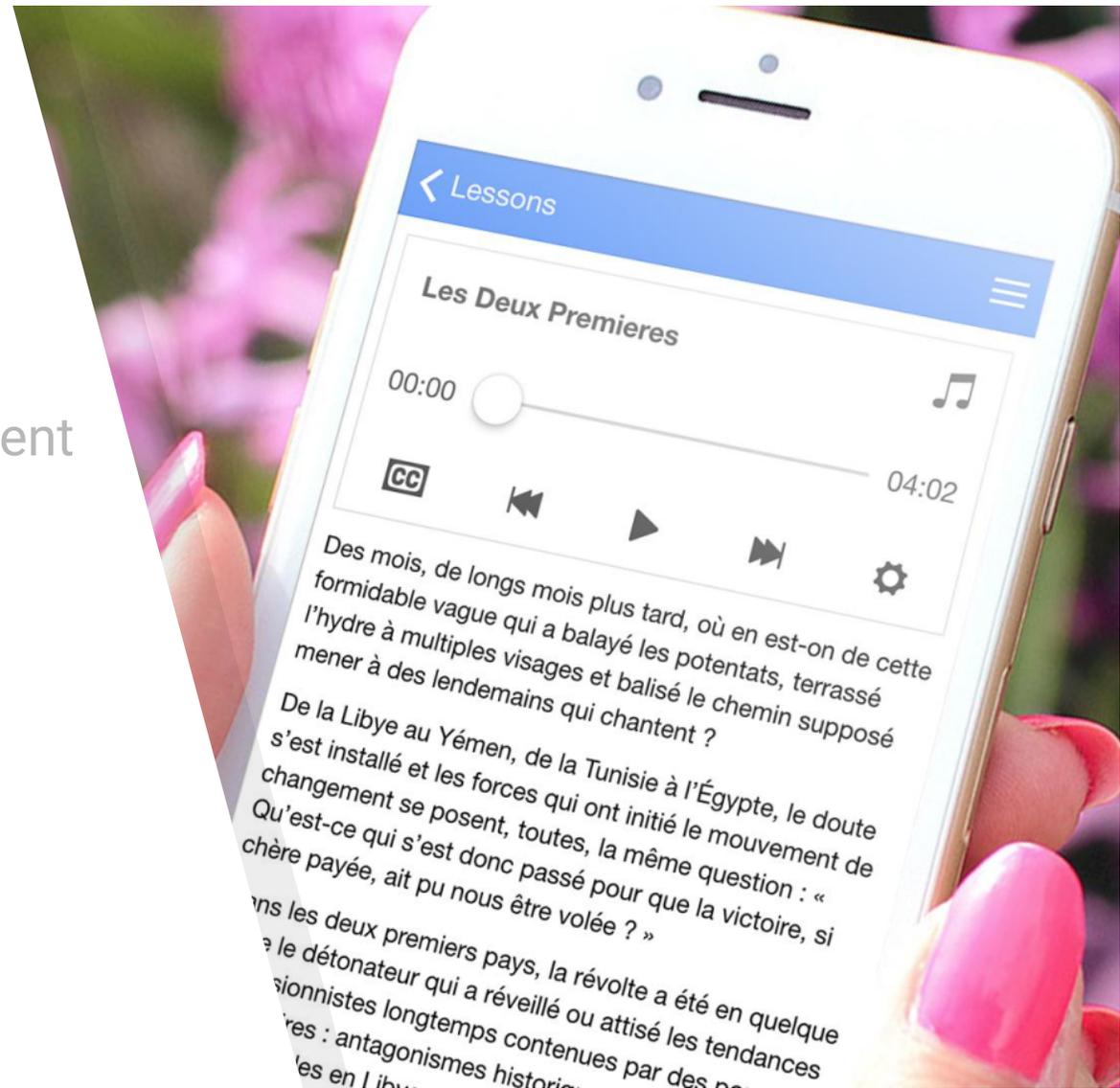
Modeling and Simulation

- ▶ System modeling
- ▶ Simulation
- ▶ Model-based testing



SW Design and Development

- ▶ Requirements definition
- ▶ Architectural design
- ▶ Agile development and continuous testing



Safety and Security

- ▶ Threat modeling
- ▶ Secure coding
- ▶ Security requirements



Project Mgmt. / Analytics

- ▶ Risk Assessment
- ▶ Regulatory Compliance
- ▶ Project Management



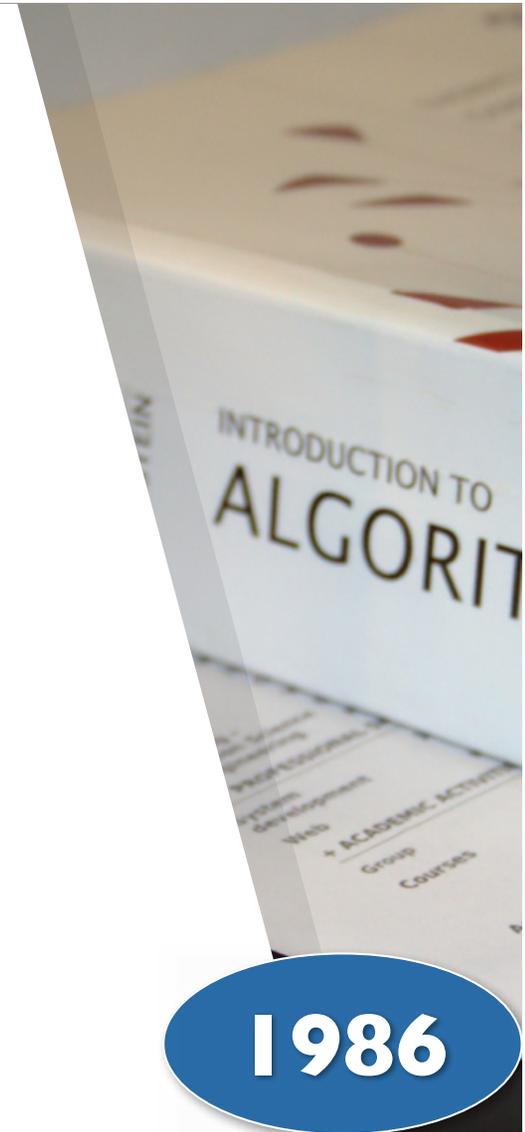
My Main Points

- ▶ **Use-driven research is interesting, challenging and measurably valuable**
- ▶ Funding realities are changing; clearer value demanded
- ▶ Examples of use-driven research (**models help**)



Graduate Study at UCI

- ▶ Great environment / collaborators
- ▶ Ample funding
- ▶ Field was relatively new and expanding rapidly



Curiosity-Driven Research

- ▶ Free to investigate broadly, without must concern for practice



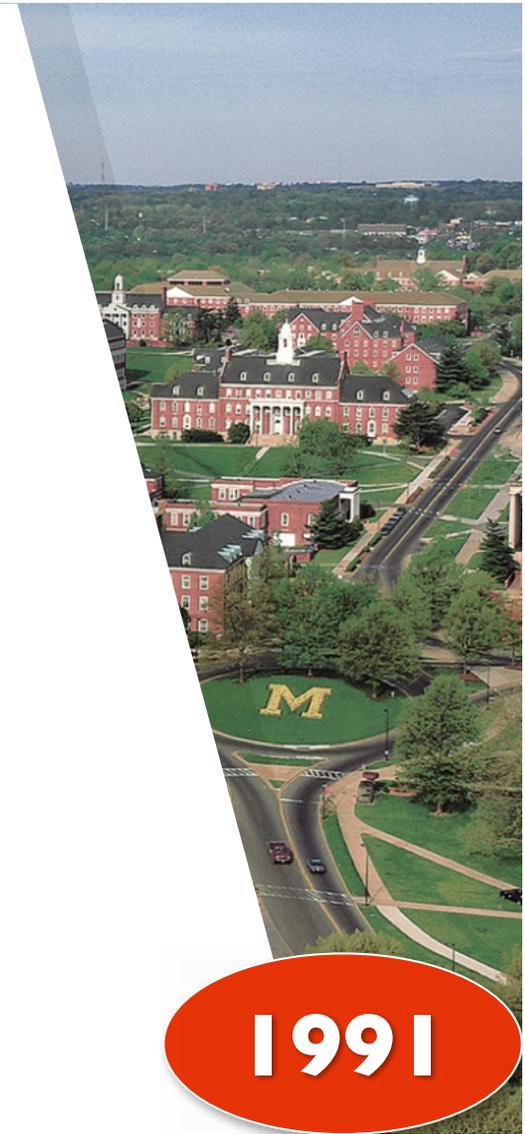
The Good and Bad

- ▶ Graduate study was a great learning experience
- ▶ Wasn't pushed to consider potential impact



Moving to UMD

- ▶ Began to create my research program
- ▶ Began to see value in striving for greater impact
- ▶ Built a partnership with AT&T Bell Laboratories



AT&T Bell Labs

- ▶ Software Production Research Department (SPRD) was intentionally located near one of AT&T's key development sites in Naperville, IL
- ▶ Created to support 5ESS digital switch development
- ▶ 5ESS - Flagship product for AT&T
 - ▷ ~20 MLOC
 - ▷ Massively configurable, supporting over 50 patch lines
 - ▷ Extreme reliability requirements
 - ▷ Engineering staff exceeded 5k FTE



Openness and Impact

- ▶ Department collaborated with many academic partners
- ▶ Looking for the “Twin-Win”
 - ▷ Basic research advancements
 - ▷ Solutions to practical problems



Quest for Practical Use?		No	Yes
Quest for Fundamental Understanding?	Yes	Pure Basic Research (Bohr)	Use-Driven Basic and Applied Research (Pasteur)
	No		Pure Applied Research (Edison)

Pasteur's Quadrant

Staffing Focused on Three Fundamental Capabilities

- ▶ Empirical data
- ▶ Data visualization
- ▶ Algorithms/Formal methods



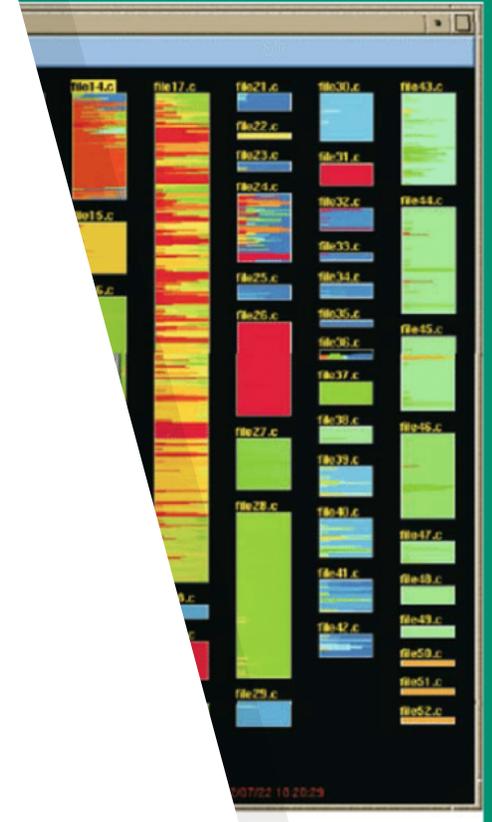
Empirical Data

- ▶ Key tool for understanding problems, tracking progress towards a solution, and quantifying the solution's value
- ▶ Twin Win
 - ▷ Research: The Code Decay Project
 - ▷ Practice: Analysis methods for evaluating new tools



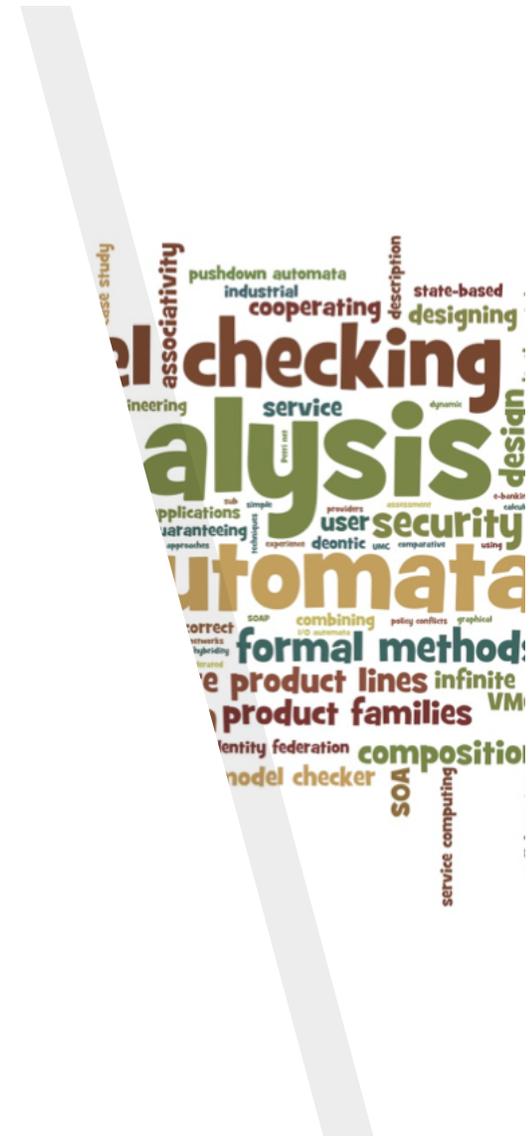
Data Visualization

- ▶ Need ways to explore and make sense of data
- ▶ Twin Win
 - ▷ Research: SeeSoft widely cited
 - ▷ Practice: SeeSoft eventually the basis for a spin-off company



Algorithms/Formal Methods

- ▶ Embodies automatable technical insight
- ▶ Twin Win
 - ▷ Research: Verisoft model checker
 - ▷ Practice: Used Verisoft to analyze and improve CDMA call-processing software



SPRD's Success in Pasteur's Quadrant

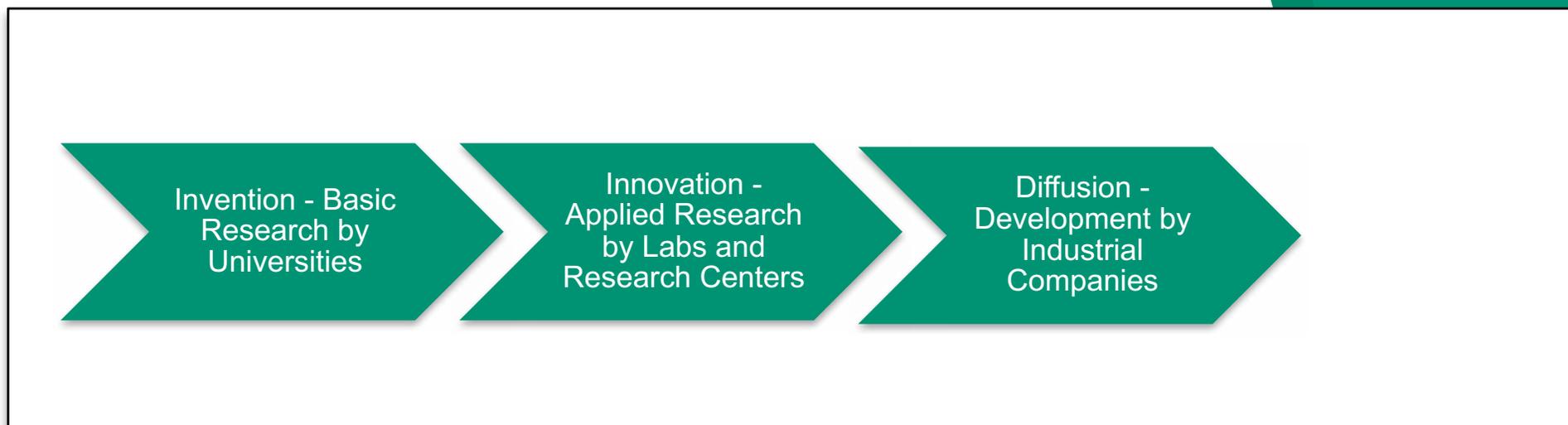
- ▶ SPRD merged innovative, validated theory with practical solutions
- ▶ A truly unique and successful research environment
 - ▷ Publishing highly influential research
 - ▷ Developing, applying and commercializing research output
 - ▷ Developing researchers who continue to have substantial success in academic and industrial positions

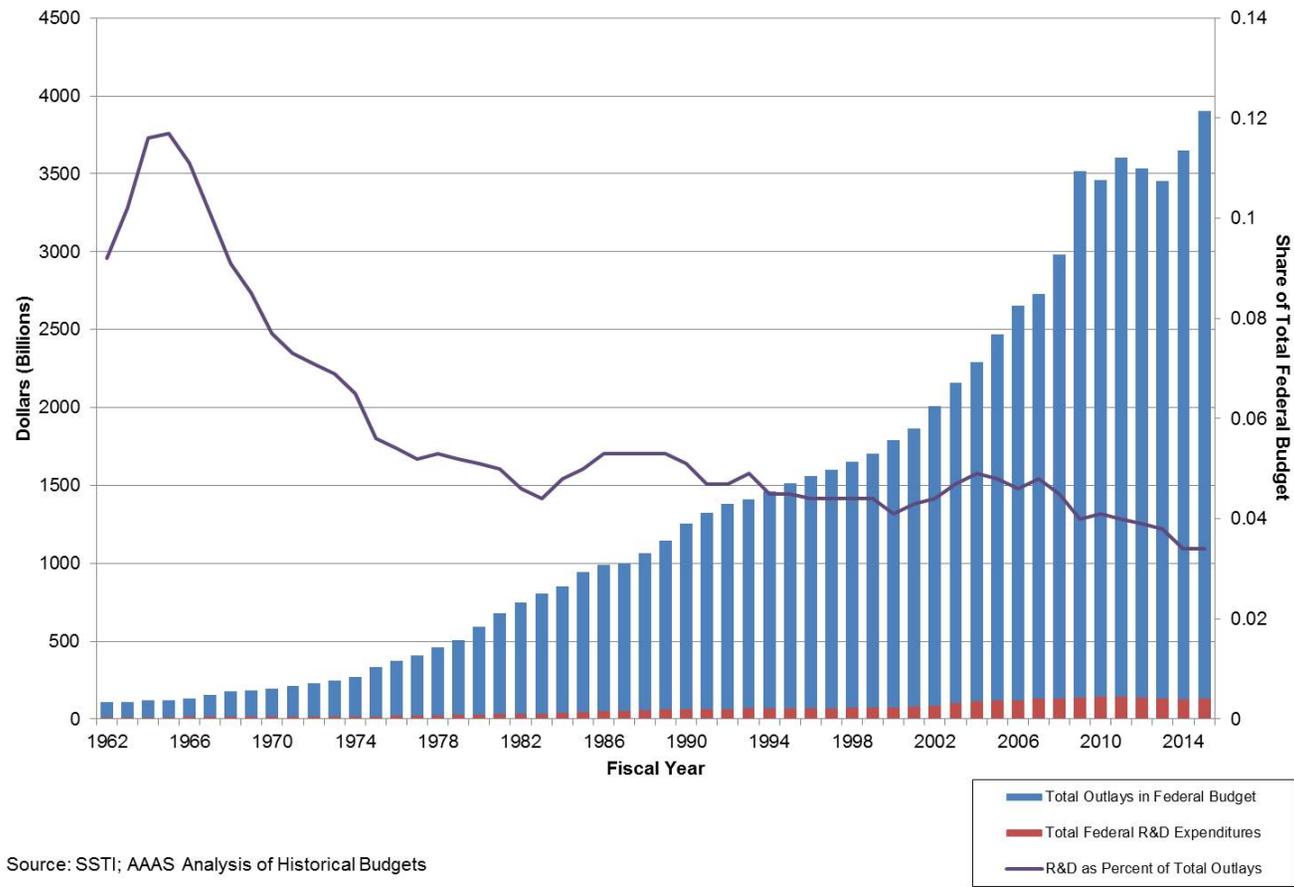
My Main Points

- ▶ Use-driven research is interesting, challenging and measurably valuable
- ▶ **Funding realities are changing; clearer value demanded**
- ▶ Examples of use-driven research (**models help**)



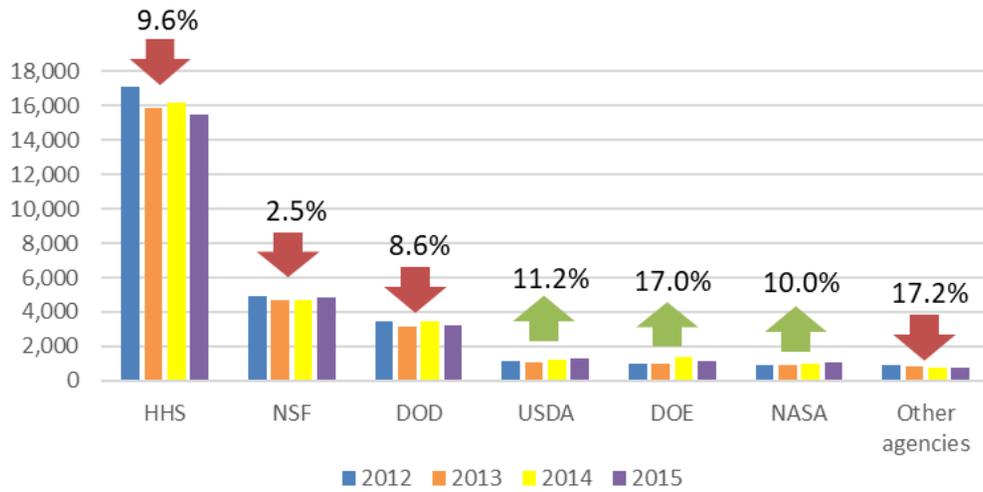
Linear Model of Innovation





Federal R&D Expenditures as a Share of US Budget at Lowest Point in 60 Years

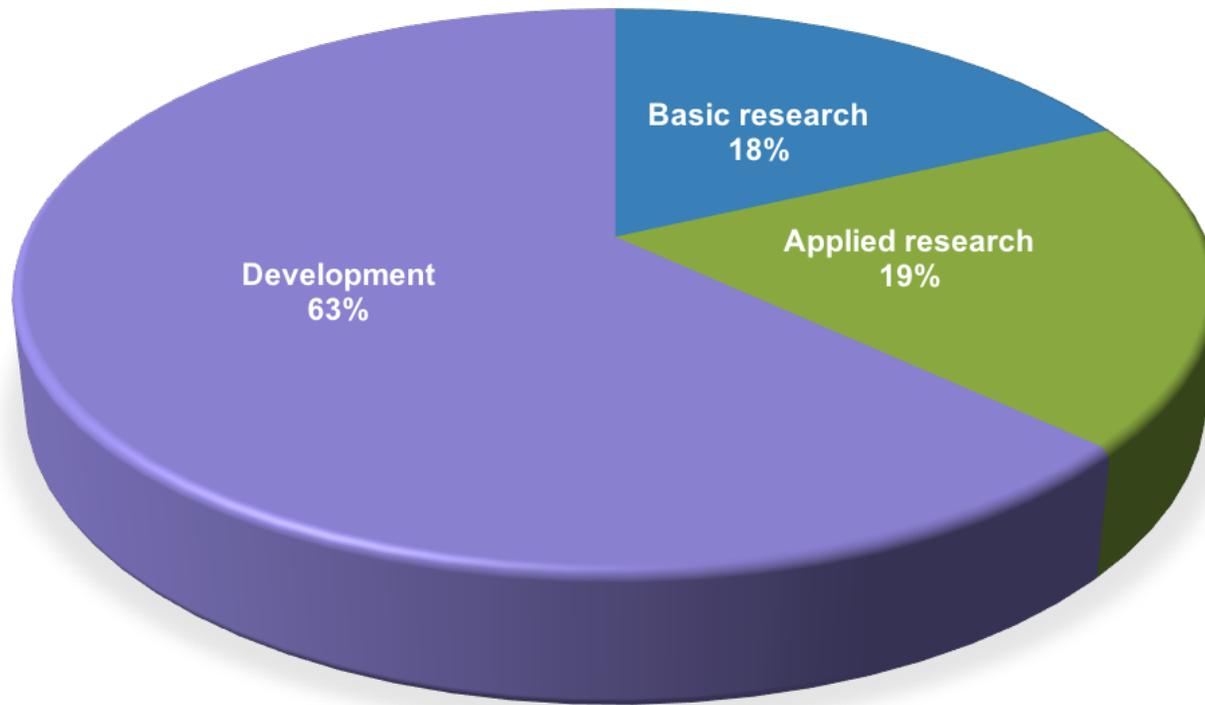




Source: NSF, NCSES, Survey of Federal Science & Engineering Support to Universities, Colleges, & Nonprofit Institutions.

Federal Funding is Flat to Decreasing in Several Agencies That Traditionally Fund CS and Engineering

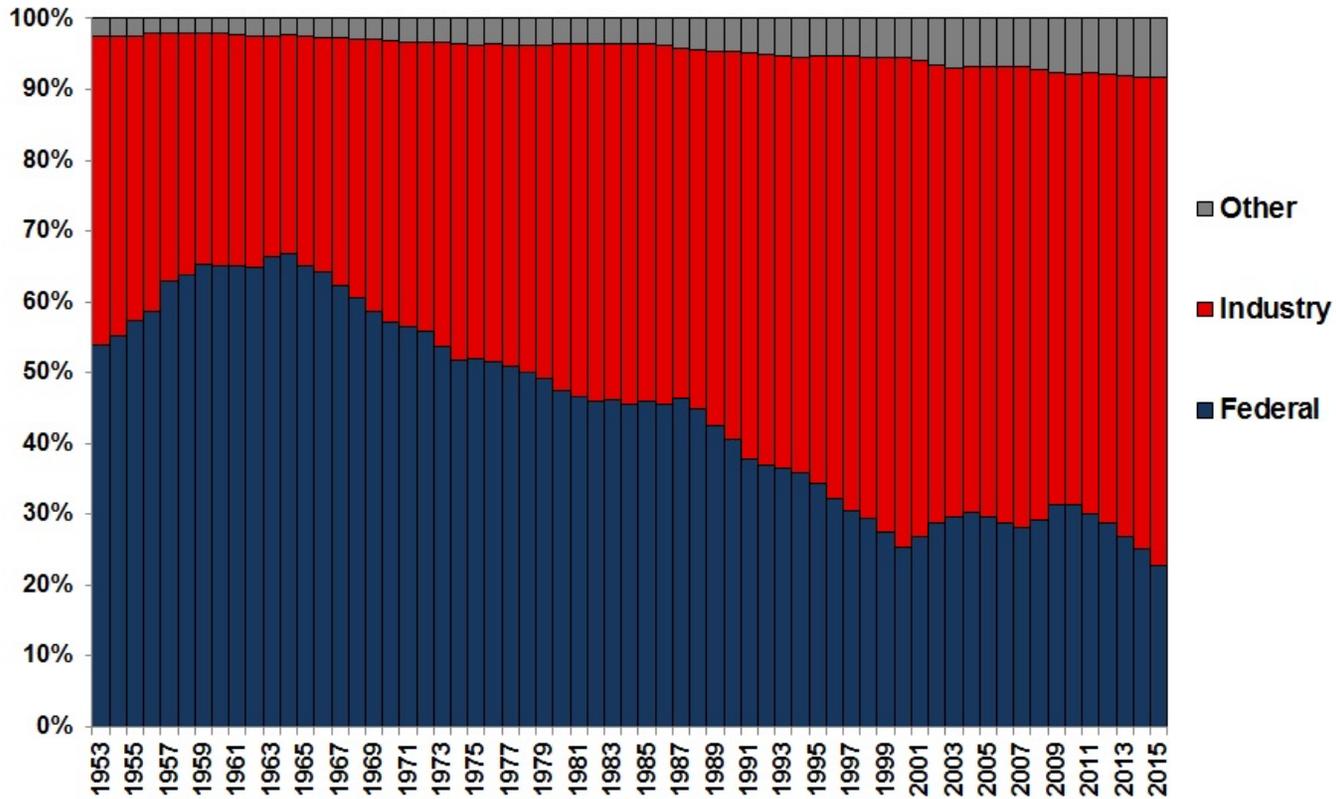




Source: National Science Board's *Science and Engineering Indicators 2014*

Basic Research Accounts for Less Than 1/5th of Total US R&D Spending

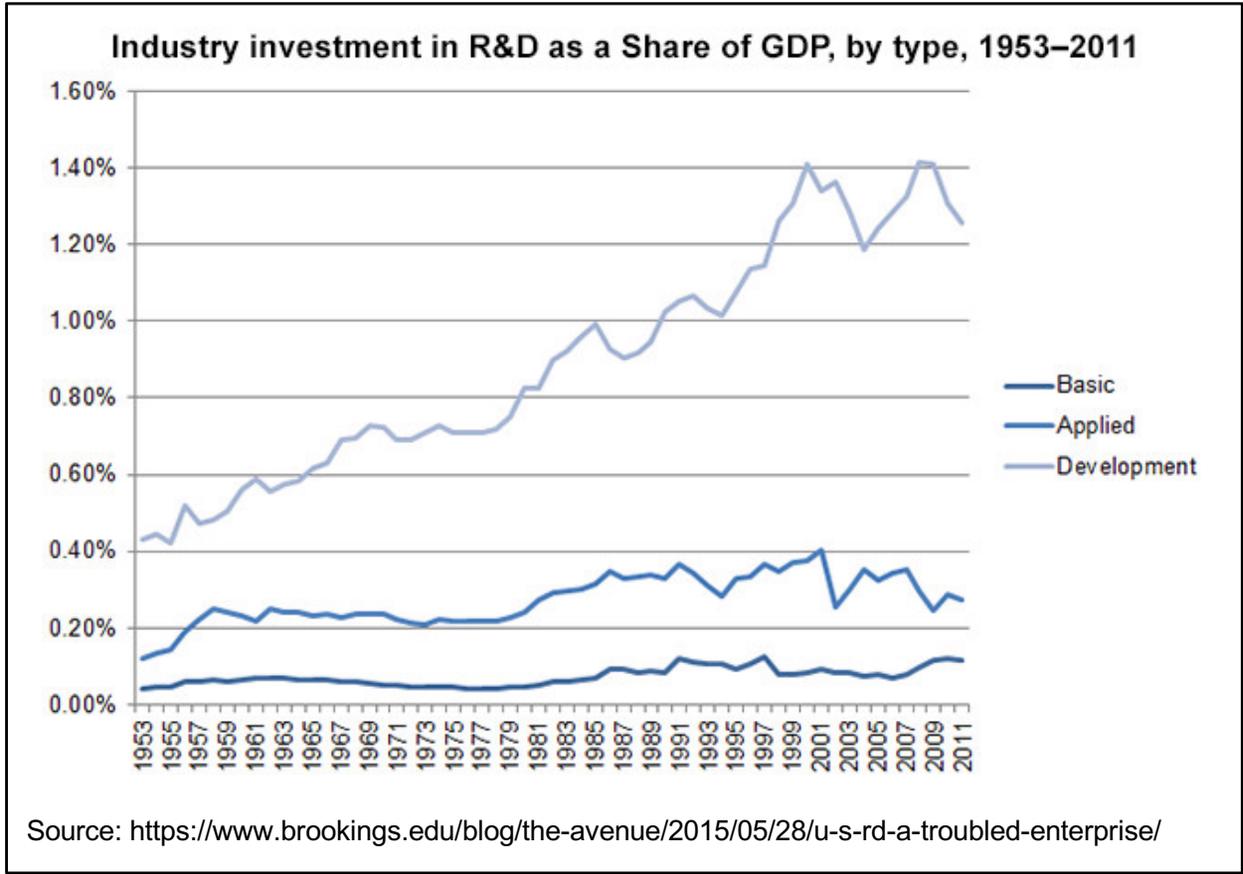




Source: National Science Foundation, *National Patterns of R&D Resources* series. Constant-dollar conversions based on GDP deflators from *Budget of the U.S. Government FY 2017* © 2016 AAAS

Industry is the Largest Source of US R&D Funding





Basic Research is not Industry's Top Priority



The Decline of Basic Research in Corporate R&D

- ▶ *Killing the Golden Goose? The Decline of Science in Corporate R&D*, Ashish Arora, Sharon Belenzon, and Andrea Patacconi
- ▶ Basic research share of private R & D has fallen significantly
- ▶ Competitive pressures are pushing firms to:
 - ▷ Narrow their scope and outsource non-core functions, including research and development
 - ▷ Place greater emphasis on the “D” in R&D and favor incremental innovation
 - ▷ Rely more on research-intensive organizations and startups (and indirectly Universities) for innovation

The Business of Research is Changing

- ▶ Traditional sources of basic research funding are shrinking
- ▶ Industry funds are being directed to more applied areas with shorter payback times
- ▶ Portfolio view of funding streams is helpful
- ▶ Industry companies are more open to partnerships than ever

My Main Points

- ▶ Use-driven research is interesting, challenging and measurably valuable
- ▶ Funding realities are changing; clearer value demanded
- ▶ **Examples of use-driven research (models help)**



Eye Opening Quote #1

“Adam, I have 200 researchers on contract around the world. I don’t need another research hobby. I need a solution.”

Corporate Executive
Large Medical Device Company

Eye Opening Quote #2

“Want to help someone? Shut up and listen!”

Ernesto Sirolli

Some Useful Strategies

- ▶ Be a servant
- ▶ Build teams to solve broad problems
- ▶ Leverage models for support cross-domain work and communication
- ▶ Create something reusable in each project: ideas, tools and services

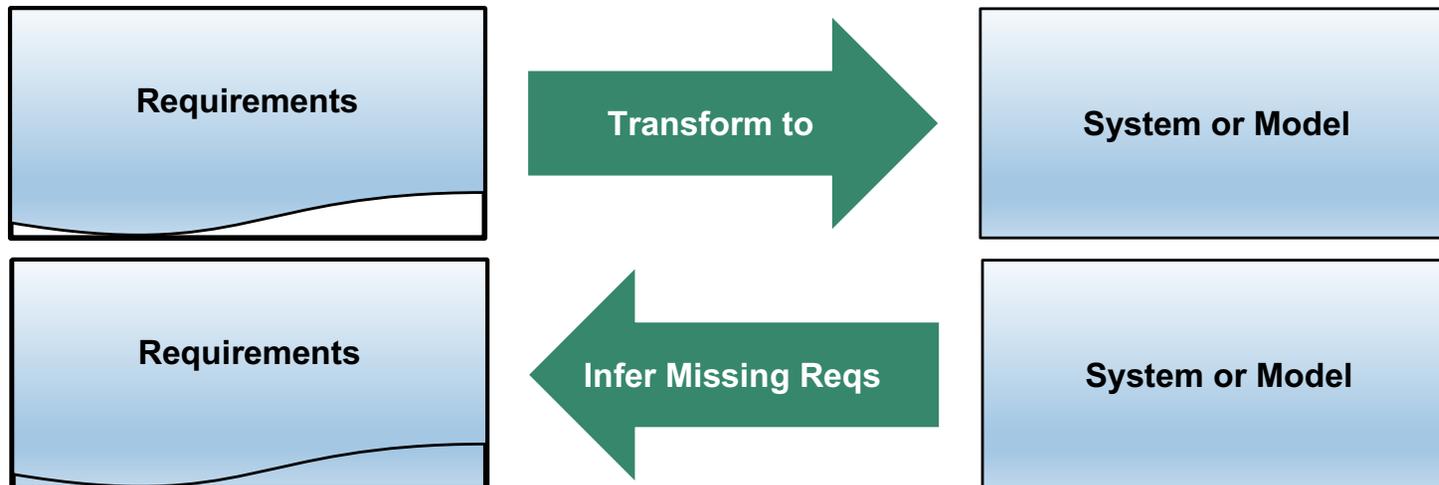
Automatically Learning Specifications

- ▶ Automotive engineering company wanted to develop their model-driven engineering capabilities
 - ▷ Based on natural language specs with many implicit requirements



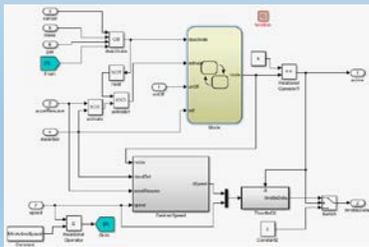
Automatically Learning Specifications

- ▶ Can we semi-automatically identify implicit / inconsistent requirements?



Automatically Learning Specifications

Models or Source Code
of Automotive Devices



Automated Test Generation
Static Analysis
Data Mining

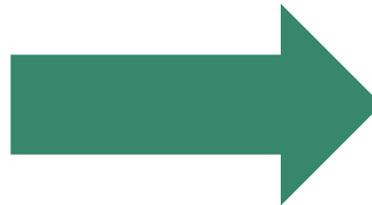
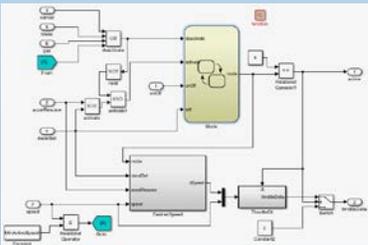
Invariants Describing
System Behavior

`brake = pressed -> mode = inactive`

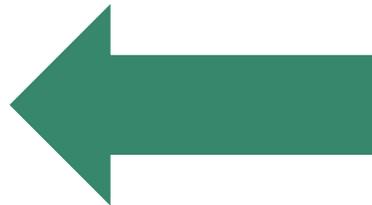
`speed < 26 mph -> mode = inactive`

Automatically Learning Specifications

Models or Source Code
of Automotive Devices



Automated Test Generation
Static Analysis
Data Mining



Additional Test Generation
to verify and refine
inferred specifications

Invariants Describing
System Behavior

brake = pressed -> mode = inactive

~~speed < 26 mph -> mode = inactive~~

speed < 25 mph -> mode = inactive

Automatically Learning Specifications



- ▶ Analysis of Resulting Specifications
- ▶ Requirement #1: The brake shall always deactivate the cruise control.
 - ▷ **brake = pressed -> mode = inactive**

Automatically Learning Specifications



- ▶ Analysis of Resulting Specifications
- ▶ Requirement #1: The brake shall always deactivate the cruise control.
 - ▷ **brake = pressed -> mode = inactive**



- ▶ Requirement #2: The cruise control shall be inactive whenever vehicle speed is less than 30 mph.
 - ▷ **speed < 25 mph -> mode = inactive**

Automatically Learning Specifications

- ▶ Developed tool for inspecting learned specifications
 - ▷ Filtering results
 - ▷ Creating state machines

1 / 1

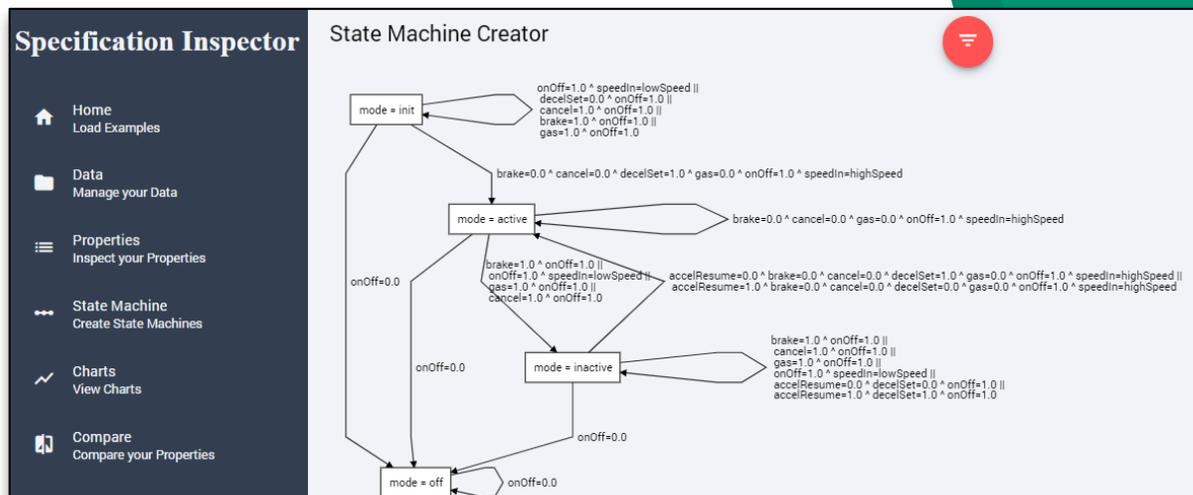
brake == 1.0

mode == inactive

1 / 1

speedIn < 25.0

mode == inactive

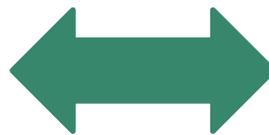
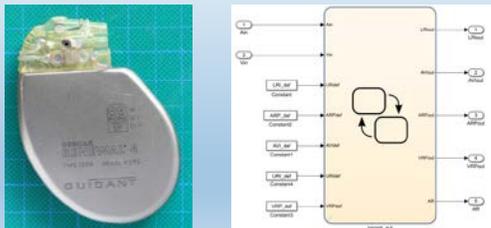


CyberCardia

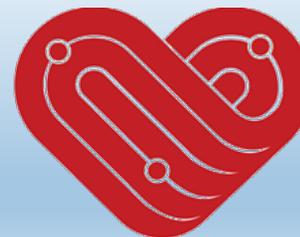
- ▶ NSF Center initiative to advance the state-of-the-art in Cyber-Physical Systems (CPS)
 - ▷ Developing novel verification techniques for CPS
 - ▷ Developing realistic heart models that can be used with in-the-loop testing of medical devices
- ▶ Goal: Reduce the need for trials involving humans and animals



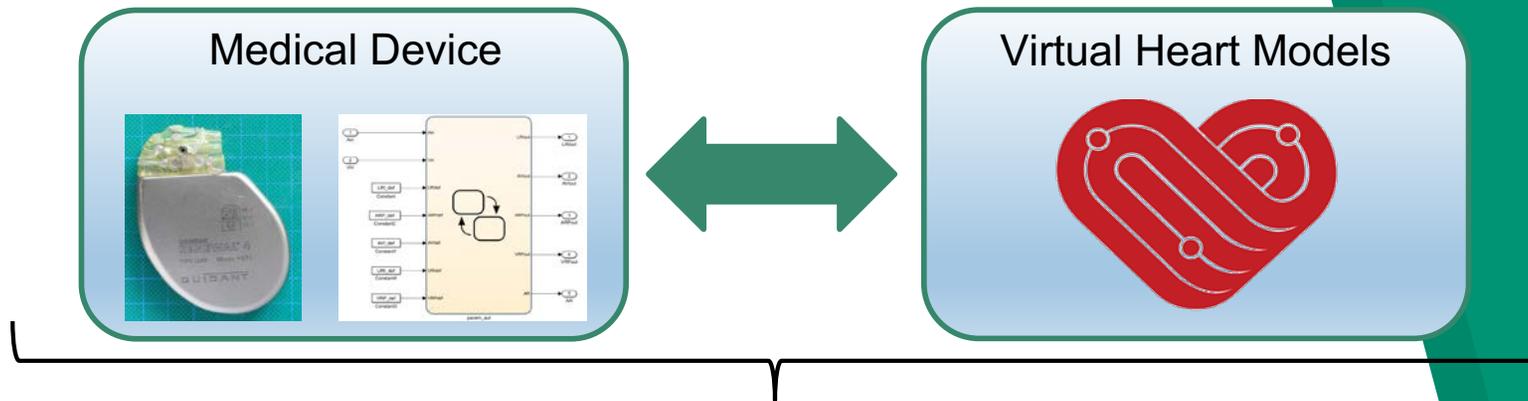
Medical Device



Virtual Heart Models



CyberCardia: Our Contribution



- ▶ Specification learning of Medical Devices and Heart Models
 - ▷ e.g. Analyze requirements for a blood infusion pump
- ▶ Requirement Ex.: If blood pressure is low and pulseOx value is good, then resume infusion and clear the warning alarm
 - ▷ `bloodPressure = good -> warning = 0`
 - ▷ `bloodPressure = good & criticalStop = 0 -> infusion = 1`
 - ▷ `bloodPressure = good & reset = 1 -> infusion = 1`

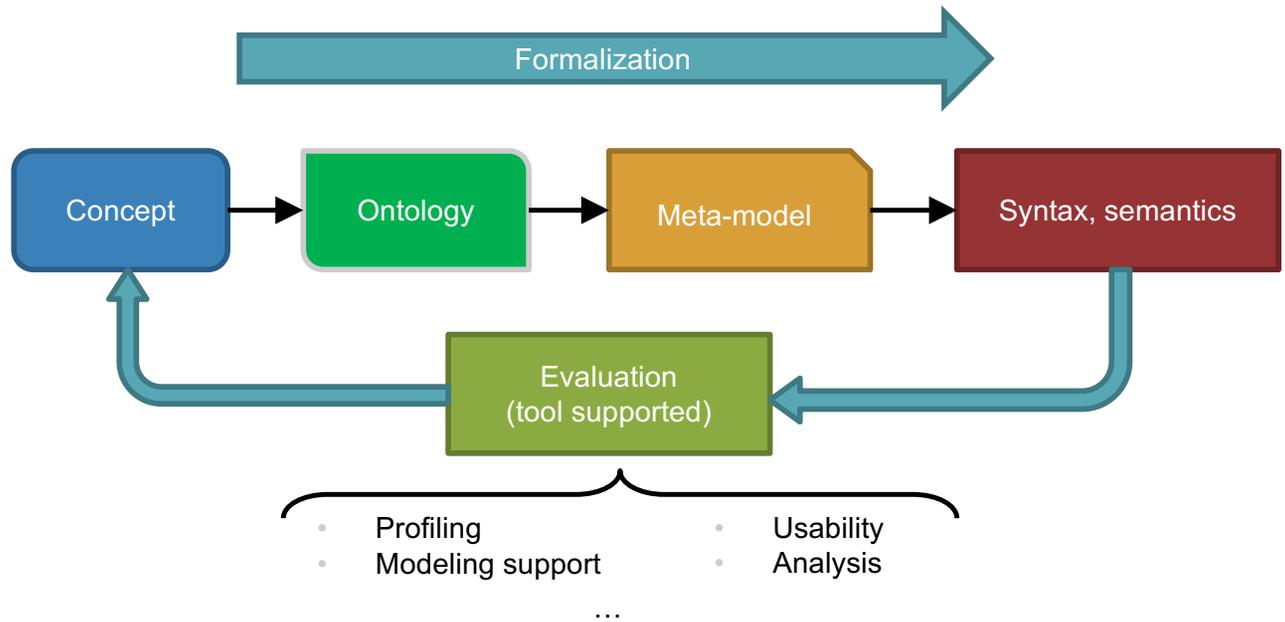
Some Useful Strategies

- ✓ Be a servant
- ✓ Build teams to solve broad problems
- ✓ Leverage models for support cross-domain work and communication
- ✓ Create something reusable in each project: ideas, tools and services

MBE Beyond Simulink, UML/SYSML

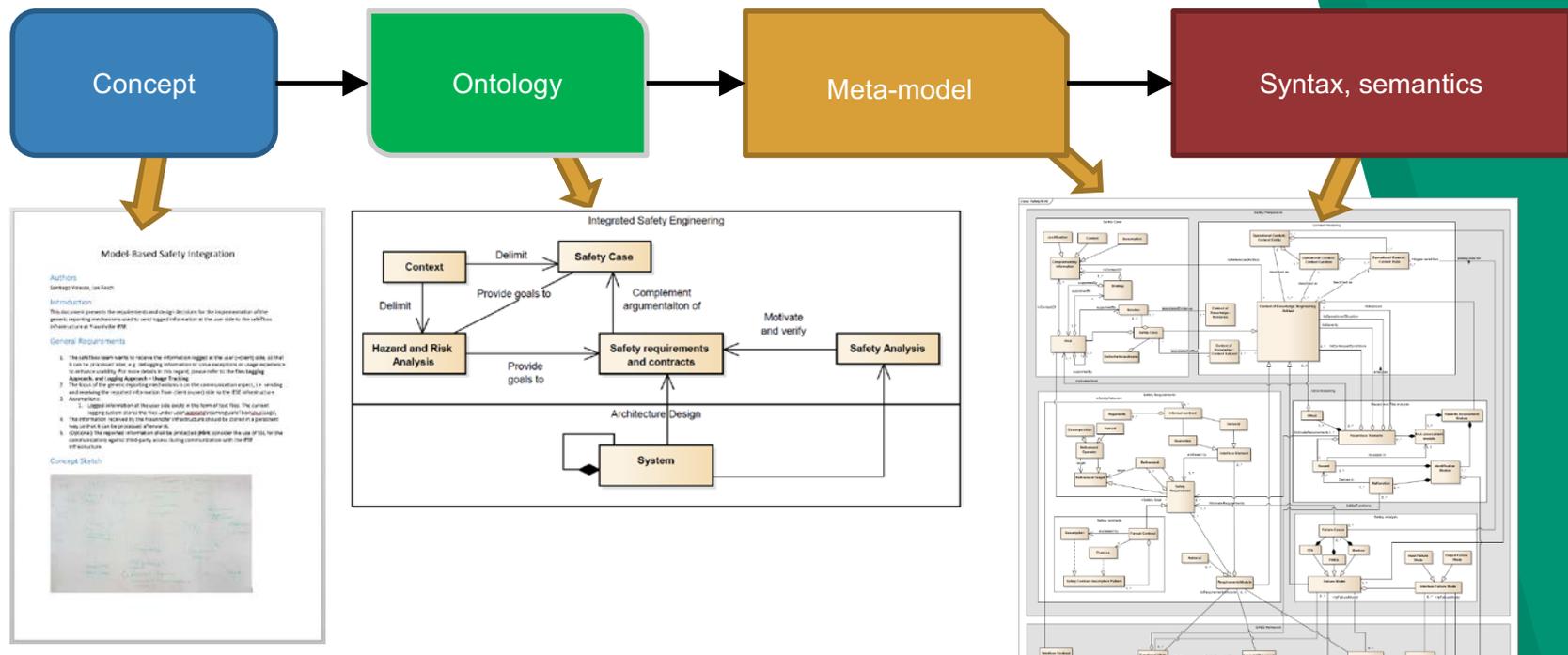
- ▶ Work with automotive manufacturer who uses component-level Simulink-style functional models
- ▶ Systems continually getting larger and more complex
 - ▷ Needs models of multiple system aspects
 - ▷ Wants to analyze dependencies across models
- ▶ Example: Advanced Driver Assistance Systems (ADAS) in modern cars provide certain functionality, but must meet functional and non-functional safety goals and standards

MBE - Agile Realization Approach



Integrated Safety Engineering

- ▷ Goal: Integrate architecture and safety artifacts
- ▷ Benefit: traceability, maintainability, automation, simulation



Analysis Example

- Opens up new possibilities for automated quality analysis

Completeness Check

- Requirements vs. Failure modes

The screenshot shows the ReqsFmConsistency tool interface. It includes a toolbar with 'Delete', 'Remove From Matrix', and 'Change Axe'. The 'Criteria' section is set to 'SafetyCriticalRequirement' with a scope of 'Drag elements from the Model Browser' and 'Dependency Criteria' set to 'Failure Modes'. A legend on the left identifies 'Failure Modes'. The main area displays a fault tree diagram with components 'CompA.cft' and 'CompB.cft', and internal errors 'A Internal Error 1', 'A Internal Error 3', and 'B Internal Error'. Below the diagram is a dependency matrix table:

	1	1	1	1	1	1
FSR 1	1	1	1			
FSR 2	2	2		1	1	
FSR 3	1				1	1
FSR 4	1				1	1
FSR 5	1			1		1

Qualitative and Quantitative Fault Tree analysis

- Minimal Cut Sets
- Event probability

The screenshot shows the Prime Implicants analysis window. It includes a 'Clone Dialog' button and a checked 'Show Negative Implicants' checkbox. The main area displays two tables:

Prime Implicants

#	Order	Prime Implicant
0	1	A Internal Error 1
1	2	A Internal Error 3, A Internal Error 3
2	1	B Internal Error

Individual Prime Implicants

#	Positive	Name	Parent Path
0	true	A Internal Error 1	

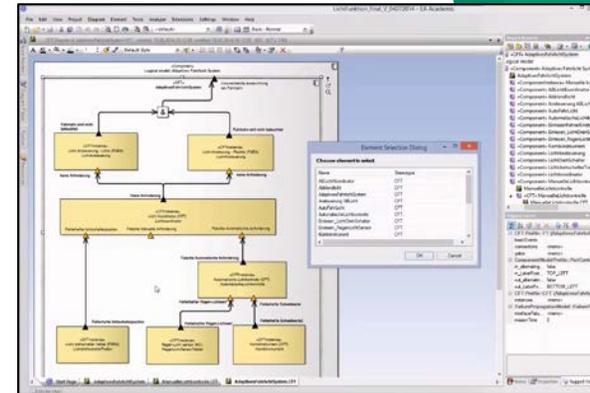
Implementation System

Profiles



+ C# Add-ins =

Modeling Tool



- No need to re-invent the wheel
- Fast prototyping
- Product definition is harder

- SysML integration
- Component Fault Trees
- Hazard and Risk Assessment
- Safety Concepts
- ...

► SafeTbox - Enterprise Architect extension

Useful Strategies

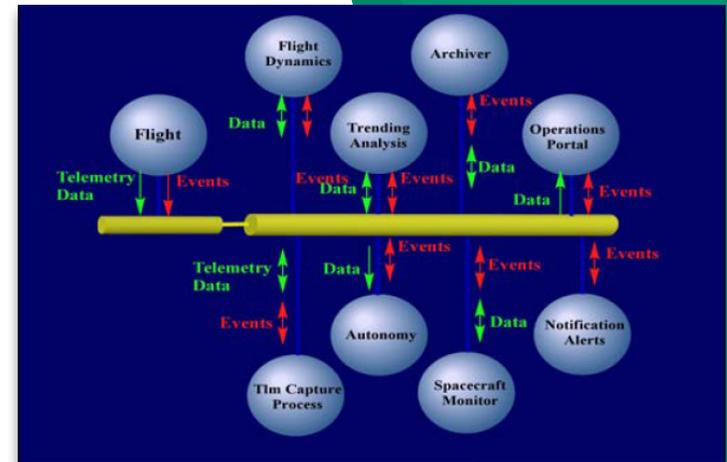
- ✓ Be a servant
- ✓ Build teams to solve broad problems
- ✓ Leverage models for support cross-domain work and communication
- ✓ Create something reusable in each project: ideas, tools and services

Augmenting Test Suites Using Models

- ▶ Many engineering companies are beginning to transition to using model-driven approaches
 - ▷ They can't make the switch overnight
 - ▷ Considerable assets exist supporting the status quo

NASA GMSEC

- ▶ Publish/Subscribe middleware used to interconnect space system components
- ▶ Had some existing test cases, but very incomplete coverage of requirements



Manually-Constructed Test Case (Partial)

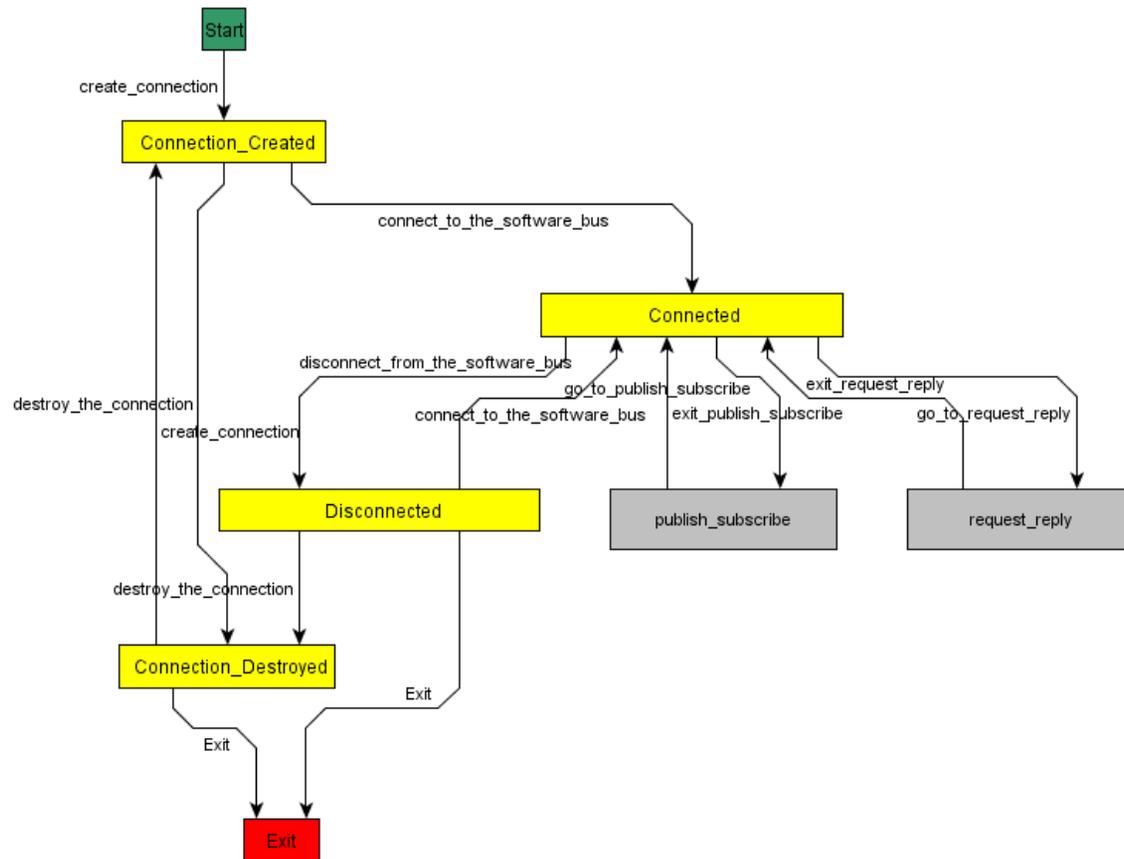
```
public static void main( String args[] ) {
    Status result = new Status();
    Connection conn = new Connection();
    ConnectionConfig cfg = new ConnectionConfig( args );

    // Create the connection
    result = ConnectionFactory.Create( cfg, conn );
    checkError( false, result, "Creating the connection object" );

    // Disconnect
    result = conn.Disconnect();
    checkError( true, result, "Disconnecting before connection is established" );

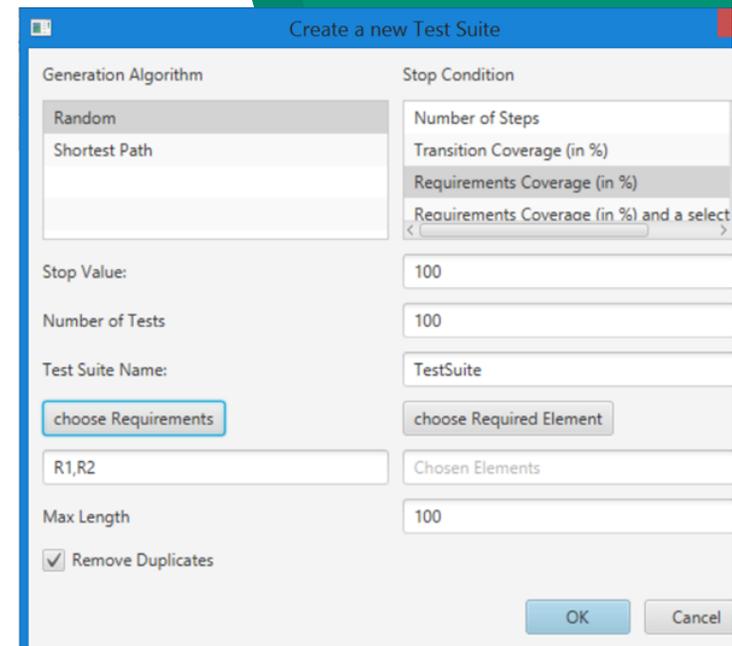
    // Connect
    result = conn.Connect();
    checkError( false, result, "Establishing the connection to the middleware" );
} //..main()
```

GMSEC Top-Level Testcase Model



Generating Test Cases from Test Model

- ▶ Tester specifies test case criteria
 - ▷ e.g. requirements coverage
- ▶ Tester specifies test suite criteria
 - ▷ e.g. 1,000 test cases
- ▶ Model Walker traverses model
 - ▷ A test case is a traversal from start node
 - ▷ Continues until stopping criteria met
 - ▷ Outputs executable test cases



The screenshot shows a dialog box titled "Create a new Test Suite" with two main sections: "Generation Algorithm" and "Stop Condition".

Generation Algorithm:

- Random (selected)
- Shortest Path

Stop Condition:

- Number of Steps
- Transition Coverage (in %)
- Requirements Coverage (in %) (selected)
- Requirements Coverage (in %) and a select

Stop Value: 100

Number of Tests: 100

Test Suite Name: TestSuite

choose Requirements: R1,R2

choose Required Element: Chosen Elements

Max Length: 100

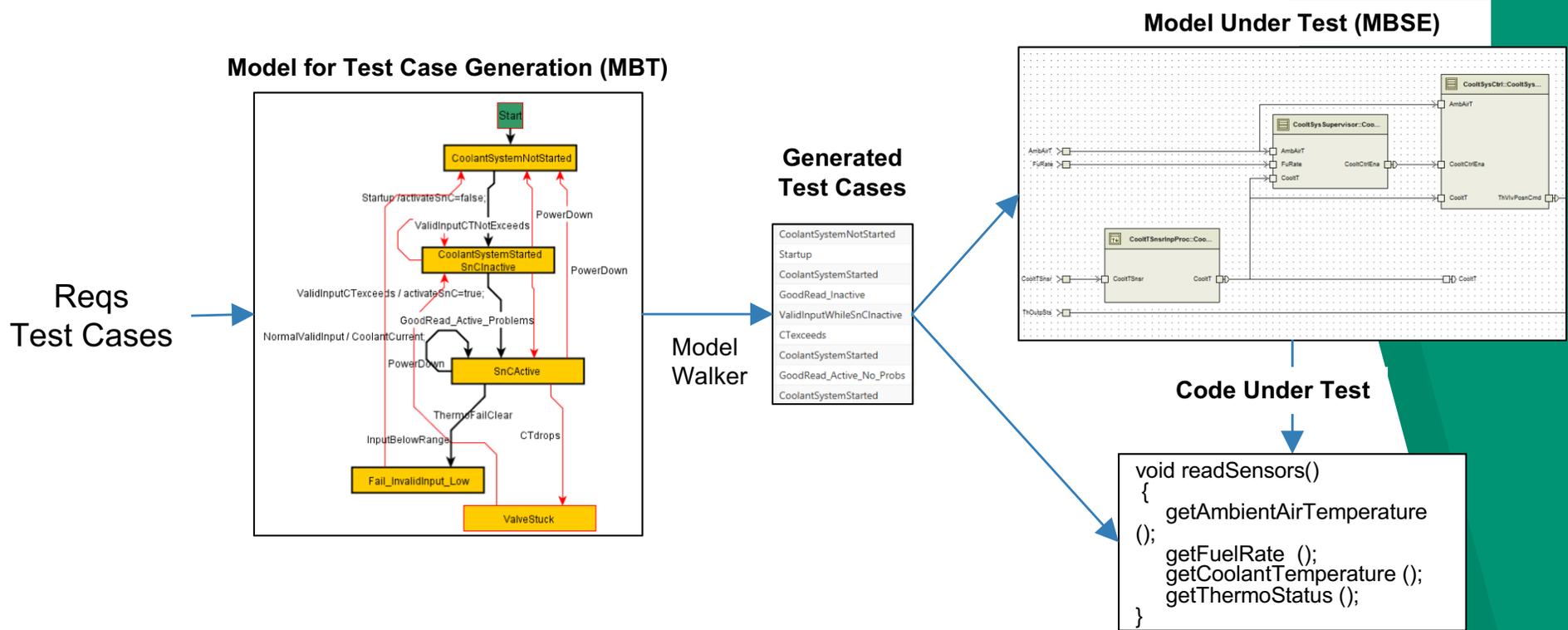
Remove Duplicates

Buttons: OK, Cancel

New Test Cases Add High Value at Low Cost

- ▶ The new tests found many defects not found by the original test cases
- ▶ The new test cases were based only information latent in the original test cases, yet covered many more combinations of nominal and off-nominal behavior

Generating Test Cases from Generated Test Models



Supporting Porting to AUTOSAR Platform

- ▶ Major heavy vehicle manufacturer is changing OS platforms
- ▶ Our plan
 - ▷ Generate test model(s) from existing application test cases
 - ▷ Generate new abstract test cases from test model
 - ▷ Iteratively execute and refine test cases against original OS
 - ▷ Based on performance, coverage and learned specifications
 - ▷ Execute vetted test cases against AUTOSAR application

Some Useful Strategies

- ✓ Be a servant
- ✓ Build teams to solve broad problems
- ✓ Leverage models for support cross-domain work and communication
- ✓ Create something reusable in each project: ideas, tools and services

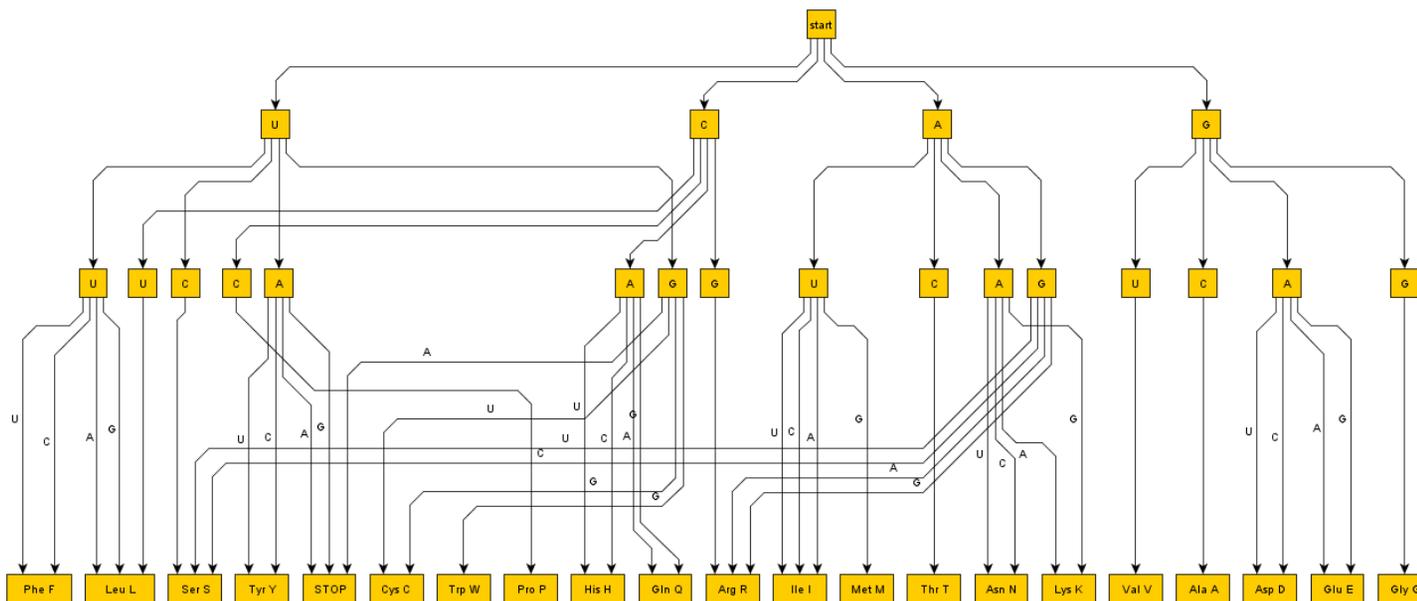


Identifying Biological Threats

Project Team

- ▶ Designing, building, validating computational methods for identifying biological threats
- ▶ Large project team combining expertise in biology, bioinformatics and software and systems engineering
 - ▷ Fraunhofer supporting architectural design, agile development and validation

Models of DNA sequences



Some Useful Strategies

- ✓ Be a servant
- ✓ Build teams to solve broad problems
- ✓ Leverage models for support cross-domain work and communication
- ✓ Create something reusable in each project: ideas, tools and services

Conclusions

- ▶ Use-driven research is interesting, challenging and measurably valuable
 - ▷ Don't artificially limit yourself to one kind of research style
- ▶ Funding realities are changing; clearer value demanded
 - ▷ Don't be an ostrich!
- ▶ You can do use-driven research
 - ▷ Listen to customers, solve their problems, build teams, use models

Questions?

Adam Porter

Professor: University of Maryland
Executive Director: Fraunhofer CESE
aporter@cs.umd.edu
aporter@cese.fraunhofer.org



Photo Credits

- ▶ Pencils - Photo Credit:
- ▶ Zot on - <https://www.flickr.com/photos/50854518@N05/10201180195/>
- ▶ Crash test - <https://www.flickr.com/photos/54519810@N02/7920091208/>
- ▶ Tower -<https://www.flickr.com/photos/104342908@N08/35897610470/>
- ▶ UMD aerial - <https://www.flickr.com/photos/126497846@N03/14601582999/>
- ▶ ATT Logo -
https://upload.wikimedia.org/wikipedia/commons/thumb/3/31/AT%26T_logo_2016.svg/1200px-AT%26T_logo_2016.svg.png
- ▶ Word cloud - <http://fmt.isti.cnr.it/~mtbeek/header/wordle42.jpg>
- ▶ Test tube - <https://tanyagrove.files.wordpress.com/2011/05/test-tubes.jpg>
- ▶ WhiteSands - Dollarphotoclub_52522495.jpg
- ▶ Infusion pump - <https://www.flickr.com/photos/44312356@N04/9250811463>