Government Microelectronics Assessment for Trust (GOMAT)

Melanie D. Berg[1], Kenneth A. LaBel[2]

1. AS&D, Inc., Work performed for NASA GSFC, melanie.d.berg@nasa.gov

2. NASA GSFC

To be presented by Melanie Berg at the National Defense Industrial Association Field Programmable Gate Array Assurance Workshop***, McLean, VA February 28th 2018***

Acronyms

- Application specific integrated circuit (ASIC)
- Defense Microelectronics Activity (DMEA)
- Electronic Design Automation (EDA)
- Framework for Assessing Security and Trust in MicroElectronics (FASTIME)
- Field programmable gate array (FPGA)
- Government Microelectronics Assessment for Trust
- Intellectual Property (IP)
- Information Technology (IT)
- Input/Output (I/O)
- Model Based Mission Assurance (MBMA)
- NASA Electronic Parts and Packaging (NEPP)
- Physical unclonable function (PUF)
- Register Transfer Language (RTL)
- Verification and Validation (V&V)

To be presented by Melanie Berg at the National Defense Industrial Association Field Programmable Gate Array Assurance Workshop, McLean, VA February 28th 2018

Synopsis of Assurance Plan

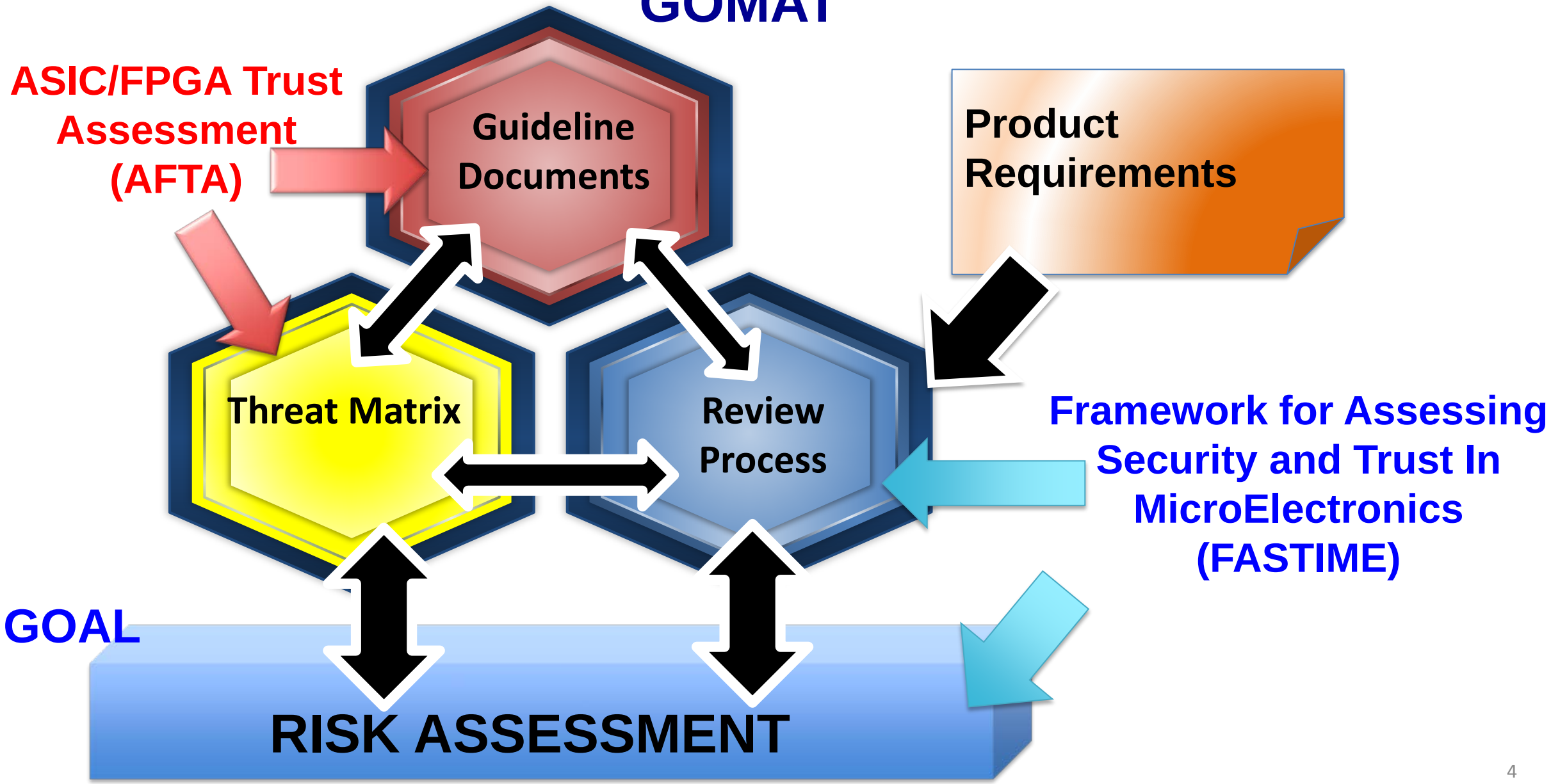
ASIC: Application specific integrated circuit

FPGA: Field programmable gate array

- The government is developing a systematic framework for practicing security and trust in ASIC and FPGA applications.
- Goal: User is provided guidance in mitigation best practices; correspondingly, missions are expected to follow guidelines to the best of their abilities; and a risk assessment is performed on implementation.
- There are three flows:
  - (1) FPGA designer flow; (2) Designer ASIC flow; and (3) FPGA supplier flow.
  - Separate with unique assurance approaches yet many similarities.
  - We will discuss high-level generalities.
- Activity and Support:
  - Government process established under Defense Production Act Title III
  - Process is currently targeting a critical mission's FPGA designer flow

The methodology incorporates work/research performed by a variety of groups: NASA, The Aerospace Corporation, RAMBUS, Global Foundries, Mentor Graphics, Synopsys, Xilinx, Graf Research, Sandia National Laboratories, and Microsemi.

Government Microelectronics Assessment for Trust: GOMAT

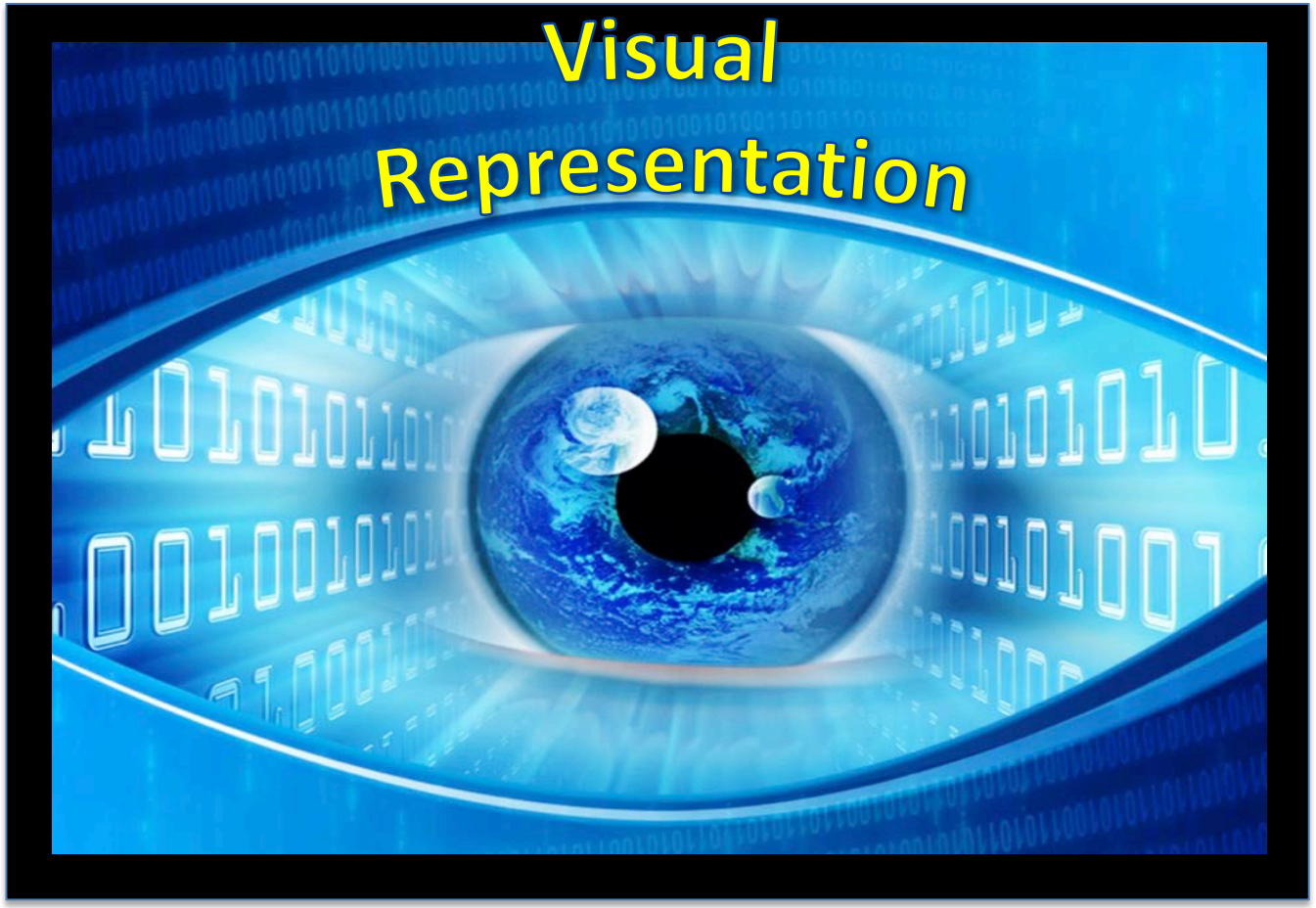


To be presented by Melanie Berg at the National Defense Industrial Association Field Programmable Gate Array Assurance Workshop, McLean, VA February 28th 2018

Modeling threats, vulnerabilities and mitigation helps to develop a framework for gathering system information. Visual direction for system evaluation.

Game Theory:

- Jonathan Graf(Graf Research)
- Brandon Eames (Sandia National Laboratories)



Reliability and Radiation:

- NASA Model Based Mission Assurance (MBMA)
- NASA Head Quarters, NASA Electronic Parts and Packaging , and Vanderbilt University.

Primary Design Cycle Vulnerabilities

Design Cycle Preparation

EDA Tools

IP Cores

Mostly External threats except for Personnel making acquisitions.

Design Cycle and Deployment

Design Data Base

EDA tools

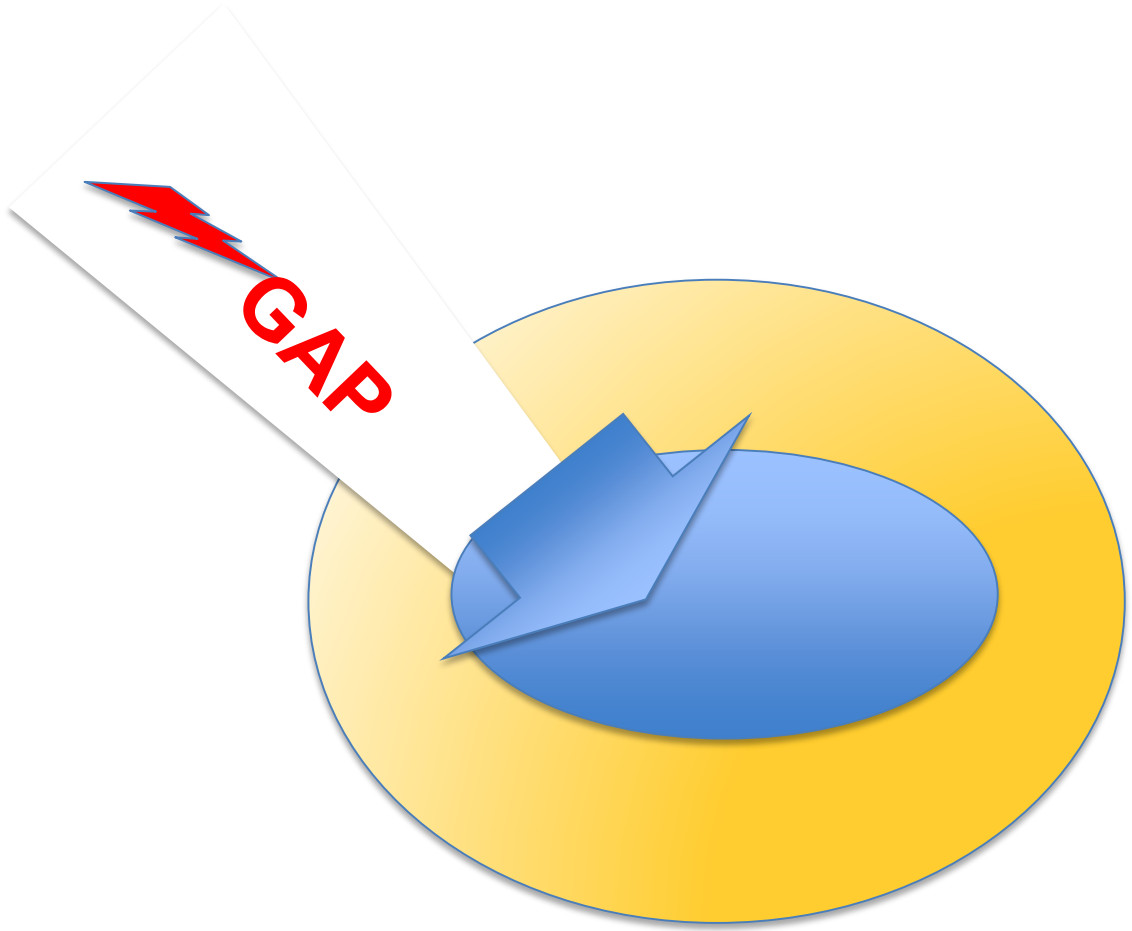
Electronics/IT

Gap in acquisition mitigation

External or internal threats.

Example: Gap in acquisition mitigation can access design environment

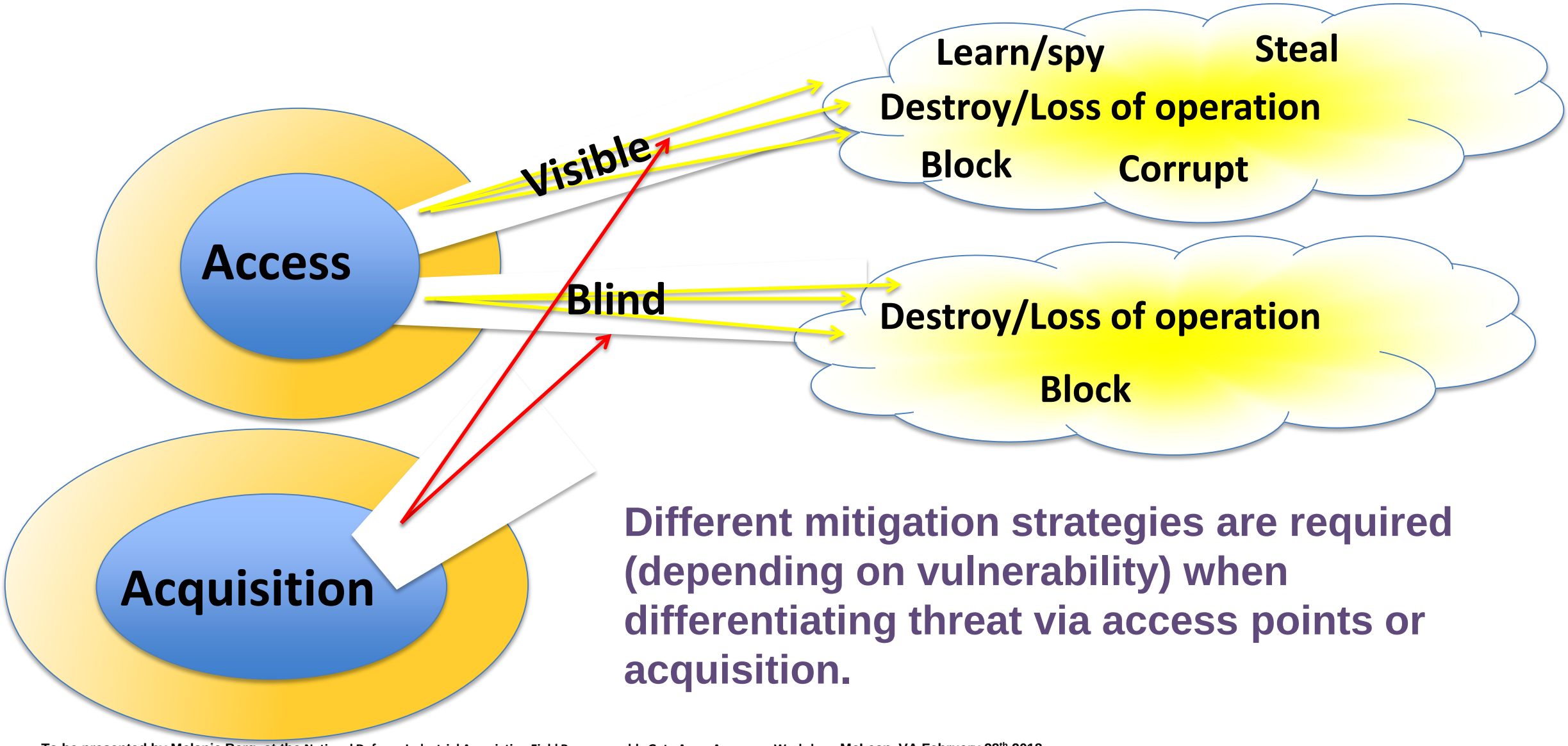
Mitigation Is Never 100% Bulletproof



- Adversary learns the system under analysis including mitigation.
- Adversary tries to detect or create gaps in mitigation.
- Adversary attacks system via gap.
- Must be taken into account in risk analysis.
- Do additional layers or dynamic layers of mitigation need to be implemented?

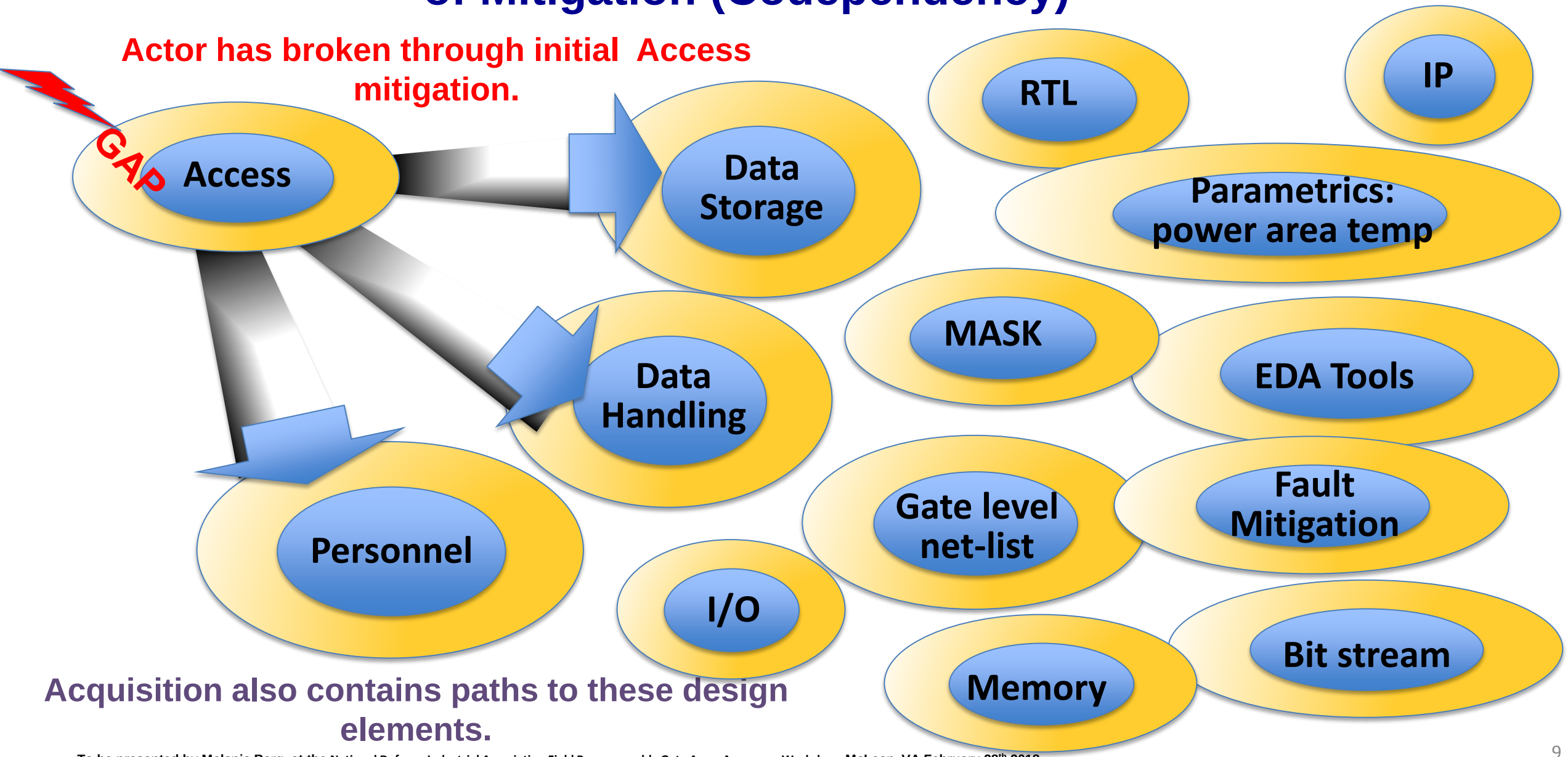
Goal is to make the likelihood to adversary infiltration take longer than a component is active.

Gaps in Mitigation:
Channels of Vulnerability and Circumstances



Different mitigation strategies are required (depending on vulnerability) when differentiating threat via access points or acquisition.

Accessibility into Internal Design Elements: Multiple Layers of Mitigation (Codependency)



Acquisition also contains paths to these design elements.

Note Mitigation Application and Strength Must Be Carefully Assessed

- Piling on mitigation can add risk.
- Mitigation complexity might have hidden modes that are blind to the review team or unreachable by the EDA tools:
  - System lock out,
  - Unwarranted self-destruct,
  - Flags that ease adversary's learning phase.

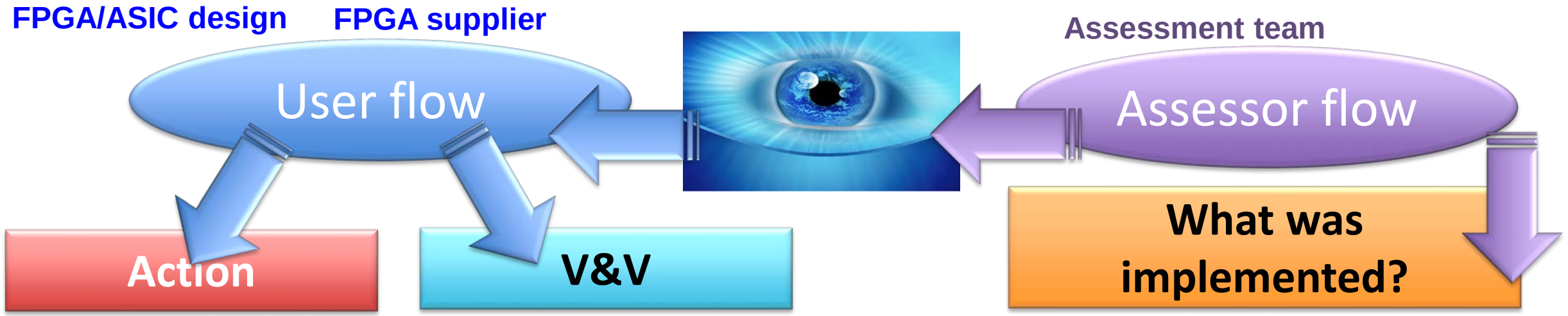
Mitigation eats access to all!!!!!!!!!!!!

When Mitigation becomes a threat!

Proposed Security and Trust Assessment Process: GOMAT

Increasing V&V is not a sufficient solution– it's about employing threat mitigation, assessing mitigation, and finding potential gaps (that exist in the mitigation and V&V coverage).

Distinct Difference Between User and Assessor Responsibilities



- User: Obtain requirements and guidelines. Implement and verify.
- Assessor: Identify known vulnerabilities. Predict potential vulnerabilities

User Flow versus Assessor Flow

User flow

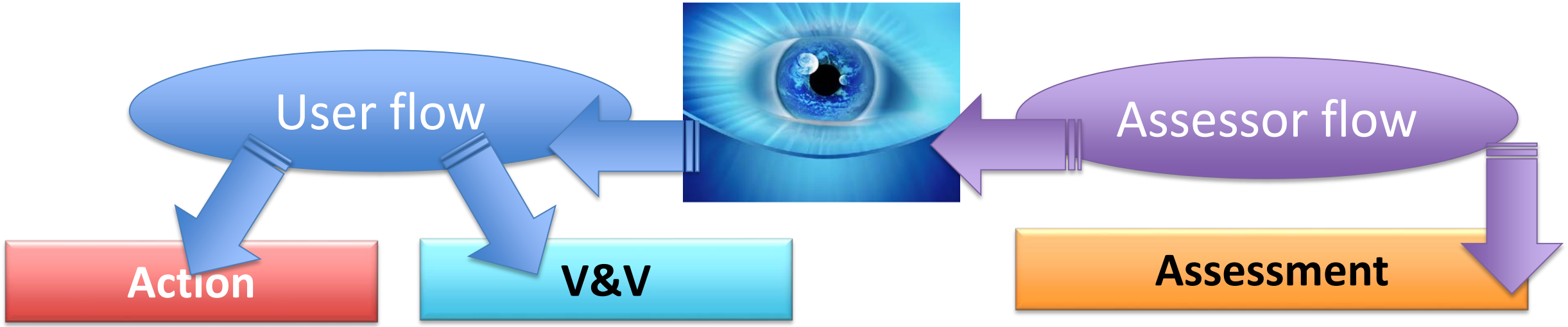
Assessor flow

- Acquisitions
- Environment
- Design
- Deployment

- V&V?
- V&V?

- Categorize User Flow
- Gather information
- Assess Implementation
- Assess V&V
- Compile Assessments

General Assessor Considerations and Goals



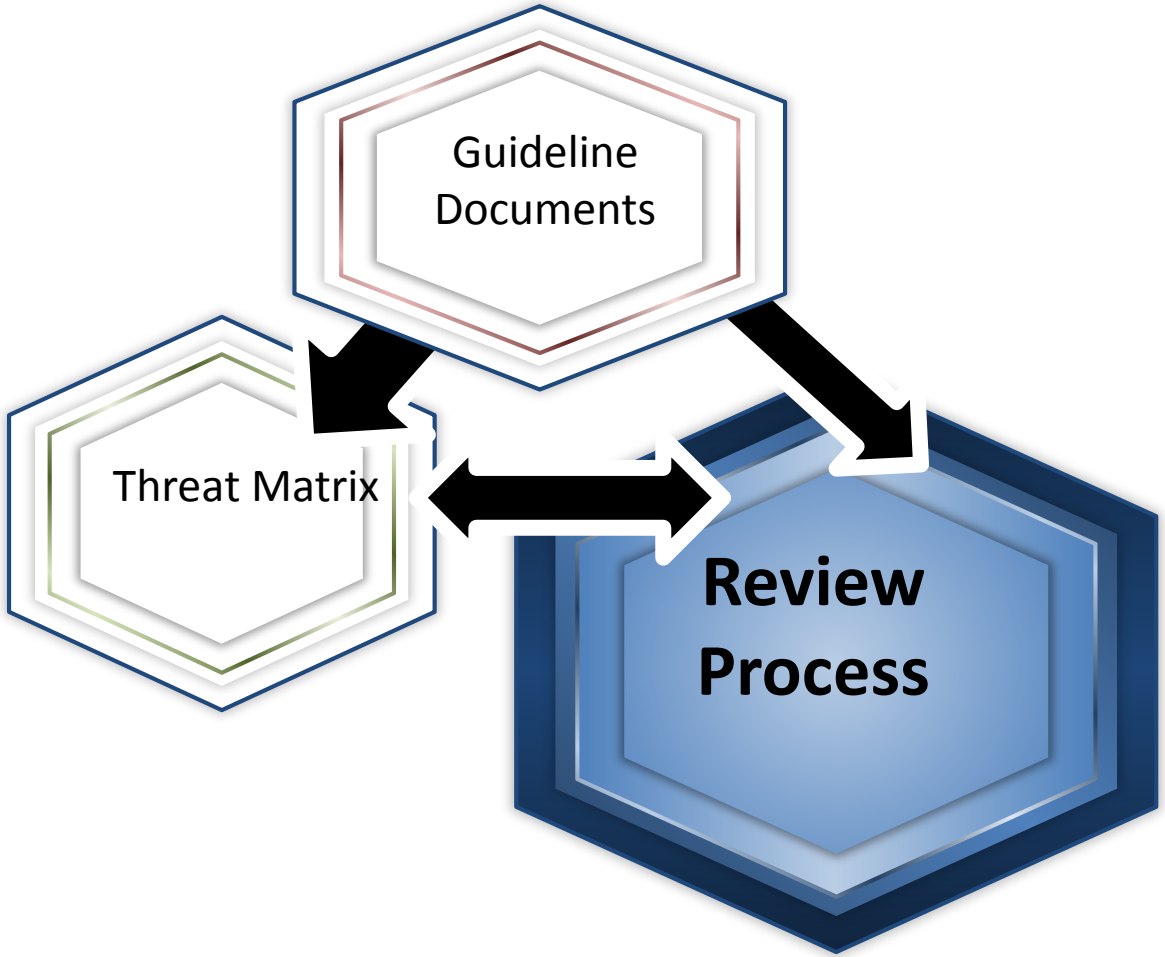
- Does design implementation need to be improved?
- Does verification coverage need to be increased?
- Do additional restrictions need to be applied?
- Have loose control compromised the product?

Identify known vulnerabilities

Predict potential vulnerabilities

FASTIME: Review Process

V&V: Verification and Validation
EDA: Electronic design automation



- Creates visibility and traceability for each step of the design process and potential contribution to threat.
- Requires an external assessment team.
- For the manufacturer's design process evaluation, it is unlikely that the trust and security assessment team will have access to all files to perform V&V.
- Hence, detailed checks of the manufacturer's V&V coverage and mitigation processes are expected to be performed by the assessment team.
- Employs established "checklist" approach.
- Enables risk analysis because of detailed information gathering.

Does not restrict EDA tools. However assesses coverage.

To be presented by Melanie Berg at the National Defense Industrial Association Field Programmable Gate Array Assurance Workshop, McLean, VA February 28th 2018

Conventional User Review Process: FPGA Specific Path and Checklist

FPGA Specific

2 players

- FPGA Checklist is created to direct **designers** (best practices).
- FPGA Checklist is used by the **design review team** to assess the following:
  - Reliability of design practice (How well does the design follow design rules).
  - Coverage of verification (how well does the design meet system FPGA requirements).
    - Requirements list/test matrix.
    - Specification documentation.

Controlled document

Checklist is used to assess how well the design and verification was implemented. It does not contain mission specifics. However, it points to a requirements matrix and assesses requirements coverage/fulfillment.

NASA Procedures and Guidelines

- 300-PG-8730.0.1, Assurance Activities for Digital Electronics for Spacecraft, Instruments, and Launch Vehicles.
  - Establishes guidelines for assurance personnel to monitor FPGA design and development activities.
- 500-PG-8700.2.8-A, FPGA Development Methodology.
  - Describes procedures and guidelines governing the creation of a robust flight FPGA development process.
  - Includes process review checklists.
- 500-PG-8700.2.7B, Design of Space Flight FPGAs.
  - Collection of best design practices for FPGA devices used for space flight designs.
  - Includes a design review checklist.
- NASA PLD Handbook NASA-HDBK-4008 - 12/02/2013 **PLD: Programmable Logic Device**
  - https://standards.nasa.gov/documents/viewdoc/3315901/3315901

Requires update!

Presented by Marco Figueiredo at the the Single Event Effects (SEE) Symposium and the Military and Aerospace Programmable Logic Devices (MAPLD) Workshop, La Jolla, CA, May 19-22, 2014 and published on nepp.nasa.gov.

To be presented by Melanie Berg at the National Defense Industrial Association Field Programmable Gate Array Assurance Workshop, McLean, VA February 28th 2018

Conventional NASA Mission Requirements Verification

- **FPGA Requirements Verification Matrix (RVM) should be created to list all requirements with their identification number and qualify them according to:**
  - Verification type (Simulation, Analysis, Inspection or Test)
  - Verification level (HDL, Card, Box, Instrument, Spacecraft)
  - Verification unit (simulation test-bench, prototype or Engineering Design Unit (EDU), proto-flight or Engineering Test Unit (ETU), or Flight Unit (FU)).
- **The verification type should be followed by a reference to where the verification can be found.**
- **Test environment should address the TEST AS YOU FLY, FLY AS YOU TEST philosophy.**

GOMAT incorporates these procedures with additional assurance for Trust and Security.

Presented by Marco Figueiredo at the the Single Event Effects (SEE) Symposium and the Military and Aerospace Programmable Logic Devices (MAPLD) Workshop, La Jolla, CA, May 19-22, 2014 and published on nepp.nasa.gov.

GOMAT is Supplemental To Conventional Reviews

- Conventional design review will evaluate fulfillment of mission requirements and general design quality. GOMAT review team takes assurance activity for Security and Trust further.
- Acquisition and access are assessed:
  - V&V performed by the target group:
    - EDA tools used are investigated.
    - Determine coverage.
    - Analyze simulation waveforms.
    - Evaluate tool report outputs.
    - Any lack in coverage is considered a risk/gap.
    - Parametric evaluations beyond normal scope.
  - FPGA configuration management and security monitoring.
  - Design environment (electronics/IT … storage and transfer).
  - Radiation effects as an adversary.
  - Personnel vetting.
  - IP Core Vetting.
  - Strict best practices (bad designs leave back doors for bad actors).
  - Assessments are conducted and further V&V is considered if necessary and able.

FASTIME Review Process: Use of An Assurance Checklist

- **Derived from NASA design review checklist and information gathered from partnering organizations.**
- **Assessments are divided into subcategories with associated risks.**
- **Links to previously assessed items are included (do not want to spend time on vetted items if its listed risk-level is acceptable).**
- **New column is added to link to Guidelines and Requirements.**

Traceability!!!!

n	(example section for component V&V)	Comments	Guidelines/Requirements Link	Risk Metric
N.1	List component	links to component implementation	TAGn0	
N.2	List component V&V	links to component test plan and verification modules	TAGn1	
N.3	Reported coverage	Links coverage reports	TAGn2	
N.4	Assess coverage (are coverage numbers accurate, files verified)			

Example: FPGA Security Features Subsection

PUF: Physical unclonable function

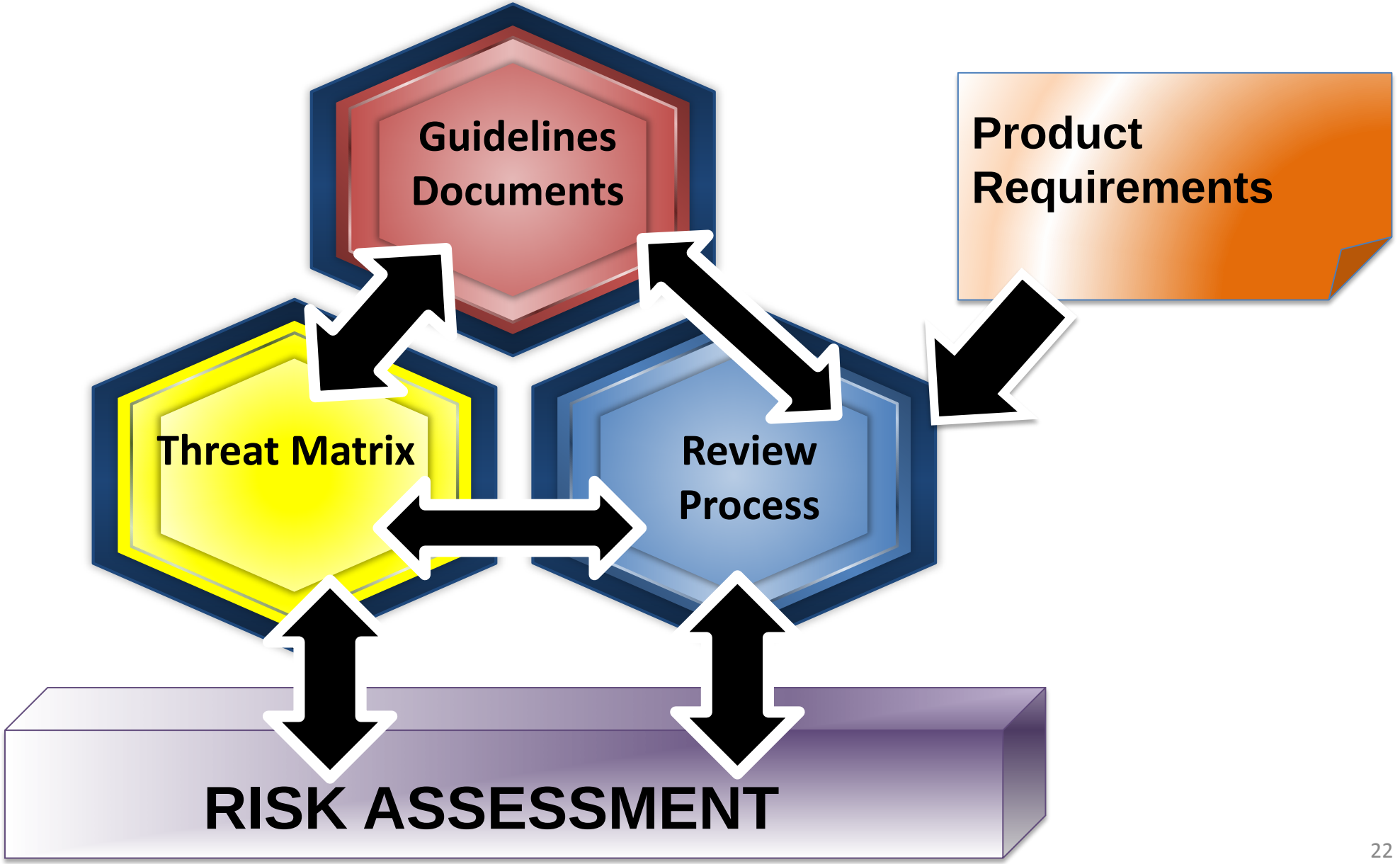
3	FPGA Security Features	Comments	Guidelines/Requirements Link	Risk Metric
3.1	Does FPGA require a Key?	A key is required. Requirement ##.##		
3.2	If a Key is required, what type of Key is being implemented (e.g.: embedded PUF, soft PUF , stored Key, components (memory versus ring oscillator);	links to datasheet: Embedded PUF – ring oscillator.		
3.3	Provide link to Key implementation radiation results (Single event effects, total dose, and prompt dose);	No radiation data is available		
3.4	Assess functional coverage of implementation. Is there potential for lockout due to Key access failure ? Example of failure can be due to radiation effects, adversary learning, or gaps in mitigation.	No tests have been performed to determine lockout threat		
3.5	If no lockout, show proof.			

Link to Requirements Matrix

RISK!

Depending on target environment

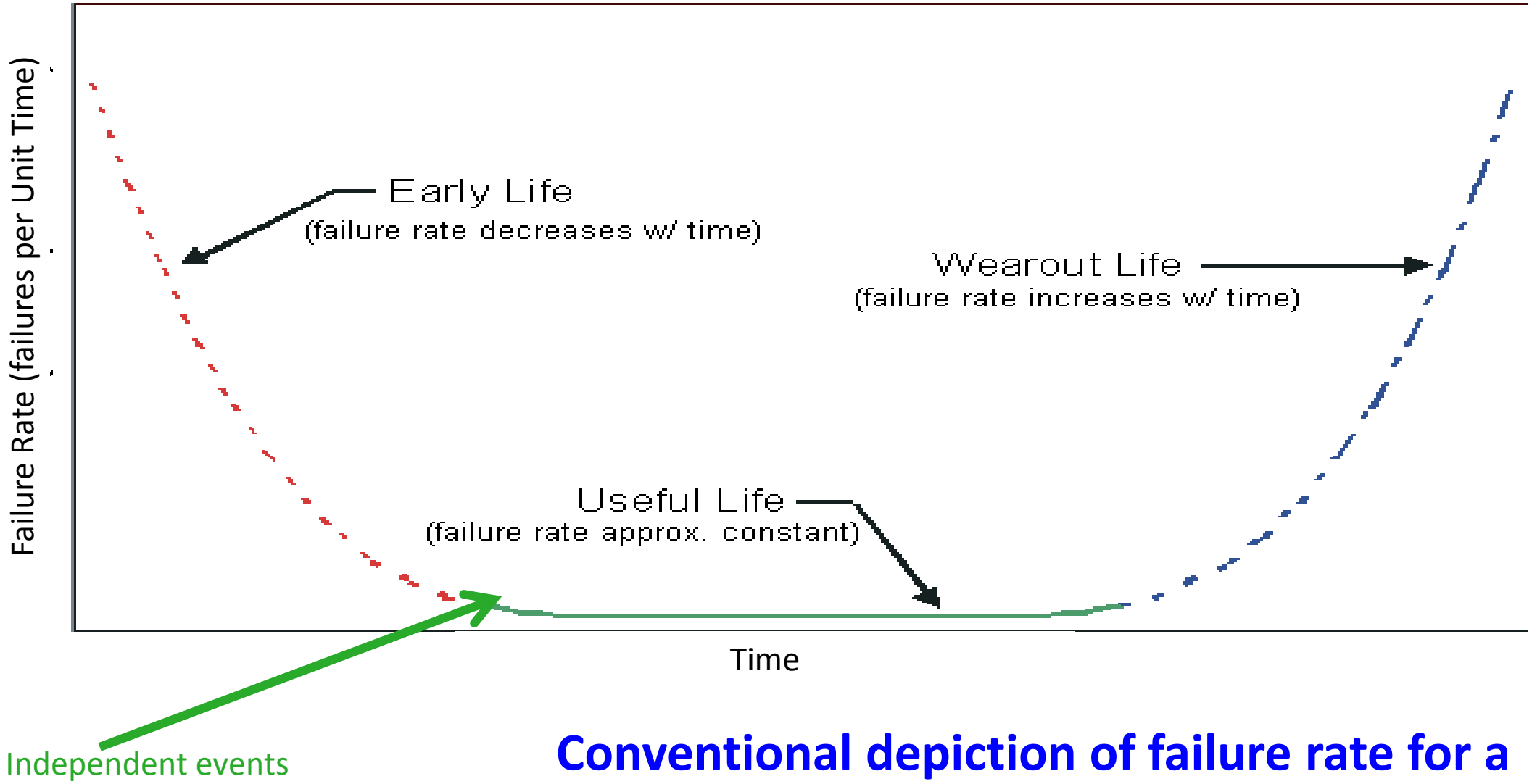
GOMAT:
Risk Assessment



GOAL

To be presented by Melanie Berg at the National Defense Industrial Association Field Programmable Gate Array Assurance Workshop, McLean, VA February 28th 2018

Weibull Failure Rate (λ(T)) Bathtub Curve



Conventional depiction of failure rate for a complex system.

Additional Modes of Failure Are Included in The Failure Rate Analysis When Analyzing Threat Space

- We denote adversary access as mitigation failure.
- Two major concerns:
  - Probability of access and
  - Outcome of access.
- Probability of Access: access is most likely not a constant failure rate (failure is not random – it is forced).
  - There is a learning process with a calculated plan.
  - Failure rate increases over time due to the adversary getting smarter.
  - Some people assume Bayesian is the answer.
- Outcome: Not all mitigation failures are detrimental – e.g.,
  - Multilayer mitigation forces the adversary to learn more.
  - Length of access time could be longer than deployment time,
  - Inability for the adversary to act upon access, or
  - Adversary action is insignificant to system behavior.

But before We Jump The Gun…

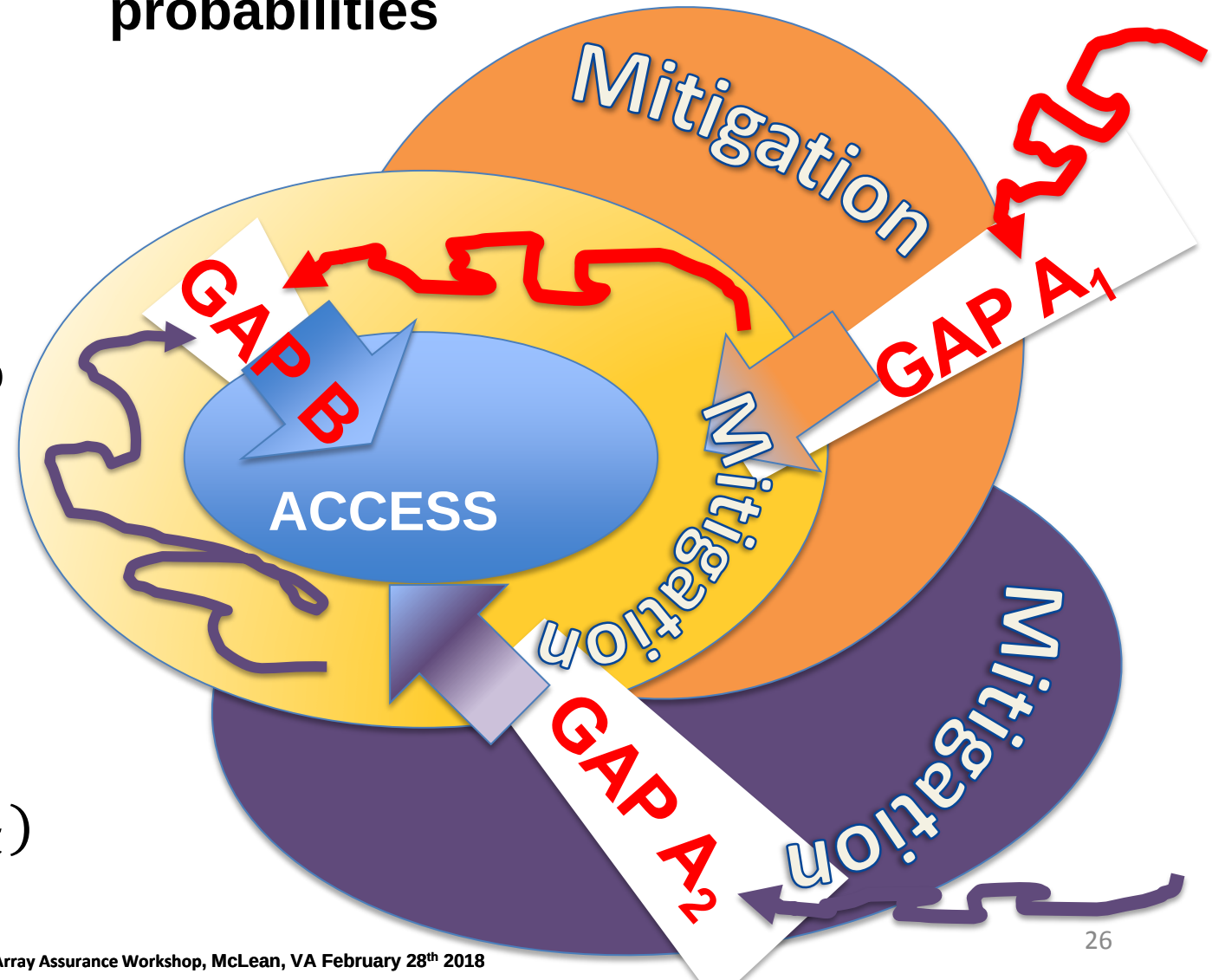
- Sometimes simply identifying critical gaps can be sufficient for risk analysis.
- No need for complex probability analysis.
- However, it is critical to understand the conditional dependencies of potential adversary access.
- Challenge: there are little data on adversary learning:
  - They can either learn how to make a gap in the mitigation or
  - They can learn how to find an existing gap.
- Best to overestimate potential of the adversary. However, be careful, this can cause:
  - Overdesign (cost, area, power),
  - Risk is increased because of complexity – poorly implemented mitigation and potential for lock out.
- Always take into account multiple layers of mitigation.

Multilayered Mitigation: Simple Example

- Every gap must be learned by the adversary.
- Some gaps are blatant and some require extensive learning.
- Multilayered mitigation has conditional probabilities.
- If there is only way to gain access to Gap B – what are all the paths that can reach Gap B?
- Probability of infiltration (GAP B) given Gap A has been infiltrated ($P(B|A_i)$) is based off of learning and increases over time.

$$P(B) = \sum_i P(A_i) * P(B|A_i)$$

Must understand all GAP A_i probabilities



Determining Gap Probabilities: State-space Coverage As It Pertains to Risk

- The goal of coverage for trust is to find all vulnerabilities (that are not physical).
- State-space coverage:
  - Simulation,
  - Emulation,
  - Formal methods,
  - Static methods.
- Cannot simply depend on coverage statistics:
  - 80% of what is covered?
  - What if the most crucial portion is not covered?
- Important to understand EDA tool engines – their benefits and their inadequacies.
- Lack of coverage insinuates potential gaps. Gap risk depends on:
  - Accessibility and Outcome
  - Not just coverage.

Bayesian Methods: Monte Carlo, Game Theory, or Markov Chain Analysis

- When does it make sense?
  - Determining the best mitigation strategy or
  - If a failure occurred, trying to predict the path of entry (access).
  - You're looking backwards.
- Problem –
  - Usually based off of very little information.
  - Assumptions are made.
  - One wrong assumption can drastically throw off the analysis and result in extremely bad conclusions.
- Benefit –
  - Can get a conclusion without data (hopefully it is reliable).
  - Exhaustive assumptions and state analysis can paint a fairly good picture of threat path potential.
  - Software is available!!!!!!!!
- Markov Chain modeling can be used as Bayesian or classical. We will assume classical usage when determining risk factors for the checklist.

FASTIME Strengths

V&V: Verification and Validation

- Differentiation between user flow and assessor flow:
  - Guidelines and requirements are provided to the target team and are used as references for the review process (what should be done).
  - Actual implementation is reviewed.
- Framework takes into account:
  - Observed gaps.
  - Potential gaps (unobtainable information, lack in V&V coverage, not vetted personnel).
  - Multiple layers of mitigation (co-dependencies).
  - Potential for adversary's learning process as it pertains to the actual implementation of mitigation.
  - Full ecosystem (personnel, IT, tools, design process, data handling, etc,…)
- Risk analysis is robust:
  - Includes V&V coverage… **coverage is not the only element that defines risk.**
  - Risk metrics are more than colors or simple strength descriptions.
  - Risk metrics are based on time-to-infiltration and weighted outcome.
  - Risk items can be red-lined for immediate attention.
- Eventual integration with model based system engineering tools.

Vulnerabilities are determined by coverage of guidance, requirements, and implementation discrepancies.