



U.S. ARMY COMBAT CAPABILITIES DEVELOPMENT COMMAND – ARMAMENTS CENTER

Army Software Suitability Statement Criteria for AI

Adam Hilburn

DEVCOM Armaments Center | QE&SA



SOFTWARE SUITABILITY STATEMENT OBJECTIVE



- **Proof is needed to make a release decision**
- **SW Quality engineers collect evidence to increase confidence**
- **Evidence is documented in the Software Supportability Statement [SSS]**



SW DEVELOPMENT LIFECYCLE



Static Testing

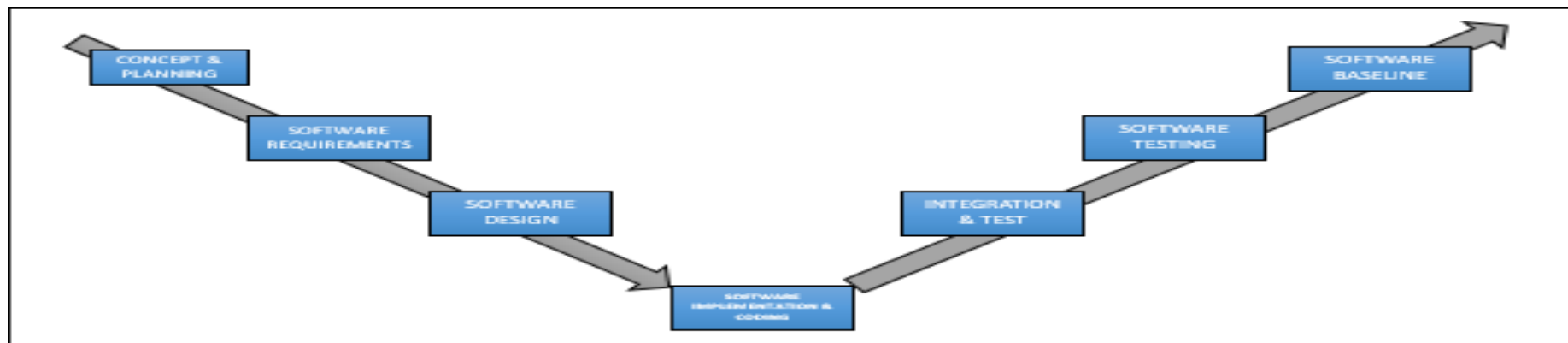
Dynamic Testing

SSS

Risk Analyses

Defect Tracking

CM





STATIC TESTING TO FIND DEFECTS



- **2/3 defects originate during reqts and design**
- **SRS, SDD, Test Procedures**
- **Suitability**
- **Evaluation criteria**
 - Defined input domain [interfaces]
 - Exception handling [handle off nominal situation – GPS loses signal]
 - Risks factored in [enemy corrupts our data]
- **Prepare for Dynamic testing**



No Priority 1 or 2 defects based on a suitable testing strategy



RISK ANALYSES



- **Collaboration**
- **Clear context of intended functionality**
- **Risk: hazard, defects, vulnerabilities**
- **Identify risks using correct mission profile [factor everything in]**
- **Assess risks and define mitigations**
- **Understand level of SW control**
- **Dictates suitable testing strategy**
- **Use risks as evaluation criteria → DfX**



Vulnerabilities Mitigated by SW functionality have been Verified



DYNAMIC TESTING



- **Predefined expected outputs**
- **Suitable testing environment**
- **Cover all risks**
- **Trace to prioritized functionality – verify SW controls**
- **Understand limitations and residual risks**



No Priority 1 or 2 defects based on a suitable testing strategy



DEFECT TRACKING



- **Data Review Boards – manage defects**
- **Root cause analyses process ending in regression**
- **Goal is finding 2/3 defects during development**



No Priority 1 or 2 defects based on a suitable testing strategy



CONFIGURATION MANAGEMENT



- **Formal Configuration Management process**
- **Need to ensure we are testing and releasing the correct version**
- **Witness SW builds**



No Priority 1 or 2 defects based on a suitable testing strategy



SW DEVELOPMENT LIFECYCLE



Static Testing

Dynamic Testing

SSS

Risk Analyses

Defect Tracking

CM





AI LIFECYCLE



Select
Algorithm

Suitable
Data
[static
testing]

Train Model

Dynamic Testing
SSS

Risk Analyses

CM

Defect Tracking



SELECT ALGORITHM



- Was Quality factored in?
- SQE involved with selection – opportunity to understand inner workings and parameters – prepare for testing
- Is it suitable for intended military environment?
- Is there a confidence level for each decision?
- Can it be taught to recognize hacks?
- Suitable to handle expected types of data?





SUITABLE DATA – STATIC TESTING



- **Certify the data**
- **Unbiased**
- **Clear categories**
- **Operational environment**
- **Expected results**
- **Explainable input domain – boundaries understood – analyses done with SQE oversight**
- **Coverage of input domain**



No Priority 1 or 2 defects based on a suitable testing strategy



RISK ANALYSES



- **Collaboration**
- **Risk: hazard, defects, vulnerabilities**
- **Identify risks using correct mission profile [factor everything in]**
- **Assess risks and define mitigations**
- **Understand level of SW control**
- **Use risks as evaluation criteria → DfX**



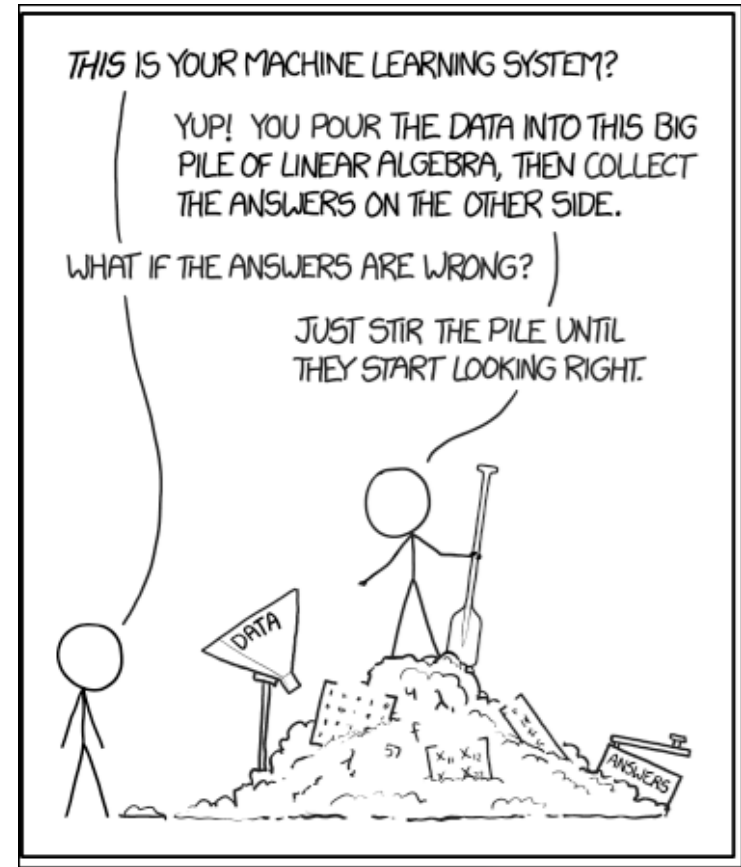
Vulnerabilities Mitigated by SW functionality have been Verified



TRAIN MODEL



- Process to prove suitable learning
- Define criteria for incorrect responses
- Prove suitable level of repeatability
- Expected results pre-defined
- Withheld data set



No Priority 1 or 2 defects based on a suitable testing strategy



DYNAMIC TESTING



- **AI solution integrated with traditional SW solutions**
- **Suitable testing strategy**
- **Predefined expected outputs Suitable testing environment**
- **Cover all risks**
- **Cover all functionality**



No Priority 1 or 2 defects based on a suitable testing strategy



DEFECT TRACKING



- **Learning Data Review Boards – manage failures related to learning**
- **Failure = incorrect response**
- **Root cause analyses process ending in regression**
- **Goal is finding 2/3 defects during development**



No Priority 1 or 2 defects based on a suitable testing strategy



CONFIGURATION MANAGEMENT



- Formal CM process for a system that is continuously learning
- Need to ensure we are testing and releasing the correct version
- Witness process of locking a system from learning before dynamic test



No Priority 1 or 2 defects based on a suitable testing strategy



AI LIFECYCLE



Select
Algorithm

Suitable
Data
[static
testing]

Train Model

Dynamic Testing
SSS

Risk Analyses

CM

Defect Tracking



CHALLENGES



- We are viewed as the bad guys
- Delicately tell someone their baby is ugly
- Complexity | Can't eliminate all defects
- Schedule driven
- Lacking documentation to review - S&T and Agile



6 stages of debugging

1. That can't happen
2. That doesn't happen on my machine
3. That shouldn't happen
4. Why does that happen?
5. Oh, I see
6. How did that ever work?



PATH FORWARD



- **Continuously improve through collaboration**
- **Define suitable processes [learning reqts – eval criteria (testable)]**
- **Data certification process**
- **Learning DRB and CM process [define what an ideal/suitable process looks like]**
- **Drive down to measures**



COLLABORATE WITH US



Adam Hilburn MBB, CSQE, CTEL-TM
Acting SQRSE Competency Dean
AFC | DEVCOM AC @ Picatinny Arsenal | QE&SA
973-724-4877
adam.r.hilburn.civ@army.mil

Thank you.