18 January 2019

Ministry of Defence

Requirements Assurance in Machine Learning (ML) Applications

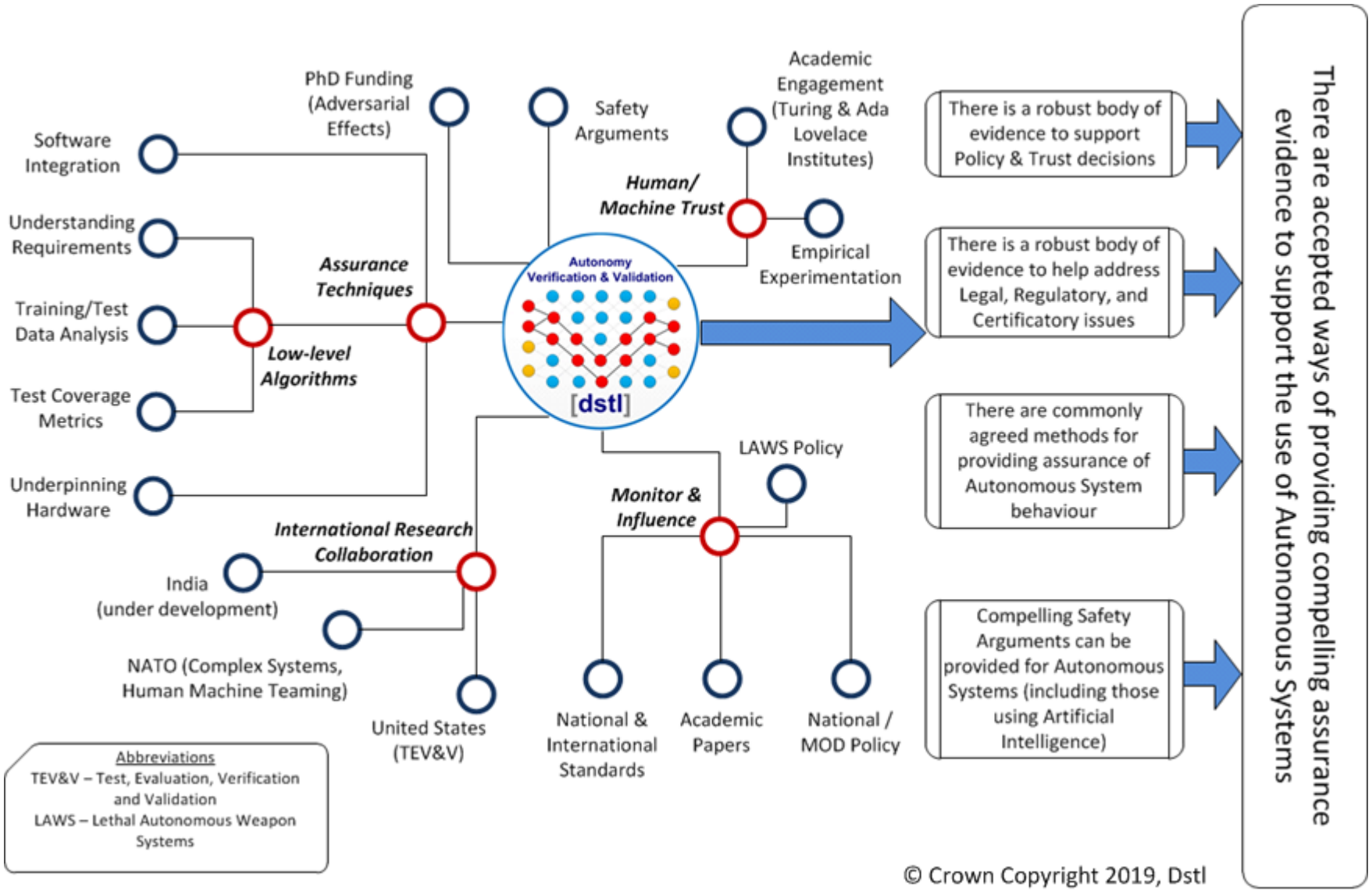
Dr Alec Banks and Rob Ashmore

SafeAI 2019

DSTL/PUB113099

Disclaimer

The contents of this presentation should not be interpreted as representing the views of the MOD, nor should it be assumed that they reflect any current or future MOD policy. The information contained in this presentation cannot supersede any statutory or contractual requirements or liabilities and is offered without prejudice or commitment.



Starting Out, The"4+1" Principles[1]

P1: Software safety requirements shall be defined to address the software contribution to system hazards.	**Define system-level requirements**
P2: The intent of the software safety requirements shall be maintained throughout requirements decomposition.	**Refine into something you can code against**
P3: Software safety requirements shall be satisfied.	**Code what you intended - verification**
P4: Hazardous behaviour of the software shall be identified and mitigated.	**Look for new system-level hazards**
P4+1: The confidence established in addressing the software safety principles shall be commensurate to the contribution of the software to system risk.	**Target resources at highest risks**

P1 - Principle 1 (and so on)

[1] Hawkins, R, Habli, I, Kelly, T., 2013. The Principles of Software Safety Assurance.

ML Challenges the "4+1" Principles[1], [2]

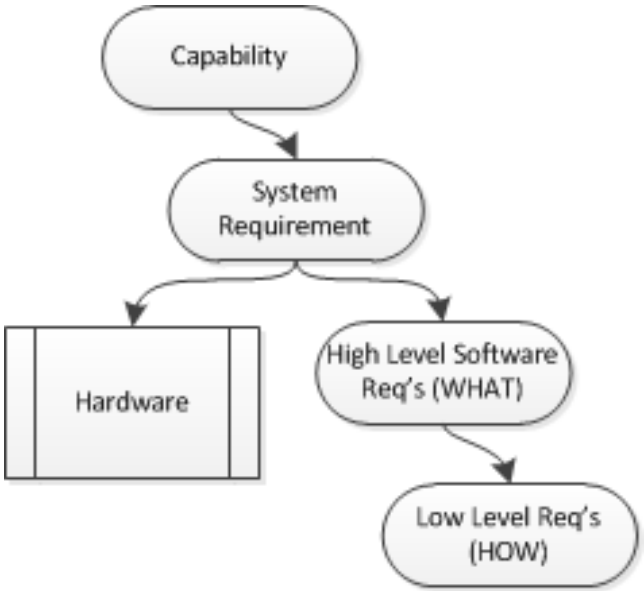
P1: Software safety requirements shall be defined to address the software contribution to system hazards.	Define system-level requirements	OK
P2: The intent of the software safety requirements shall be maintained throughout requirements decomposition.	Refine into something you can code against	?
P3: Software safety requirements shall be satisfied.	Code what you intended - verification	?
P4: Hazardous behaviour of the software shall be identified and mitigated.	Look for new system-level hazards	OK (ish)
P4+1: The confidence established in addressing the software safety principles shall be commensurate to the contribution of the software to system risk.	Target resources at highest risks	OK (ish)

P1 - Principle 1 (and so on)

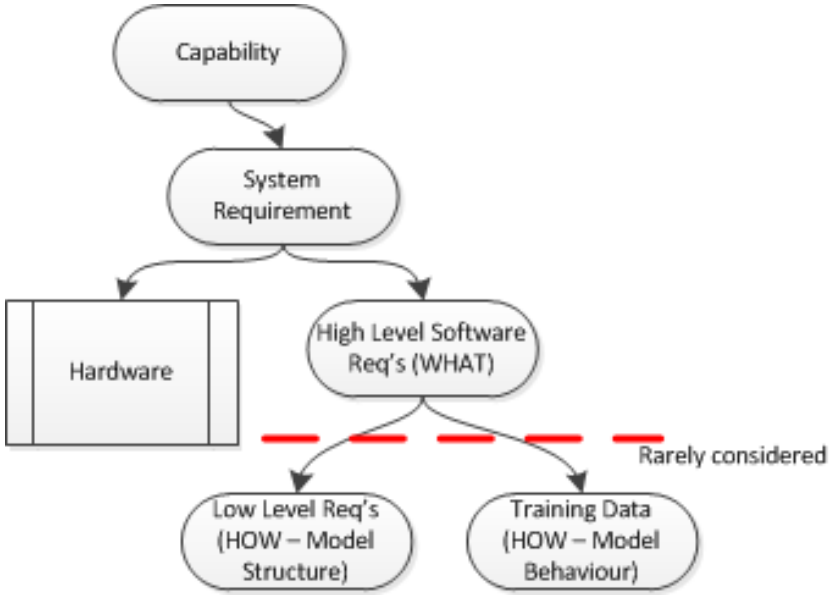
[1] Hawkins, R, Habli, I, Kelly, T., 2013. The Principles of Software Safety Assurance.
[2] Ashmore, R, Lennon, E., 2017. Progress Towards the Assurance of Non-Traditional Software.

Requirements Refinement

- Traditional Systems



- ML Systems



Existing Considerations for Requirements Assurance

Using RTCA DO-178C[3] as an example, requirements should be:

- R1. Compliant with High Level Requirements
- R2. Accurate and consistent
- R3. Compatible with target computer
- R4. Verifiable
- R5. Conforming to standards
- R6. Traceable
- R7. Algorithmically correct

THESE WILL STILL APPLY! For example to the training algorithm

[3] RTCA, 2011. Software Considerations in Airborne Systems and Equipment Certification. DO-178C.

Considerations for the Assurance of ML Training Data

Training Data abstractly forms a significant component of Low Level Requirements. We propose that it should:

- D1. Relate to the intent of the HLR (R2 and R7)
- D2. Not contain bias (R7)
- D3. Be sufficient (R1)
- D4. Be syntactically and semantically correct (R2 and R7)
- D5. Address normal and robustness behaviours (R1)
- D6. Be self-consistent (R2)
- D7. Conform to Standards (R5)
- D8. Be compatible with the target computer (R3)
- D9. Be verifiable (R4)

The Indicative Example

- Each of the training data assurance considerations are ‘coloured’ using an unmanned air vehicle landing system as an indicative fictional example.
- We believe that the approach is domain agnostic.

- Finally, the workshop may be interested to note that the Safety of Autonomous Systems Working Group (SASWG) are publishing algorithmic-level framework guidance for autonomous system safety[4].

[4] Safety of Autonomous Systems Working Group, 2019. Autonomous Systems — Algorithm-Level Objectives.

References

[1] Hawkins, R, Habli, I, Kelly, T., 2013. The Principles of Software Safety Assurance. 31st International System Safety Conference, Boston, Massachusetts USA, 2013.

[2] Ashmore, R, Lennon, E., 2017. Progress Towards the Assurance of Non-Traditional Software. In Developments in System Safety Engineering, ISBN 978-1540796288.

[3] RTCA, 2011. Software Considerations in Airborne Systems and Equipment Certification. DO-178C.

[4] Safety of Autonomous Systems Working Group, 2019. Autonomous Systems — Algorithm-Level Objectives. Available from: https://www.amazon.co.uk/Safety-Assurance-Objectives-Autonomous-Systems/dp/1790421225 (or free PDF, from Feb 2019, https://scsc.uk/publish.html)

18 January 2019

Ministry of Defence