FROM THE GROUND UP: BUILDING SAFETY-CRITICAL SOFTWARE FACTORIES

A MULTI-MODEL APPROACH TO SOFTWARE PRODUCTIZATION

KELLY GASPERSKI, P.ENG. | NISHI HOSSAIN

SOFTWARE PRODUCT ASSURANCE CONFERENCE 2025 | ESA ESTEC

COMMERCIALIZATION OF SPACE

- The space industry has entered a new revolution.
- Technology has become sufficiently *advanced*, *available*, and *affordable* to pursue new projects and businesses.

Space is profitable!

- Satellite communications
- Earth observation
- Manufacturing & robotics
- On-orbit service and maintenance

- Space tourism
- Debris recycling
- Asteroid mining
- Research and technology testbeds

KEY STAKEHOLDERS

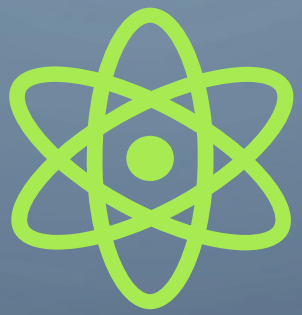
1. Agencies and space tech companies
2. Companies in other industries looking to expand
3. New space startups

Competing to fulfill an expanding market's needs based on Reputation, Profitability, Scalability

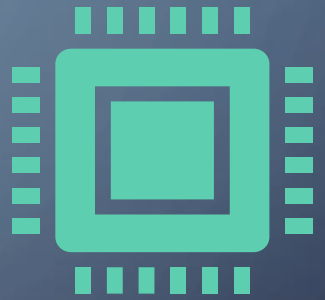
QUALITY is imperative to success.

THE CHALLENGE OF QUALITY IN SAFETY-CRITICAL SYSTEMS

Quality must be built-in from the start;



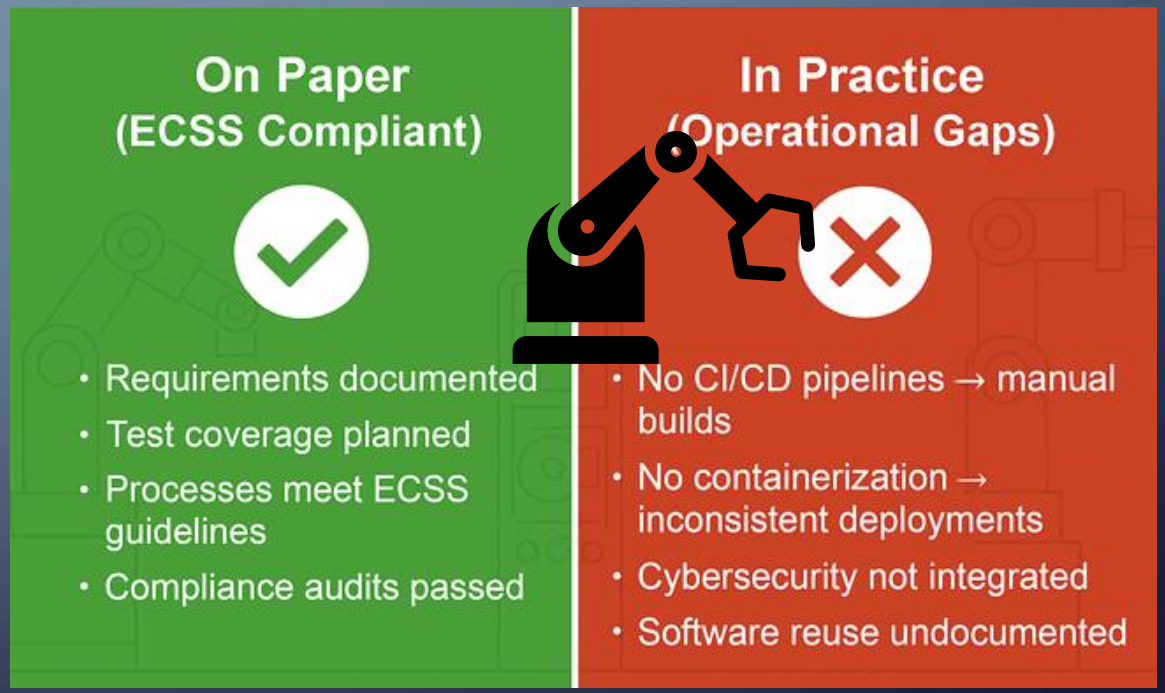
Safety-critical systems have zero tolerance for failure;



Standard quality models have gaps due to lagging technological advances.

CASE STUDY: SPACE ROBOTIC ARM
PITFALLS OF A SINGLE QUALITY MODEL

- Following a single quality model (ex. ECSS, ISO, IEEE) is not enough.
- Process gaps lead to:
  - integration issues
  - compliance overhead
  - limited reusability
  - increased risk



CASE STUDY: SPACE ROBOTIC ARM
ECSS QUALITY MODEL GAPS

CI/CD Pipelines

ECSS does not address automated build/test/deploy pipelines — a modern necessity for reproducibility and productization.

Containerization & Cloud Deployment

ECSS is silent on Docker & Kubernetes practices, which are now central to modular product delivery.

Tool Qualification

ECSS assumes static tools, not dynamic pipelines and SaaS-based platforms.

Software Reuse Qualification

ECSS lacks detailed guidance on qualifying reused open-source or AI-generated code.

Cybersecurity

ECSS does not address resilience against cyber threats. Missing secure coding practices, vulnerability scanning, software supply chain assurance, and runtime intrusion monitoring.

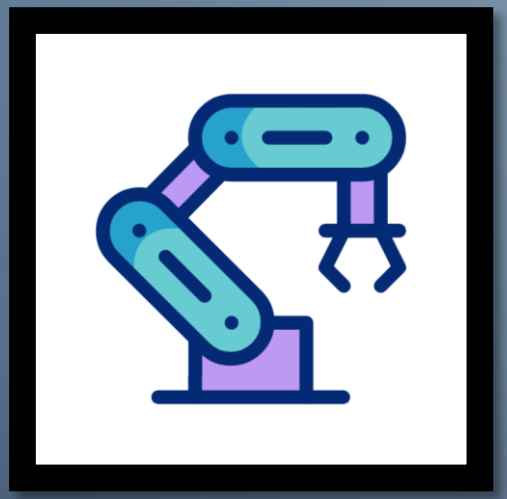
SOLUTION: MULTI-MODEL APPROACH

Standard	Strengths	Gaps/Limitations	Pairings
ECSS (Space)	Safety-critical rigor, requirements traceability, verification & validation, config management.	Weak in CI/CD, containerization, modern toolchains, reuse of OSS/AI, cybersecurity.	ISO 12207 for lifecycle structure, IEEE 29119 for testing, ISO 27001 for security, CMMI for maturity
ISO/IEC 12207	Broad software lifecycle processes, requirements → maintenance coverage, adaptable to domains.	Not prescriptive on DevOps, testing depth, security, tool qualification.	ECSS for rigor, IEEE 29119 for test discipline, CMMI for process maturity, ISO 27001 for security.
IEEE 29119	Strong test lifecycle definition (planning, design, execution, reporting).	Focuses only on testing, doesn't cover: design, deployment, or security.	ECSS/ISO for requirements & design, CMMI for process integration, ISO 27001 for cyber coverage.

SOLUTION: MULTI-MODEL APPROACH

Standard	Strengths	Gaps/Limitations	Pairings
CMMI	Process maturity, continuous improvement, scaling capability, quantitative management.	Not domain-specific, doesn't define safety-critical compliance or cyber requirements.	ECSS/ISO for compliance, IEEE for testing, ISO 27001 for security.
ISO/IEC 25010	Defines quality attributes (portability, reliability, maintainability, usability).	Doesn't define lifecycle processes or compliance frameworks.	Complements ECSS/ISO lifecycle, adds quality checks to CI/CD & containerization.
ISO/IEC 27001 or NIST SP 800 series	Cybersecurity management, risk controls, supply chain assurance, vulnerability management.	Not focused on functional safety or lifecycle assurance.	ECSS/ISO/IEEE for lifecycle rigor, CMMI for embedding cyber into continuous improvement.
DO-178C / DO-330 (Avionics, optional for space)	Safety-critical software assurance, tool qualification (DO-330).	Doesn't cover full lifecycle agility or productization.	Combined with ISO/ECSS for lifecycle, ISO 27001 for security, CMMI for scaling.

CASE STUDY: SPACE ROBOTIC ARM
MULTI-MODEL IN ACTION



REQUIREMENTS:

- Motor control
- Sensor fusion
- Safety interlocks
- Communication
- Fault detection

SOLUTION:

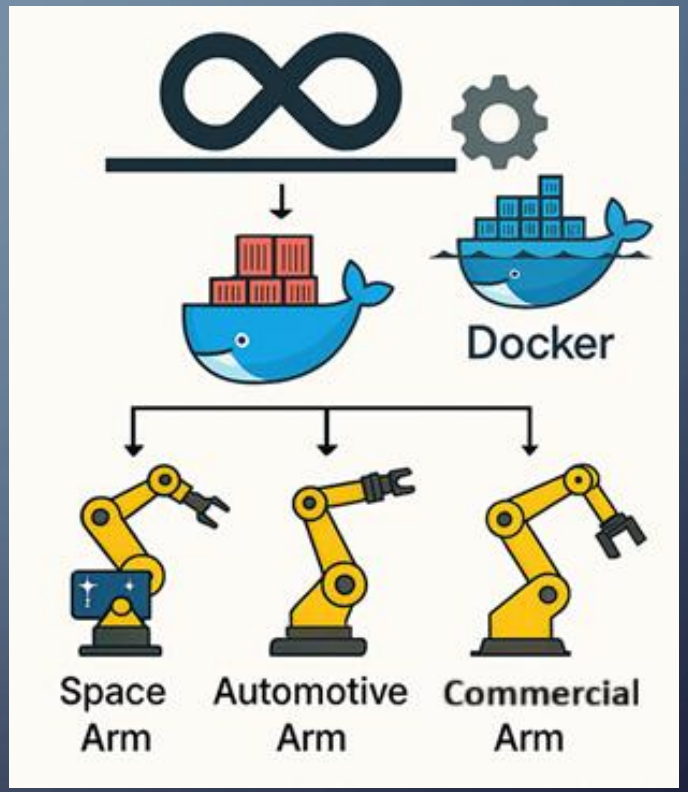
- ECSS → Safety-critical rigor
- ISO/IEC 12907 → Lifecycle coverage
- IEEE 29119 → Testing discipline
- CMMI → Process maturity
- ISO 27001 → Cybersecurity

SOFTWARE PRODUCTIZATION

- Commercialization demands the ability to generate high-quality, compliant software in a ***repeatable*** and ***scalable*** fashion.

Software factories are the solution.

- CI/CD Pipelines → automate processes, improving both quality and efficiency.
- Containerization → ensures reproducibility, isolation & portability, and scalability.



CASE STUDY: SPACE ROBOTIC ARM
MULTI-MODEL FACTORIES IN ACTION

REQUIREMENTS:
- Repeatable
- Scalable
- Modular (reusable components)
- Maintainable

SOLUTION:
- CI/CD Pipelines
  → ISO 12207 + IEEE 29119 + CMMI + NIST SSDF
- Containers
  → ISO 25010 + ISO 27001 + NIST 800-190

BUILDING SAFETY-CRITICAL SOFTWARE FACTORIES

1. Use a multi-model approach to defining quality requirements.
2. Establish your qualified software development toolchain.

   *REMINDER: tools must be qualified for use **prior** to beginning development!*

   - Source Code Repository & IDEs
   - CI/CD tools (orchestrator, build and test frameworks, static analysis, security scans, documentation)
   - Signed Image Registry for trusted, version-controlled software artifacts
   - Domain Overlays - Adapt for space, automotive, medical, consumer needs

3. Deploy → Deliver safe, repeatable, auditable software.

CONTINUOUS IMPROVEMENT

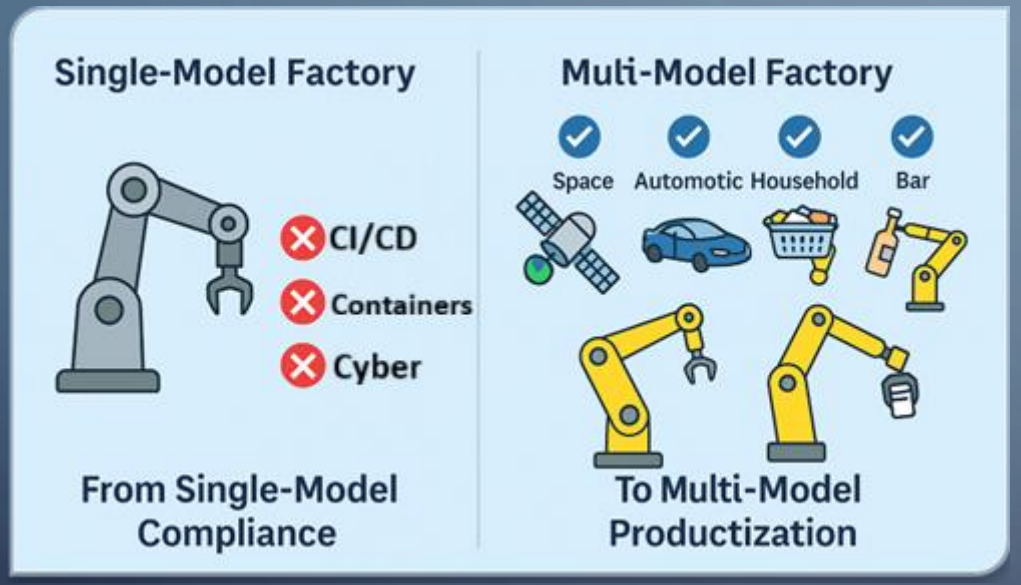
Pipeline containerization

- Supports multiple software factories in parallel
- Enables companies to meet differing project needs, quality models, timelines

Process Maturity Models

- CMMI Levels measure repeatability, predictability, optimization
- IEEE 29119 ensures standardized testing maturity
- Enables reduced variability in deliverables, predictable compliance audits
- Integrates lessons learned to strengthen the foundation

KEY TAKEAWAYS



- Single quality model approach may achieve compliance, but it leaves critical gaps.
- Multi-model approach enables end-to-end assurance supporting safety, agility, and security.
  - CI/CD + Containerization = reproducibility & scalability.
  - Tool & reuse qualification = trustworthy automation.
  - Cybersecurity = resilient systems.

Multi-model software factories are not just compliant — they are built for modern productization.

THANK YOU!

Kelly Gasperski, P.Eng.

Software Engineer | Safety & Quality Assurance Specialist | Analog Astronaut

Nishi Hossain

Software PA at MDA Space