Andrew W. **Lyjak**

Software Assurance/Verification/Development · Systems Engineer

6205 N Oakland Ave, Indianapolis IN, 46220

(+1) (734)604-6163 | andrew.lyjak@gmail.com | spacelyjak.rtfd.io | alyjak | andrew-lyjak | @spacelyjak

"This is the real secret of life – to be completely engaged with what you are doing in the here and now. And instead of calling it work, realize it is play."

Education

University of Michigan — *Ann Arbor, Michigan*

M.E. in Space Systems Engineering — *2009 - 2010*

GPA: 3.769/4.000

University of Michigan — *Ann Arbor, Michigan*

B.E. Aerospace Engineering — *2005 - 2009*

- Minor in German
- GPA: 3.456/4.000

Skills

Programming	Python, LaTeX, ReStructuredText, Bash, Linux, R, Rust, C, C++
Web	Django with Python, Javascript, HTML, CSS, SQL
Systems Engineering	Regulation and Certification, Verification Planning, Fault Tolerance Analysis, Risk and Change Management
Software Quality	git, Trac, Subversion, JIRA, Software Development Planning, Quality Audits
Languages	English, German

Experience

Software Mission Assurance Engineer — *Hawthorne, CA*

SpaceX — *2010 - Present*

Software Certification for NASA Safety and Quality Requirements — *2010 - Present*

- Wrote the Flight Software Development Plan. This document defines SpaceX's custom agile development process for designing, developing, and verifying flight software.
- Designed, wrote, and maintain the Cargo Dragon Flight Software "Computer Based Control Systems" documentation. This material is used to demonstrate compliance with NASA software safety requirements and also defines many of the software safety test and analysis activities executed for verification of software safety related to the Dragon cargo vehicle.
- Perform process audits to verify software quality, and compliance with NASA requirements for the Dragon cargo vehicle.
- Maintain audits, and safety documents through all changes to system design and software for each new commercial cargo resupply mission.

Independent Verification and Validation (IVV) Contract Management — *2010 - Present*

- Serve as the technical point of contact for SpaceX's IVV Contracts for safety critical software.
- Provide the contractor with sufficient data to effectively evaluate the safety of the flight software. Contractor evaluates system design, source code, and verification results.
- Ensure the contractor's feedback is internally addressed and incorporated to create a better product for each commercial crew and cargo mission.
- Manage contracts associated with IVV of flight software for Crew Dragon, Cargo Dragon, and the Autonomous Flight Termination System.

Software Process Tool Development — *2012 - Present*

- **branchdiff** - Developed an application to view the differences between two Subversion branches to facilitate merge decisions between them. Displays differences as commits or as the set tickets referenced within those commit messages.
- Developed an application for viewing change over time for various Trac ticket queries.
- Performed trade studies on various software development ticketing systems. The study factored into SpaceX's decision to adopt JIRA across multiple business domains.
- **ReadTheManual** - Installed, modified, and adminstered an internal fork of ReadTheDocs for use within the SpaceX intranet. This service is used to build and distribute documentation for over 160 internal projects.
- **Tracegraph** - Designed and developed a library for defining systems relationships across information housed within multiple data silos. The library is used for verification tracking purposes to support compliance tracking of SpaceX processes to customer requirements.

Design and Development of the SpaceX Software Standard

2013 - Present

- Developed a software engineering standard for SpaceX. The standard provides a set of requirements to be applied to different classifications of software development, classifications include A, for safety and mission critical software, through D, used for desktop, R&D, or other non-critical applications. The standard is used to evaluate the quality of all softare processes related to the Commercial Crew system.

Fault Tolerance Analysis

2016 - Present

- Co-developed the **Fault Tolerance Analyis Process**. This process is used to evaluate autonomous and/or operator controlled electromechanical systems. Analysis is used to assess the fault tolerance of a design in order to determine its capabilities and develop fault detection, isolation, and recovery logic for managing redundant capabilities.
- Developed test and analysis plans for verifying fault tolerance as defined through the Fault Tolerance Analysis products.
- With a team of 6 other engineers, executed the Fault Tolerance Analysis Process against 25 separate autonomous control systems within the Crew Dragon Architecture over the course of a year.