

ANDREW LYJAK

6205 N Oakland Ave, Indianapolis IN, 46220

+1 734 604 6163 ♦ andrew.lyjak@gmail.com

EDUCATION

ME Space Systems Engineering

2009 - 2010

University of Michigan, GPA: 7.538/8.000

BE Aerospace Engineering

2005 - 2009

Minor in German

University of Michigan, GPA: 3.456/4.000

INTERESTS

Design and Development Processes, Distributed Systems, Resilient Systems, Autonomy, Automation, Software Certification, Verification of Complex Systems, Model Driven Development, Continuous Integration, Agile Development

SKILLS AND TOOLS

Software Development Process Design, Software Process Audits, System Certification, Verification Planning, Fault Tolerance Analysis, Python, Javascript, HTML, CSS, Bash, Linux, R, C, C++, SQL, L^AT_EX, ReStructuredText, Graph Analysis, git, Trac, Subversion, JIRA, Confluence

EXPERIENCE

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 SpaceX flight software.
- Designed and wrote the Cargo Dragon Flight Software "Computer Based Control System" certification documentation. This material was used to demonstrate SpaceX compliance with NASA software safety requirements and also defined many of the software safety test and analysis activities executed for verification of software safety.
- Perform process audits to verify software quality, and compliance with NASA requirements.
- Maintain audits, and safety documents through all changes to system design and software.

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.
- manage contracts associated with IVV of flight software for Crew Dragon, Cargo Dragon, and SpaceX's Autonomous Flight Termination System.

Software Process Tool Development

2013 - 2014

- **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 – Trac, JIRA, Phabricator, Redmine. The study factored into SpaceX's decision to adopt JIRA across multiple business domains.
- **ReadTheManual** - Installed, Modified, and Administered an internal fork of ReadTheDocs for use within the SpaceX Intranet. Used to distribute documentation for over 160 internal projects.

- **Tracegraph** - Designed and Developed a protocol and library for defining systems relationships for verification tracking purposes. Library is used for compliance tracking of spacex processes to customer requirements.

Design and Development of the SpaceX Software Standard

2013 - 2015

- Developed a software engineering standard for SpaceX. The standard provides a set of requirements that can be used for 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 requirements are categorized to account for different aspects of software design and development, ranging from change management, risk management, verification processes, and technical and analytical requirements for safety critical software. The standard is used to evaluate all software processes related to the NASA Commercial Crew contract.

Design, Development, and Execution of the SpaceX Fault Tolerance Analysis Process

2016 -

Present

- Co-developed the **Fault Tolerance Analysis Process**. This process is used to evaluate autonomous and/or operator controlled electromechanical systems. These analyses systematically assess a system's fault tolerance in order to determine the design's robustness as well as develop fault detection, isolation, and recovery algorithms for managing a fault tolerant system's 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 SpaceX Engineers, executed the Fault Tolerance Analysis Process against 25 separate autonomous control systems present within the Crew Dragon Architecture over the course of a year.