UL 4600
Technical Overview

October 10, 2019
Deborah Prince, Underwriters Laboratories
Dr. Philip Koopman, Edge Case Research
Webinar Goals

UL 4600: Standard for Safety for the Evaluation of Autonomous Products

- Overview for technical stakeholders
  - Comments due Friday November 1

- Goals for this Webinar
  - Orientation to standard for technical audience
  - Key principles to keep in mind when commenting
  - How to get a copy and submit comments
  - Q&A
Why UL?

Underwriters Laboratories: working for a Safer World for 125 years
- Published first safety standard in 1903
- Focus on research, education, and more than 1,700 standards

UL’s Standards Development process
- Consensus process
- Open, transparent, and timely
- Continuous standards maintenance
UL 4600 Standards Technical Panel (STP)

**STP is the voting consensus body**

<table>
<thead>
<tr>
<th>Company</th>
<th>Institute/Company Name</th>
<th>Company</th>
<th>University/Institution</th>
<th>Organization</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANSYS</td>
<td>Beijing Research Institute of Automation for Machinery Industry</td>
<td>Intel Corp</td>
<td>Nanyang Technological University</td>
<td>Robert Bosch LLC</td>
</tr>
<tr>
<td>Argo AI</td>
<td>Center for Auto Safety</td>
<td>Intertek</td>
<td>NIO</td>
<td>UBER ATG</td>
</tr>
<tr>
<td>Aurora Innovations</td>
<td>Consumer Product Safety Commission</td>
<td>Liberty Mutual Insurance Company</td>
<td>Nissan North America Inc</td>
<td>UL LLC</td>
</tr>
<tr>
<td>AXA XL</td>
<td>Daimler Trucks North America</td>
<td>Locomation</td>
<td>Oak Ridge National Laboratory</td>
<td>University of York</td>
</tr>
<tr>
<td>Azevtec Inc</td>
<td>Edge Case Research</td>
<td>The MITRE Corp</td>
<td>Penn DoT</td>
<td>University of Waterloo</td>
</tr>
<tr>
<td>Babst, Calland, Clements &amp; Zomnir</td>
<td>Infineon Technologies AG</td>
<td>Munich Re America</td>
<td>Renesas Electronics Europe GBMH</td>
<td>US DoT</td>
</tr>
</tbody>
</table>
Timeline

- **Initial drafting**
  - July 2018: Announced intent to develop UL 4600

- **STP revisions**
  - June 2019: STP meeting to discuss first full draft
  - Three rounds of STP comment & draft revisions completed

- **Stakeholder comments**
  - Oct 2019: Stakeholder preliminary draft available
  - Stakeholder comments due Nov 1, 2019

- **Target final version release Q1 2020**
Technical Overview

- Orientation to current preview draft version
  - Contents and organization subject to change!

- UL 4600 Scope
  - Fully Autonomous Vehicle (AV) operation
  - No human driver/supervisor

- Main principles
  - Safety case is front and center

- Guide to review & comments
Goal: structured way to argue that AV sufficiently safe
- Non-prescriptive, safety case approach
- Trace all safety goals (claims) to evidence
- Checks and balances (self-audit and independent)

Monitoring and feedback
- Detect invalid assumptions & gaps in coverage

System Level + Life Cycle approach
- Includes fault recovery, supply chain issues, expected misuse

Reference lists to improve completeness
- Prompts & epistemic defeaters for coverage (#DidYouThinkofThat?)
- Ability to argue that some prompts aren’t applicable
Why UL 4600?

- Autonomous systems have unique needs
  - No human supervision, non-determinism, ...
  - This version: highly automated vehicles

- System level approach needed
  - Functional safety, SOTIF, road tests, simulation all play a role
    - But need a framework to put the pieces together
  - Adapt as technology evolves

- Cooperate rather than compete
  - Can accept work products from ISO 26262, ISO/PAS 21448, etc.

- Goal: guidance on “Is system engineering rigor sufficient?”
Traditional safety standards are prescriptive

- “Here is how to do safety” (process, work products)
  - ISO 26262, ISO/PAS 21448, IEC 61508, MIL-STD 882, etc.
- But, we’re still figuring out some aspects of AV safety

UL 4600 is goal based: “be acceptably safe”

- Use a Safety Case to argue system is acceptably safe
  - Define what safe means; argue that AV meets that definition
  - Do **NOT** prescribe any particular engineering approach
  - **DO** require a set of minimum acceptable topics for safety case
- Require use of any good system engineering process (not just V)
What’s A Safety Case?

- A structured argument backed by evidence
  - Notation agnostic / use any reasonable notation
- SubGoal/Claim: “AV will not hit pedestrians”
  - Hypothetical Arguments
    - “AV will detect pedestrians of all types”
    - “AV will stop or avoid collision detected pedestrians”
    - “We have identified & mitigated risks caused by difficult to detect pedestrians”
  - Hypothetical Evidence
    - “Here are results of detect & avoid tests”
    - “Here is analysis of coverage of different types of pedestrians”
    - “Reliability growth data shows high pedestrian coverage”
UL 4600 Scope

System level safety for autonomous operation & lifecycle

TOP LEVEL GOAL: AV SAFETY CASE IS ACCEPTABLE (Hypothetical/Simplified)

- CONTEXT DEFINED
- HAZARDS IDENTIFIED
- RISKS MITIGATED

SYSTEM (Item scope: Vehicle + Infrastructure)
- ODD SPECIFIED
- PROMPT ELEMENTS TAILORED TO ODD & SYSTEM
- RIGOROUS DEVELOPMENT PROCESSES
- RIGOROUS OPERATIONAL PROCESSES
- SAFETY CULTURE

ADDRESS PRIORITY OF PROMPT ELEMENTS
- TRACEABILITY WITHIN SAFETY CASE & TO UL4600
- REASONABLE INDUCTIVE STEPS / AVOIDS PITFALLS
- METRICS MONITOR SAFETY CASE VALIDITY
- SELF-AUDITS
- INDEPENDENT ASSESSMENT

FAULT MODELS DEFINED
- VEHICLE (SYSTEM & SOFTWARE)
- AUTONOMY PIPELINE
- DATA, NETWORKING, SERVICES
- ROAD USERS
- LIFE CYCLE & SUPPLY CHAIN
- MAINTENANCE & INSPECTIONS
- TOOLS & COMPONENTS

HAZARDS MAPPED TO RISK-BASED INTEGRITY
- FAULT RESPONSE & ODD VIOLATION STRATEGY
- MITIGATIONS IDENTIFIED & SUFFICIENT
- DEPENDABILITY ISSUES ADDRESSED
- FEEDBACK TO MANAGE UNKNOWNS

© 2019 Philip Koopman
Out of Scope for UL 4600

- Related topics
  - ADAS features
  - AV testing safety (but, see BSI/PAS 1881)
  - Ethical guidelines (but, see IEEE P7009)

- Human factors
  - Human attention (as driver; as safety supervisor)
  - How to argue humans will behave as required
  - How to argue human safety supervisor will react correctly

- Details of security
  - Requires security plan; maps security plan to safety
  - Does not attempt to define what is in security plan
Prompt Elements: #DidYouThinkofThat?

Extensive lists of safety case topics, hazards, etc.

- Good practices & Pitfalls (lessons learned & bad practices to avoid)

Prompts must be considered, not necessarily adopted

- **Mandatory**: you have to do this
- **Required**: can deviate **ONLY** if inherently inapplicable
  - E.g., if no machine learning, then can deviate from ML requirements
- **Highly Recommended**: can deviate with non-trivial rationale
- **Recommended**: entirely optional
- **Examples**: illustrative reminders; do not have to address each one

Many processes and technique areas are lightly constrained

- E.g., Identify hazards, but use any reasonable technique
Operational Design Domain (ODD)

- Define relevant ODD considering:
  - Infrastructure
  - Weather & road conditions
  - Object & event ontology
  - Own and other vehicle conditions
  - ... many other things

- Exiting ODD must be safe
  - Due to environment change (unexpected snow)
  - Due to ODD ontology gap (“what the heck is that???”)
  - Due to equipment failure (potentially using degraded modes)
UL 4600 ODD Prompt Excerpts

- **Travel infrastructure**
  EXAMPLES: types of road surfaces, road geometries, bridge restrictions

- **Object coverage** (i.e., objects within ODD)

- **Event coverage**
  EXAMPLES: interactions with infrastructure

- **Behavioral rules**
  EXAMPLES: traffic laws, system path conflict resolution priority, local customs, justifiable rule breaking for safety

- **Environmental effects**
  EXAMPLES: weather, illumination

- **Vulnerable populations**
  EXAMPLES: pedestrians, motorcycles, bikes, scooters, other at-risk road users, other road users

- **Seasonal effects**
  EXAMPLES: foliage changes, sun angle changes, seasonally-linked events (e.g., Oktoberfest)

- **Support infrastructure, if any is relied upon**
  EXAMPLES: types of traffic signs, travel path geometry restrictions, other markings

- **Localization support, if relied upon**
  EXAMPLES: GNSS availability, types of navigation markers, DSRC, other navaids

- **Compliance strategy for traffic rules**
  EXAMPLE: enumeration of applicable traffic regulations and ego vehicle behavioral constraints

- **Special road user rules**
  EXAMPLES: bicycles, motorcycles/lane splitting, construction systems, oversize systems, snowplows, sand/salt trucks, emergency response systems, street sweepers, horse-drawn systems

- **Road obstructions**
  EXAMPLES: pedestrian zone barriers, crowd control barriers, police vehicles intentionally blocking traffic, post-collision vehicles and associate debris, other road debris, other artificial obstructions

© 2019 Philip Koopman
Autonomy Pipeline candidate best practices & pitfalls

- Sensing (e.g., correlated sensor faults)
- Perception (e.g., brittle perception, ontology gaps)
- Machine learning (e.g., overfitting)
- Planning (e.g., plan exceeds vehicle capability)
- Prediction (e.g., mis-predictions, sudden changes)
- Trajectory & control (e.g., degraded vehicle capabilities)
- Timing (e.g., loss of control loop stability)
System, Environment, Lifecycle

“Item” covered by safety case includes safety related:
- Autonomy (sensors, algorithms, actuators)
- Vehicle (safety related within autonomy purview)
- Maintenance and inspection procedures
- Lifecycle issues and supply chain
- Data sources and feeds, including maps, ML training

Assumptions & supporting requirements
- ODD characterization
- Road infrastructure support
- Procedural support (e.g., safety related inspections)
Maintenance & Inspections

- Safety related maintenance
  - What maintenance is required for safety?
  - Are procedures documented?
  - How do you know it is done effectively?

- Safety related inspections
  - What/when are inspections required?
  - Detection of vehicle & infrastructure problems (e.g., loose wheel)
  - Are you trusting casual passengers with life critical inspections?
    - (Really? Is that a good idea?)
Lifecycle & Supply Chain

- Item has valid safety case at all times once deployed

- Safety related aspects of lifecycle
  - Requirements/design/ML training
  - Handoff to manufacturing
  - Manufacturing & deployment
  - Supply chain
  - Field modifications & updates
  - Operation
  - Retirement & disposal

- Update distribution & integrity
  - Version control & configuration management

Is sensor cleaning fluid life critical?

There is no “captain of the ship”
- Autonomy must assume responsibility

Interacting with people
- Occupants, cargo loading
- Pedestrians & mobility device users
- Other drivers
- Special populations
- Misuse, pranks, malfeasance

Safety related lifecycle participants
- Inspection & maintenance accuracy

Safety culture for all stakeholders
Inductive proofs are never complete
- The black swan problem – you don’t know what you don’t know

Addressed via:
- Extensive use of prompts for better coverage
- Epistemic defeaters (e.g., pitfalls)
- Monitoring required for assumptions and unknowns

Deploying with uncertainty
- You will deploy believing you are acceptably safe
- Use monitoring to reduce margin of belief uncertainty

Every observed swan is white. Therefore all swans are white.
Assessment: Trust and Verify

Self-audit
- Audit safety case for completeness
- Check technical aspects for reasonableness
- In close collaboration with the development team

Independent assessor
- Independence from developer & competence must be documented
- Check and balance on self-audit
- NOT expected to find technical defects

Developers must “own” safety
- Audits & assessments serve as a check and balance
Feedback used to mitigate risk of unknowns

- **Within product**: incidents trigger safety case update
- **At Assessment**: updates trigger assessments
- **Standards Process**: emergent issues trigger ~yearly standard update
Component Assessment

- Generalized idea of System Element out of Context (SEooC)
  - Hardware and/or software
- Idea: design-by-contract component interface
  - Assured properties (services; functions)
  - Assumptions made by component
    - Must match promises made by system
  - Component assurance context
    - Fault model
    - Subset of UL 4600 clauses assessed
  - Can assess SEooC conformance independent of system
Change & Impact Analysis

• Continual changes
  ● System functionality update
  ● Different ODD (changing ODD scope; surprises)

• Assessment in response to changes:
  ● Impact analysis
  ● If required: Update safety case
  ● If safety case updated: Update self-audit
  ● If “big” safety case change: Independent Assessment update

• “Size” of change relates to safety case, not lines of code
  ● Impact analysis informs scope of self-audit/assessments
Prompt Elements vs. Integrity Levels

- **Prompt element deviation categories:**
  - **Mandatory / Required / Highly Recommended / Recommended**
    - E.g.: “REQUIRED” can only deviate if intrinsically inapplicable

- **Integrity levels**
  - Define at least two integrity levels: **life critical & injury**
    - OK to adopt more and/or existing levels (e.g., ASIL, SIL, DAL)
  - Define level of rigor/technique use based on integrity level

- **Example: Static analysis**
  - **Required** that static analysis is used to some degree
  - Coverage, tools, tool settings **based on Integrity level**
How UL 4600 Works with Others

- **ISO 26262 – starting point**
  - Still relevant to the extent it can be applied
  - Assumes traceability of tests to design with “V”

- **ISO/PAS 21448 & SaFAD – more guidance**
  - Design and validation process framework

- **UL 4600 – #DidYouThinkofThat?**
  - Provides a template for technical safety report
  - Minimum criteria for complete coverage + feedback requirement
  - Lists of positive and negative lessons learned
  - Objective assessment criteria for safety case
UL 4600 Chapter Short Titles

Organized by practitioner skill set

1. Preface
2. Scope
3. References
4. Terms
5. Safety case & arguments
6. Risk assessment
7. Humans & road users
8. Autonomy
9. Software & system engineering
10. Dependability
11. Data & networking
12. Verification & validation
13. Tool qualification
14. Lifecycle concerns
15. Maintenance
16. Metrics
17. Assessment

© 2019 Philip Koopman
Anticipated UL 4600 Technical Benefits

- Catalog of best practices: #DidYouThinkofThat?
  - Avoid missed hazards
  - Avoid pitfalls
  - Mechanism for industry to share without sharing detailed data

- Objective, repeatable independent assessment
  - Self-audit is first level of checks and balances
    - Feedback identifies surprises/gaps
  - Independent assessment is about well-formed safety case
    - Not subjective opinion about whether developer tried hard enough
    - Prompt elements provide a safety case coverage floor
    - But, developer assumes burden for safety

© 2019 Philip Koopman
Get Involved: Submit Comments

- Commenting requires registering as stakeholder
  - E-mail to: <Deborah.Prince@ul.com>
- Use supplied spreadsheet for consideration
  - Please make as concrete & actionable as possible

<table>
<thead>
<tr>
<th>Reviewing Organization:</th>
<th>PUT YOUR ORGANIZATION HERE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Point of Contact:</td>
<td>PUT YOUR NAME and e-mail address HERE; please combine comments</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>#</th>
<th>Page</th>
<th>Clause</th>
<th>Old text</th>
<th>New text</th>
<th>Discussion</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>54</td>
<td>5.2.3.3.c.1</td>
<td>Quote the old text before change</td>
<td>Your proposed new text with change</td>
<td>Explain (could be just &quot;typo&quot; or &quot;format&quot; if that is the issue).</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Comments & Timeline

- Official version & comment spreadsheet via UL CSDS
  - Other public materials and draft at: UL4600.com

- Timeline:
  - Comments due Friday Nov 1\textsuperscript{st} via CSDS upload
  - Potentially voting draft in December

- Will Stakeholder names be public?
  - Stakeholder list itself is private
  - However, all preliminary review comments are public & attributed to commenter