FDA SIR: Architecturally-Integrated Hazard Analyses for Medical Application Platforms (NSF CNS-1565544)

CPS PRINCIPAL INVESTIGATOR MEETING
PI: John Hatcliff (KSU – hatcliff@ksu.edu)

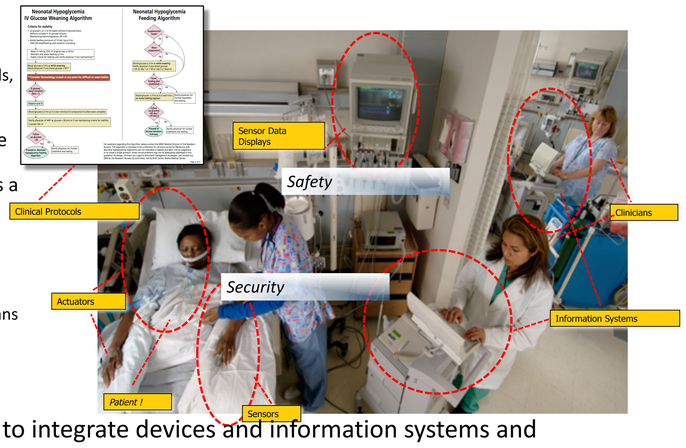
Problems / Challenges

Technology Solutions

Outreach/Impact

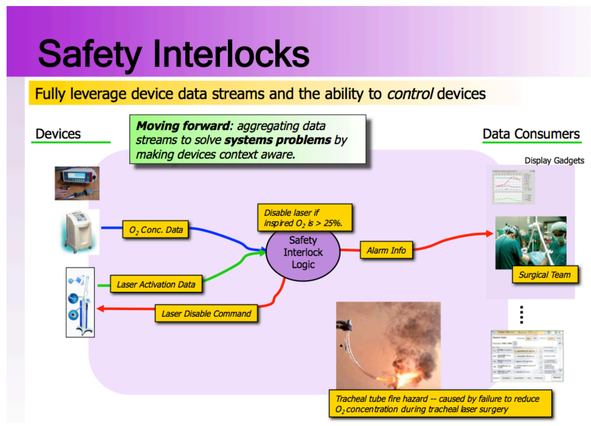
Lack of "System of Systems" Support

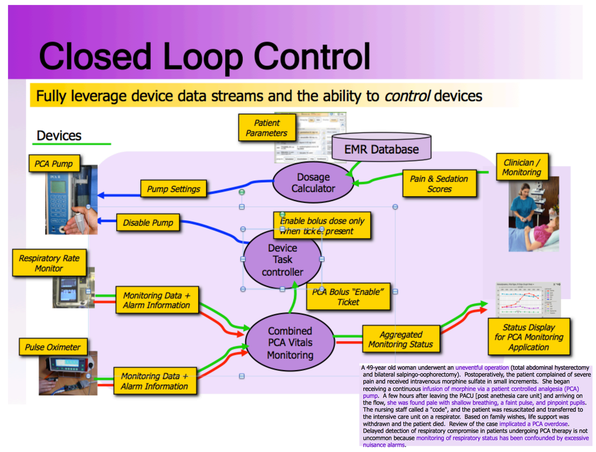
- Delivering modern medical care involves complex cyber-physical systems…
  - many medical devices, electronic medical records, clinicians/care-givers
    …all working together to achieve a goal
- Although most modern medical devices have some form of connectivity, they are not integrated so that they can work together as a system
  - devices are "unaware of their context", e.g., details of patient parameters, history, current procedures they may impact/distort readings
  - data from multiple devices is not combined to produce more meaningful information to clinicians
  - actions of multiple devices cannot be automatically coordinated to achieve greater safety and efficiency



There is no means to integrate devices and information systems and coordinate their actions as a cyber-physical *system of systems*

What Could be Achieved with System of Systems (SoS)?

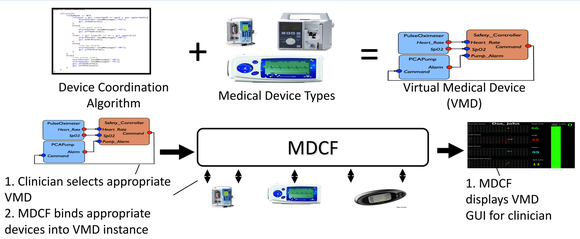




Medical Application Platforms (MAPs)

The Medical Device Coordination Framework (MDCF)

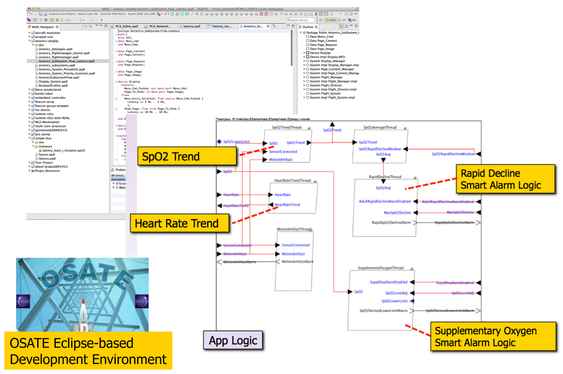
- Our project builds on an open source *Medical Device Coordination Framework* – a *medical application platform (MAP)* for integrating medical devices into systems
- The MDCF provides…
  - Publish-subscribe real-time middleware for integrating devices
  - A component-based application (app) environment for developing and running algorithms that coordinate the device data flows and actions
- Together the platform, app, and connected devices form a Virtual Medical Device – a composite system device composed of individual devices



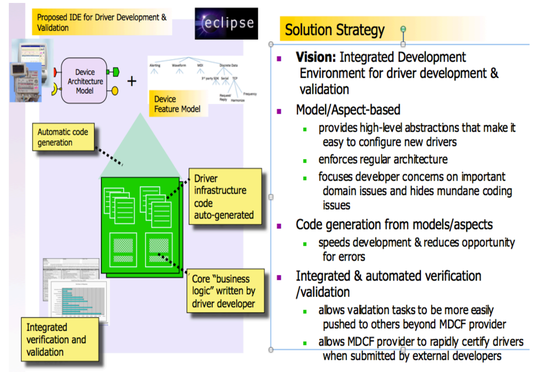
The MDCF aligns with the ASTM standard for an *Interoperable Clinical Environment* (ICE) developed by the CIMIT MDPnP project.

Component-based Development for Coordination Apps

Apps are developed in a model-based development environment based on the industry standard Architecture and Analysis Definition Language (AADL)



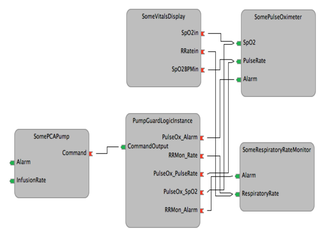
OSATE Eclipse-based Development Environment



Solution Strategy

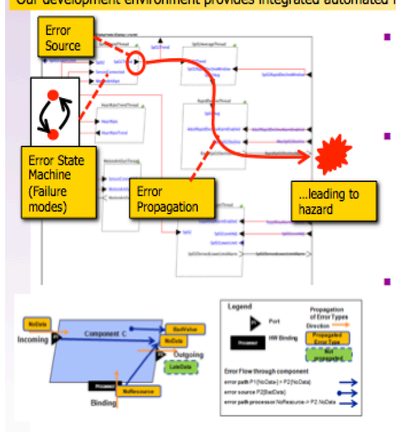
- **Vision:** Integrated Development Environment for driver development & validation
- Model/Aspect-based
  - provides high-level abstractions that make it easy to configure new drivers
  - enforces regular architecture
  - focuses developer concerns on important domain issues and hides mundane coding issues
- Code generation from models/aspects
  - speeds development & reduces opportunity for errors
- Integrated & automated verification /validation
  - allows validation tasks to be more easily pushed to others beyond MDCF provider
  - allows MDCF provider to rapidly certify drivers when submitted by external developers

Components are composed to form coordination apps



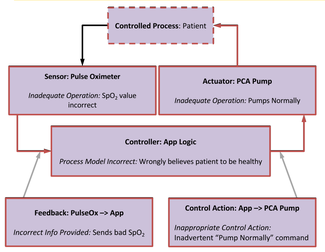
Component-wise Hazard Analysis and Risk Management

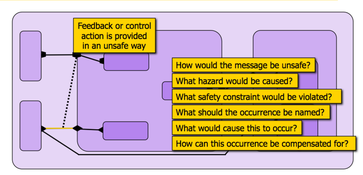
Current risk management activities are highly manual and not well-integrated with formal development artifacts. Our development environment provides integrated automated formal support for risk management activities.

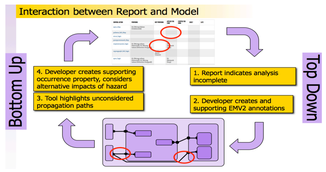


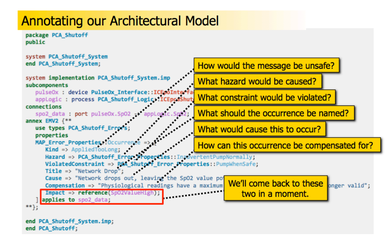
- AADL includes an Error Modeling language dedicated to modeling errors and propagation leading to hazards
- Provides the core of formal architecture-integrated risk management activities from which steps in common hazard analysis such as FEMA, FTA, can be automatically derived.
- Allows engineers to compositionally model aspects of system as it is experiencing faults failures.
  - Necessary for supporting compositional approaches to system safety

Our novel approach involves integrating Leveson's STPA Hazard Analysis with formal architecture models in AADL. Due to our auto code generational from models, hazard analysis is directly traceable to source code and architecture infrastructure components.



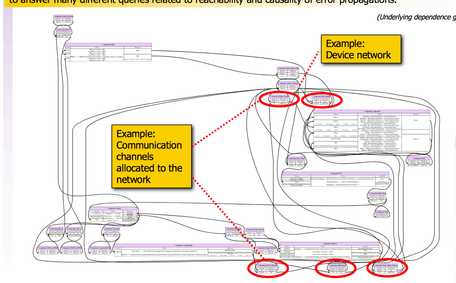


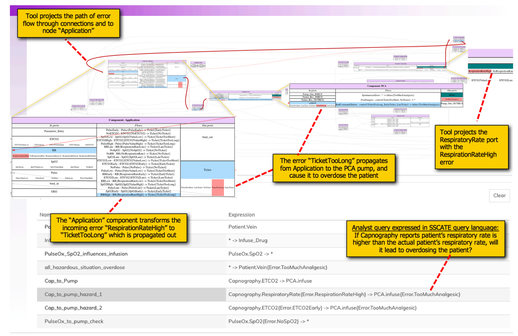




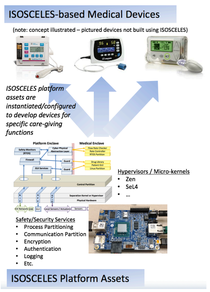
Automated Analysis & Query Capabilities

The SSCATE Analysis Tools automatically generate a dependence graph (behind the scenes) that can be used to answer many different queries related to reachability and causality of error propagations.





Industry Collaboration



Adventium LABS

DoD and DHS CPSSec funding with Adventium Labs provides technology transition pathway

Standards & Regulatory Policy

PIs are members of the AAMI / UL 2800 standard committee tasked with writing a family of standards for safety/security of medical device interoperability

- Safety/security requirements of architectures for Medical Application Platforms (MAPs)
- Framework for compositional certifications of MAPs
- Guidelines for evaluating compliance to requirements

We are actively engaged with FDA engineers to develop science-based inputs for forming regulatory policy for interoperable medical systems

- Safety evaluation eco-system for medical device interoperability platforms
- Example hazard analyses, mock 510(k) regulatory submissions for apps and other MDCF components
- Guidelines for development of third-party certification regime

Educational Material

The Open PCA Pump provides open source development artifacts for a realistic medical device – developed collaboratively with industry and FDA engineers

- User need documents and background resources for PCA Pumps
- 80+ page requirements document
- Architecture models in AADL
- Assurance case in NOR-STA commercial assurance case tool
- Suggested student projects
- Lectures on safety-critical system development – requirements, hazard analysis, assurance cases, etc.

Team

Kansas State University

John Hatcliff
Robby
Venkatesh Ranganath

Brian Larson
Hariharan Thiagaragan

Collaborators

Adventium Labs

Todd Carpenter
Rand Whillock

FDA & Underwriters Lab

Paul L. Jones (FDA)
Sandy Weininger (FDA)
Yi Zhang (FDA)
Anura Fernando (UL)