Defining & Developing a Digital Twin Through Creating Synergies Between Cyber, Software, Reliability & Safety

11/2/2023

Mary "Lisa" Williams Bates
Abhishek Ram

RAM XV Training Summit
Huntsville, Alabama

Agenda

- Defining the trade space for an affordable, efficient digital twin
  - What is a Digital Twin?
  - Engineering Tenets to Design a Product with a Digital Twin
  - Concepts needed for Digital Twin
- Critical Items Synergies
- How do you provide a real digital engineering end to end experience for the life cycle of a product from design to end of life to include training your people to work in that type of environment?
- What does an Open Systems Dev*Ops low code to no code infrastructure for low cost product development really look like?
- Effective tools for knowledge sharing for all generations. Combining videos, wikis, sharepoints, teamwork cloud with search engine optimization techniques for your on premises intranet.

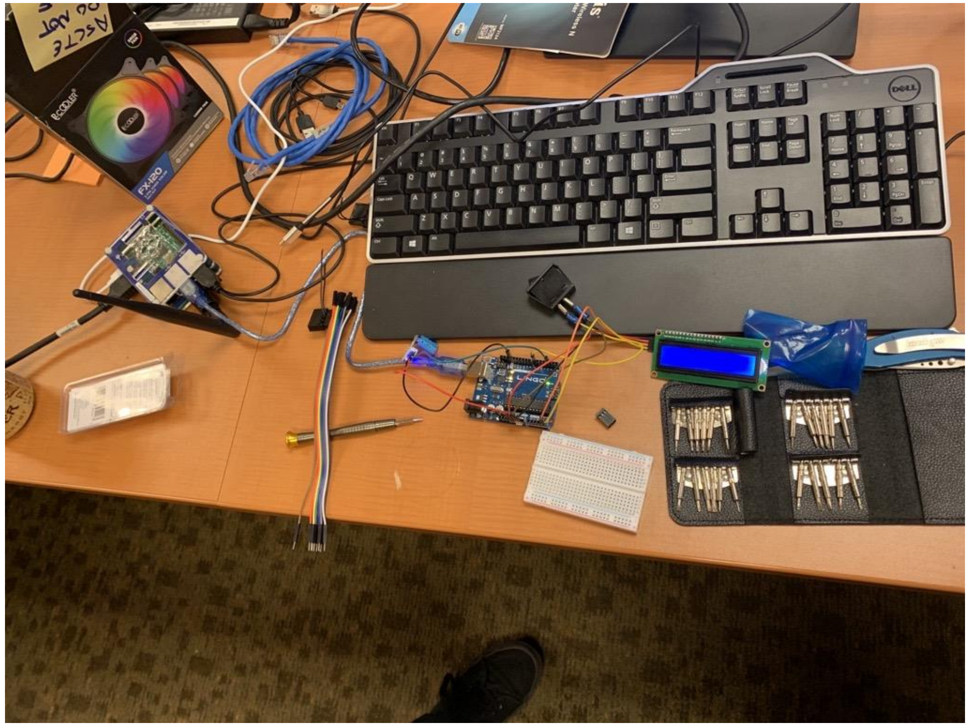
Critical Item Synergies

- What do Machine Learning, Model Based Systems Engineering, Cyber, Software, Reliability and Safety functions have in common?
  - Critical Items tie Safety, Cyber, and Reliability
- How can we synergize them in a more automated way to build a practical affordable digital twin?
  - Models Based Systems Engineering tie the Critical Items to features used in Machine Learning Algorithms to solve a known specific equipment failure mode.
    - Do you know the problem you are trying to solve and can you get a data pipeline to contribute to robust feature and data analysis?
    - Don't start with the algorithm, start with the infrastructure and data pipeline.

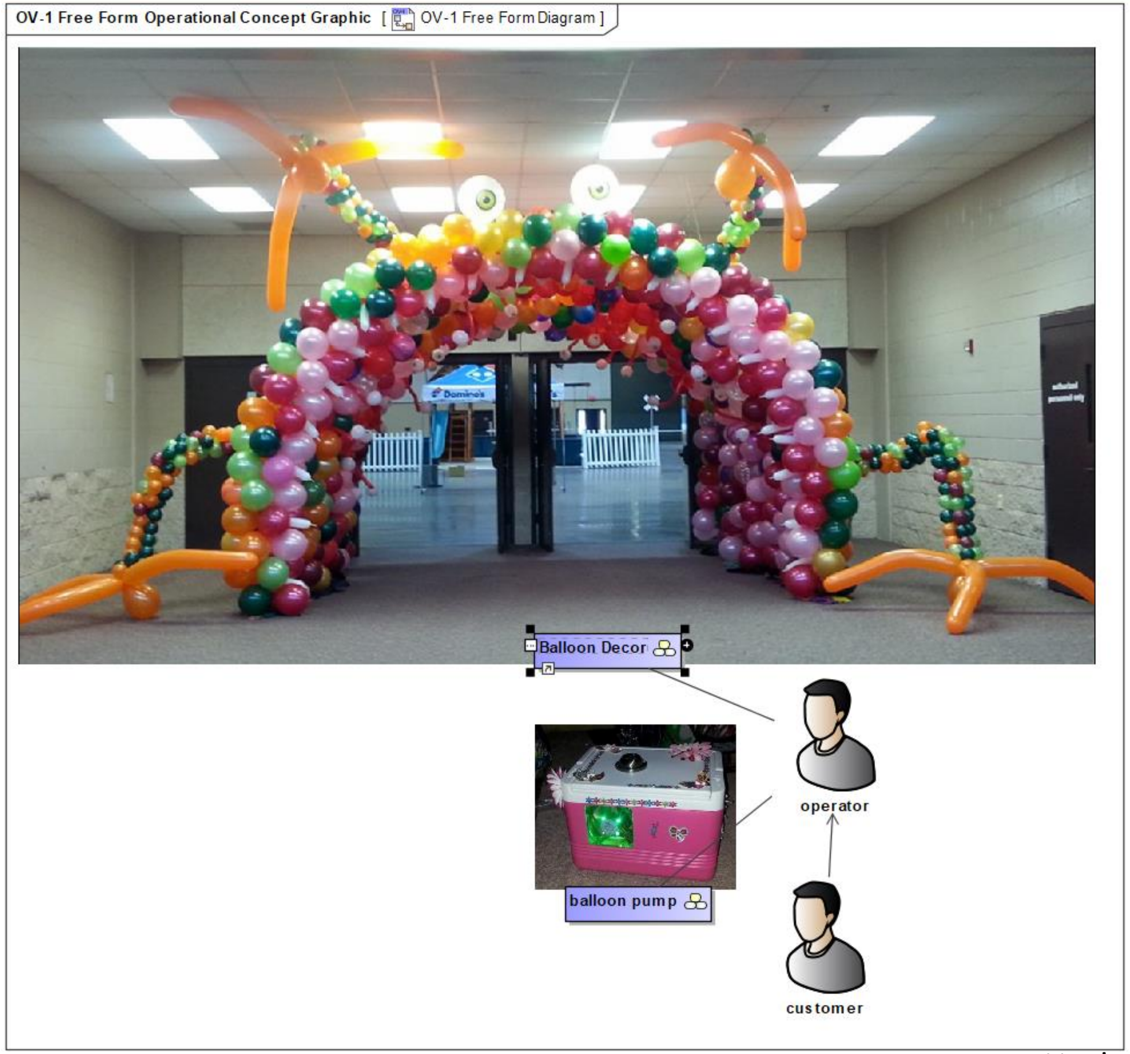
SysML Views Tailoring

This approach provides a fully modeled digital twin with the following views:

- Use Cases
- Requirements Model with traceability
- CONOPS
- Overview Diagram – OV-1
- FMECA – account for environmental variables and mechanisms of failure

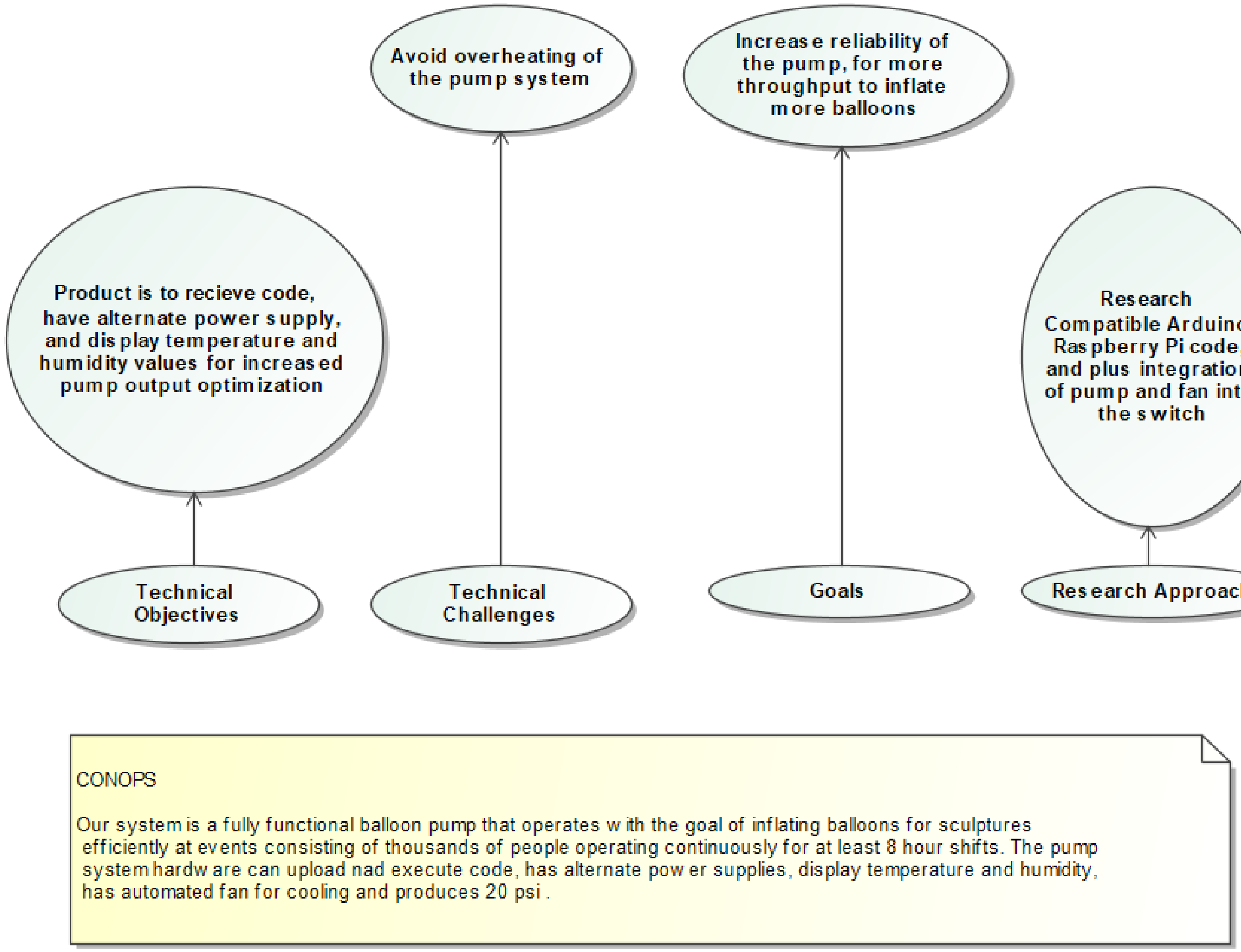


Specific Problem to Solve



- Simple real-world example that requires Reliability, Safety, Software and Cyber to build a digital twin
- Balloon pump that overheats and or loses power before mission completion
  - Elimination of those 2 issues in order to complete mission requires reliability to understand failure mechanism
  - Sensors to monitor failure mechanism
  - Safety to ensure no injuries from failure mechanism
  - Software to monitor and capture data
  - Cyber to ensure protection of hardware and software

CONOPS


Avoid overheating of the pump system
Increase reliability of the pump, for more throughput to inflate more balloons
Product is to recieve code, have alternate power supply, and display temperature and humidity values for increased pump output optimization
Research Compatible Arduino, Raspberry Pi code, and plus integration of pump and fan into the switch
Technical Objectives
Technical Challenges
Goals
Research Approach
CONOPS
Our system is a fully functional balloon pump that operates with the goal of inflating balloons for sculptures efficiently at events consisting of thousands of people operating continuously for at least 8 hour shifts. The pump system hardware can upload nad execute code, has alternate power supplies, display temperature and humidity, has automated fan for cooling and produces 20 psi .

What is a Digital Twin? (1/2)

- A review of the technology standards for enabling ... | Digital Twin (digitaltwin1.org) – This article provides an overview of different standards governing parts of the digital twin.

- There are many different variations of the definition for a digital twin which can be found in a simple search on Google.

  - VanDerHorn and Mahadevan identified 46 different definitions across literature, and these definitions share common themes, including that the digital twin must be composed of a “virtual representation”, a “physical system”, and the ability to “update via an exchange of information”, though other definitions may consist of synonymous word choice.

  - Subsequently, the authors defined a digital twin as a “virtual representation of a physical system (and its associated environment and processes) that is updated through the exchange of information between the physical and virtual systems.

What is a Digital Twin? (2/2)

A minimum viable product (MVP) digital twin for condition monitoring a piece of equipment is composed of:

Digital Twin MVP Element	Examples of MVP Element	Purpose of MVP Element
Systems Model	Cameo, SparksEA, Rhapsody	Provides the ability to model and visualize all aspects of the system
A System or Piece of Equipment	Component(s) of a System or the System Itself	Several publications/studies define a digital twin to be a virtual representation of a physical system of interest.
Design Data	System Requirements, ICD, Key performance parameters, Specifications, Interfaces, Design Configuration	• Design data describes the desired normal behavior of the system and can be subsequently used to identify abnormal system behavior. • In the example of a balloon pump, design data can include desired flow rate, outlet pressure and temperature, etc.
Internal Equipment Data	LRIDs, Logs, Sensor Data	Internal equipment data can describe the actual observable behavior in the system, and combined with knowledge of the design data, can be used to verify if the system is behaving as expected or not.
External Equipment Data	CONOPS, Failures, Field Configuration	This data includes how the system is being used and how it interacts with its environment. Even if the system itself is behaving as expected, external influences can impact system behavior and subsequently lead to abnormal system behavior.

Concepts Needed for Digital Twin Development

Digital Twin Development Concept	Purpose of Concept
Agile/Scrum Methodology	Incremental development of the product to provide value to the customer/stakeholders at every step.
*Ops Development	
Model Based Systems Engineering (MBSE) - SysML	SysML is a standard architecture modeling language that supports MBSE, where the goal of MBSE is to provide an approach for modeling system behavior, agnostic of the system itself and the tools.
Minimum Viable Product	Demonstrates early ability to meet digital twin requirements and provides/verifies next steps for agile development.
Risk and Opportunity	
Integrated Digital Environment	Provides an architectural framework in which the digital twin exists, and it can complete all its intended functions.
Acquisition Phases	
Data Management and Design	Stipulations of the data needed to complete the digital twin function and how data is intended to flow throughout the integrated digital environment.
Software for Automation	

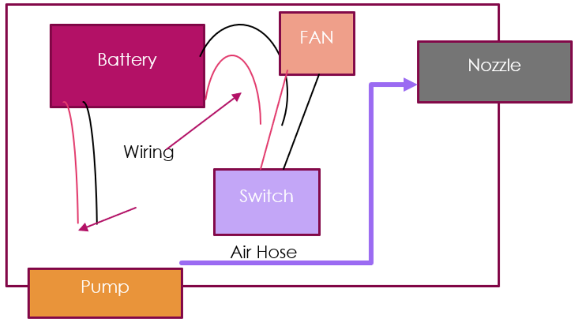
Engineering Tenets to Design a Product with a Digital Twin

- Shared Digital Environment with Customer
  - Customer has to allow for some sort of Intellectual Property Protection in Shared Digital Environment
- Software / Hardware/ Firmware in a *Ops Pipeline – Not just Jenkins but Openshift, or Rancher, or Podman containerized Microservices. Use of FOSS and Open Architecture to the fullest extent possible.
  - Why re-invent the wheel when
- Automate and prepare low code no code technical stack
  - Engineers working in the CI/CD pipeline can go do their job from a webpage or dashboard
  - Shared document creation, shared algorithm libraries, shared design libraries, persistent locked down URLs for data automation

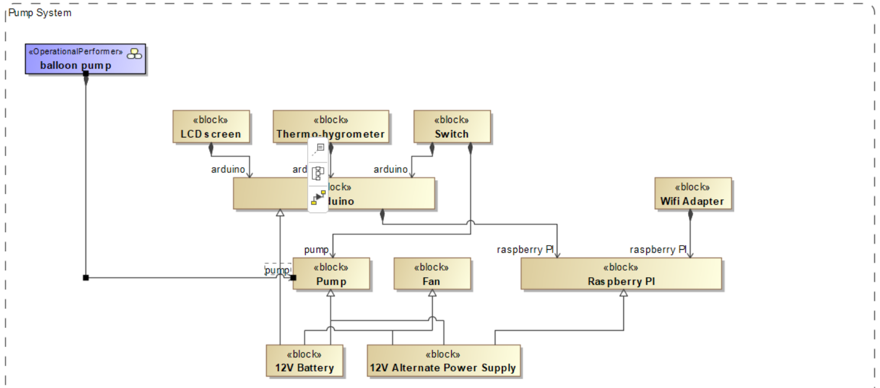
Tools Enable Communication, Collaboration and re-use

Balloon Pump Project – Experiment to Explore Concepts of Digital Twin

- Small experiment to find synergies and model a Digital Twin – Team was 9 Highschool seniors and 9 new career engineers. No one had any previous training in Digital Engineering or AIML.
  - The objective of this project is to create a portable balloon pump, with a digital twin for condition monitoring. 1 known failure mode of pump overheating.
  - Progress checks will include answering: Does the balloon pump work as intended? Is it able to track temperature & humidity? What modeling, hardware, software, documentation is needed for a Minimum Viable Product Digital twin?



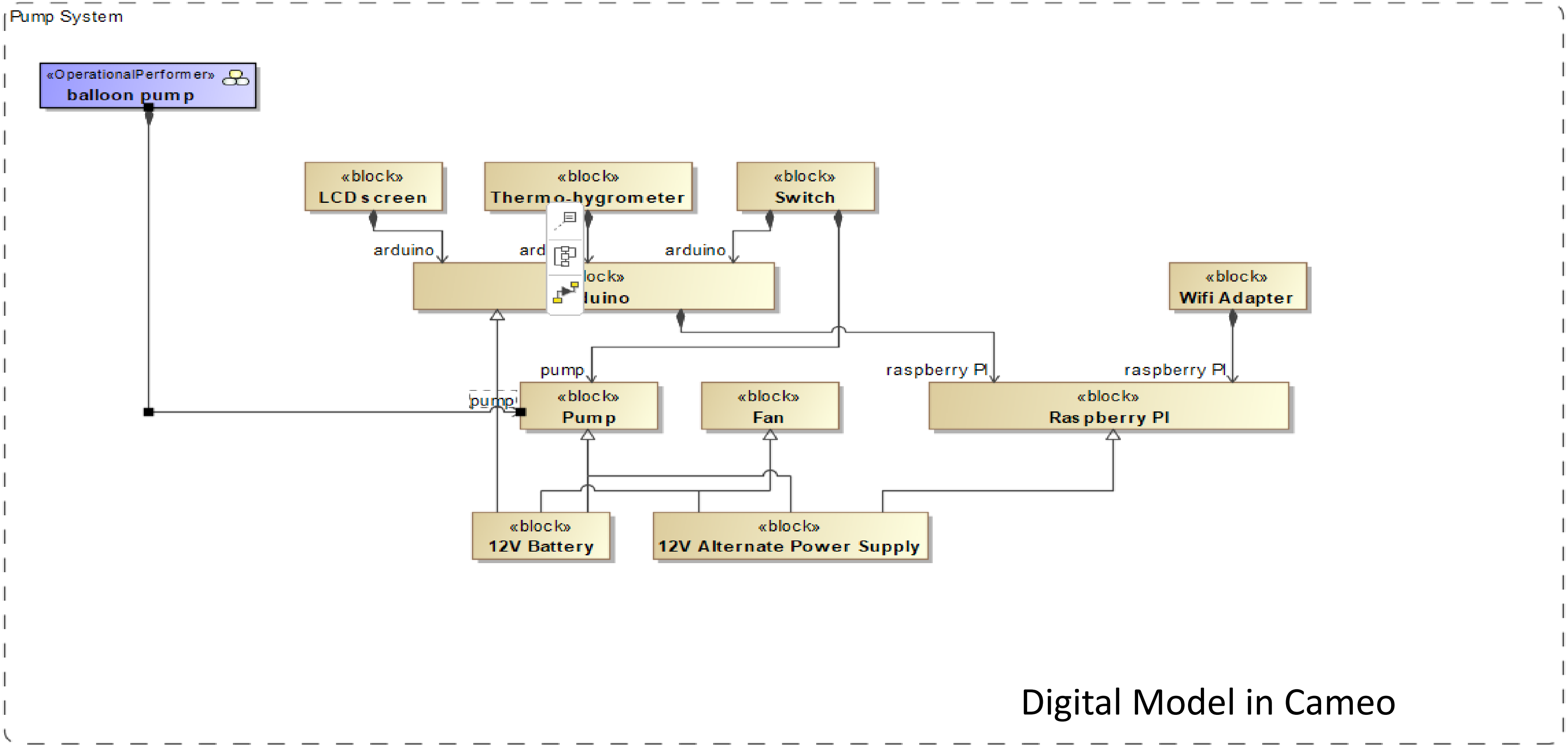
Analog Model in Powerpoint



Digital Model in Cameo

Balloon Pump Project – Digital Twin From Familiar DoDAF views to Model Based Analytics

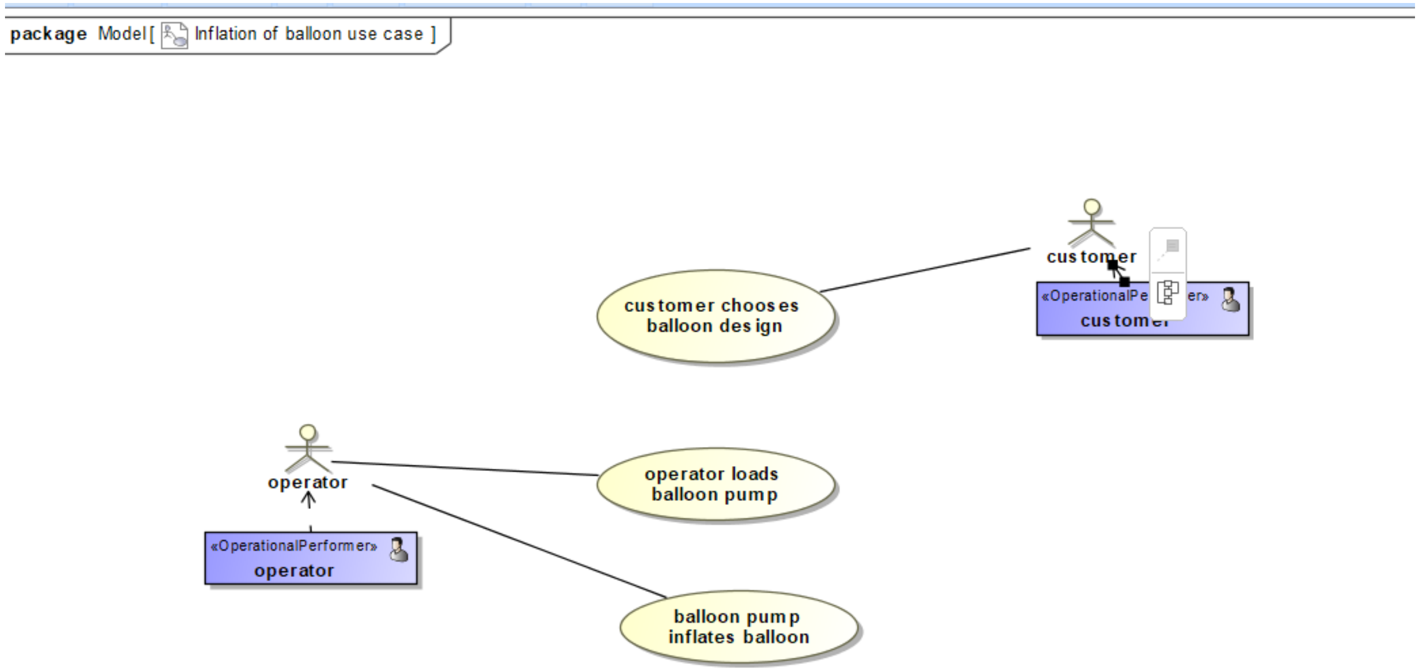
- Relational OV-1 operational view relation to Model based black box Diagrams

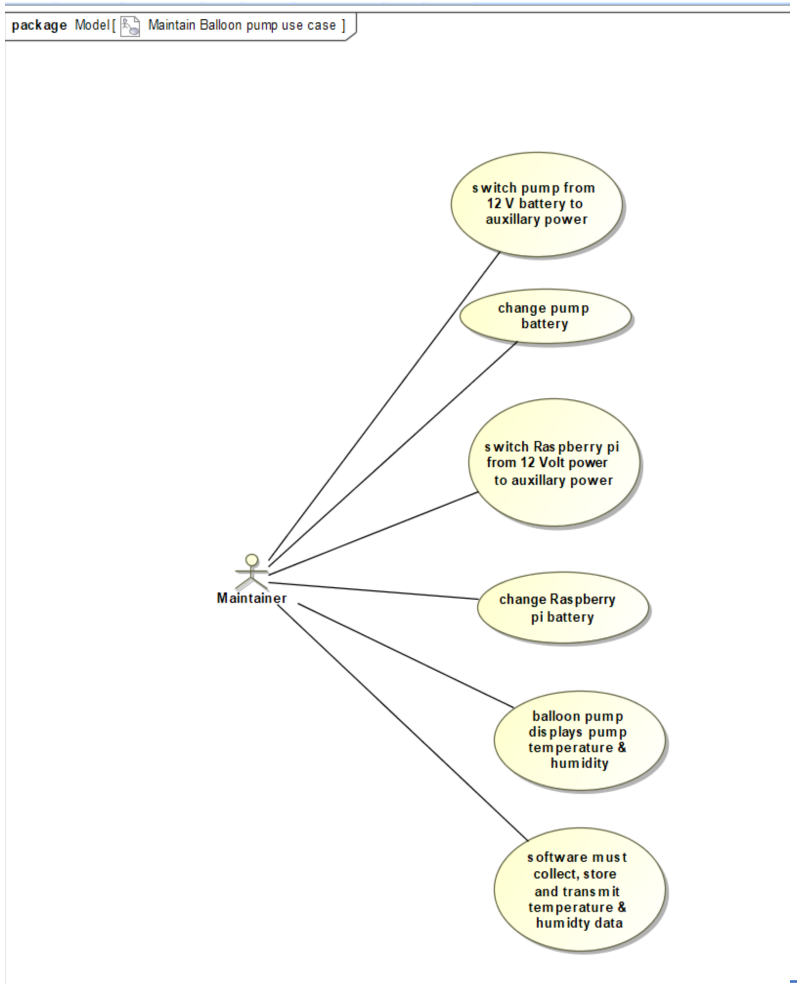


Digital Model in Cameo

Balloon Pump Project – Digital Twin From Familiar DoDAF views to Model Based Analytics

- Relational OV-1 operational view relation to Model based Use Cases





Requirements Development & Tracing

- Develop Use Cases and derive requirements for Minimum Viable Product Digital Twin

Criteria

Scope (optional): Drag elements from the Model Browser ... Filter:

#	Use Case	Name	Text	Id	Owner	Traced To	sortPriority
1	balloon pump inflates ballo	1 SW_1	This Product shall be able to upload all necessary code so that the pump can be controlled digitally. To test this, plug into power, turn on and check setting for proper results.	1	Model	Arduino 12V Battery 12V Alternate Power Fan LCD screen Raspberry PI Switch Wifi Adapter Thermo-hygrometer Pump System Pump	High
2	balloon pump inflates ballo change Raspberry pi batte change pump battery	2 HW_1	This product shall have an alternate power supply for the electronics to stay online and functional whenever necessary. The system will be tested by checking functionality using both the 12V power supply and internal battery independently.	2	Model	12V Alternate Power 12V Battery	Medium
3	change pump battery change Raspberry pi batte	3 HW_2	THe product shall have an alternate supply for the pump so that the hardware stays functional whenever necessary. The system will be tested by checking functionality using both the 12V power supply and the internal battery independently.	3	Model	12V Alternate Power 12V Battery	Low
4	balloon pump displays pun	4 HW_3	The product must display the temperature of the system in a readable form on the display for the user. The system will be validated by outputting the current temperature of the system	4	Model	Arduino LCD screen Raspberry PI Thermo-hygrometer	High
5	balloon pump displays pun	5 HW_4	The product must display the humidity of the system in a readable form on the display for the user. The system will be validated by outputting the current humidity of the system	5	Model	Arduino LCD screen Raspberry PI Thermo-hygrometer	Low
6	software must collect, stor	6 SW_2	The software must collect, store and send sensor collected data of temperature and humidity at 30 second intervals over wifi.	6	Model	Arduino Raspberry PI	Low
7	software must collect, stor	7 SYS_1	The product will not exceed temperature range of 0 Celcius - 38 Celcius. The fan must turn on to regulate system temperature so system does not overheat.	7	Model	Arduino Fan Thermo-hygrometer Raspberry PI	High
8	balloon pump inflates ballo	8 HW_5	The product must be able to output pressure to sufficiently inflate balloons up to 20 PSI. The product will be tested by fully inflating several sizes of balloons.	8	Model	Pump	Low

FMECA

- FMECA using Cameo Safety and Reliability Analyzer

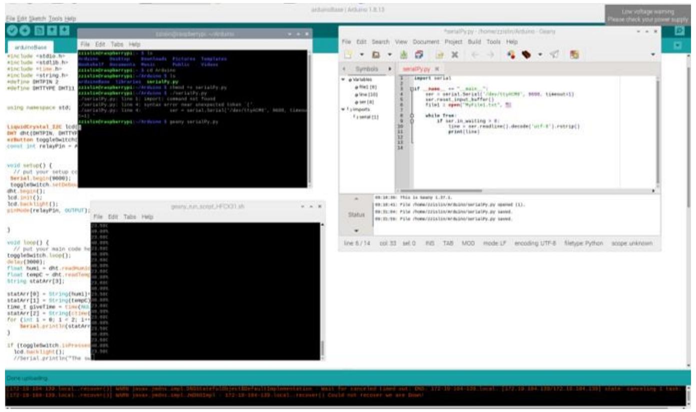
Criteria

Element Type: FMEA Item ... Scope (optional): y PI,Thermo-hygrometer,Wifi Adapter,Arduino ... Filter:

#	Id	Name	Classification	Item	Failure Mode	Local Effect Of Failure	Final Effect Of Failure	SEV	Cause Of Failure	OCC	Prevention Control	Detection Control	DET	Ox D	RPN
1	F-1	F	electrical	12V Battery	Loss of Power	Pump System Stops working	System Stops working	1	battery need charged	1	auxiliary power	pump slows down	1	1.0	1.0
2	F-2	F	mechanical	Pump	Overheating	Pump Stops working	System Stops working	1	over working of system	1	monitor increase in	increase in tempera	1	1.0	1.0

Example MVP Digital Twin of a Balloon Pump

Software & Sensor Data



Requirements

Criteria

Scope (optional): Drag elements from the Model Browser   Filter:

#	Use Case	Name	Text	Id	Owner	Traced To	sortPriority
1	balloon pump inflates ballc	1 SW_1	This Product shall be able to upload all necessary code so that the pump can be controlled digitally. To test this, plug into power, turn on and check setting for proper results.	1	Model	Arduino 12V Battery 12V Alternate Power Fan LCD screen Raspberry PI Switch Wifi Adapter Thermo-hygrometer Pump System Pump	High
2	balloon pump inflates ballc change Raspberry pi batte change pump battery	2 HW_1	This product shall have an alternate power supply for the electronics to stay online and functional whenever necessary. The system will be tested by checking functionality using both the 12V power supply and internal battery independently.	2	Model	12V Alternate Power 12V Battery	Medium
3	change pump battery change Raspberry pi batte	3 HW_2	THe product shall have an alternate supply for the pump so that the hardware stays functional whenever necessary. The system will be tested by checking functionality using both the 12V power supply and the internal battery independently.	3	Model	12V Alternate Power 12V Battery	Low
4	balloon pump displays pun	4 HW_3	The product must display the temperature of the system in a readable form on the display for the user. The system will be validated by outputting the current temperature of the system	4	Model	Arduino LCD screen Raspberry PI Thermo-hygrometer	High
5	balloon pump displays pun	5 HW_4	The product must display the humidity of the system in a readable form on the display for the user. The system will be validated by outputting the current humidity of the system	5	Model	Arduino LCD screen Raspberry PI Thermo-hygrometer	Low
6	software must collect, stor	6 SW_2	The software must collect, store and send sensor collected data of temperature and humidity at 30 second intervals over wifi.	6	Model	Arduino Raspberry PI	Low
7	software must collect, stor	7 SYS_1	The product will not exceed temperature range of 0 Celcius - 38 Celcius. The fan must turn on to regulate system temperature so system does not overheat.	7	Model	Arduino Fan Thermo-hygrometer Raspberry PI	High
8	balloon pump inflates ballc	8 HW_5	The product must be able to output pressure to sufficiently inflate balloons up to 20 PSI. The product will be tested by fully inflating several sizes of balloons.	8	Model	Pump	Low

Hardware

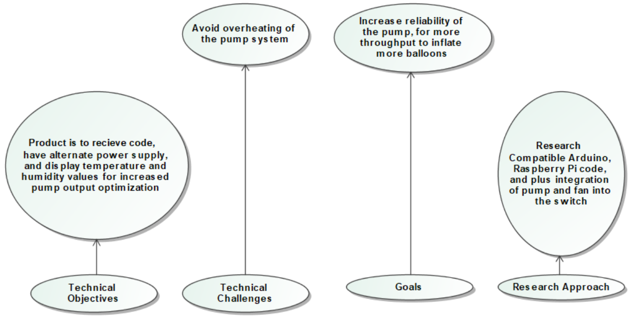
Add New Add Existing... Clone Row Delete Remove From Table Columns Report Export

Criteria

Element Type: FMEA Item   Scope (optional): y PI,Thermo-hygrometer,Wifi Adapter,Arduino   Filter:

#	Id	Name	Classification	Item	Failure Mode	Local Effect Of Failure	Final Effect Of Failure	SEV	Cause Of Failure	OCC	Prevention Control	Detection Control	DET	Ox D	RPN
1	F-1	F	electrical	12V Battery	Loss of Power	Pump System Stops working	System Stops working	1	battery need charged	1	auxiliary power	pump slows down	1	1.0	1.0
2	F-2	F	mechanical	Pump	Overheating	Pump Stops working	System Stops working	1	over working of system	1	monitor increase in	increase in tempera	1	1.0	1.0

FMECA



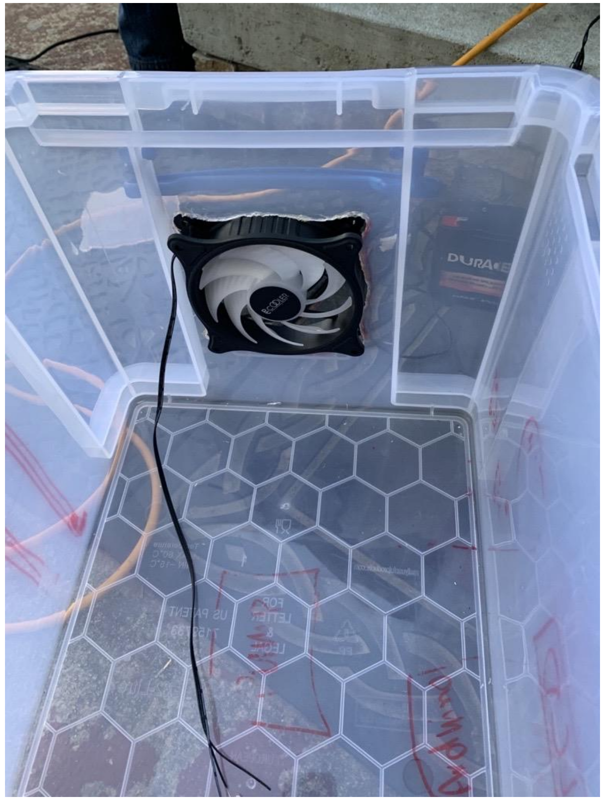
CONOPS

Our system is a fully functional balloon pump that operates with the goal of inflating balloons for sculptures efficiently at events consisting of thousands of people operating continuously for at least 8 hour shifts. The pump system hardware can upload nad execute code, has alternate power supplies, display temperature and humidity, has automated fan for cooling and produces 20 psi .

CONOPS

How Is It Done Today, and What Are the Limits of Current Practice?

- The balloon pump is currently operated manually
  - A user must hold down a button to keep the pump running until the balloon inflates to the desired length.
- The original balloon pump did not contain a fan and temperature sensor to prevent the pump from overheating.
- A powerpoint design diagram, no data collection

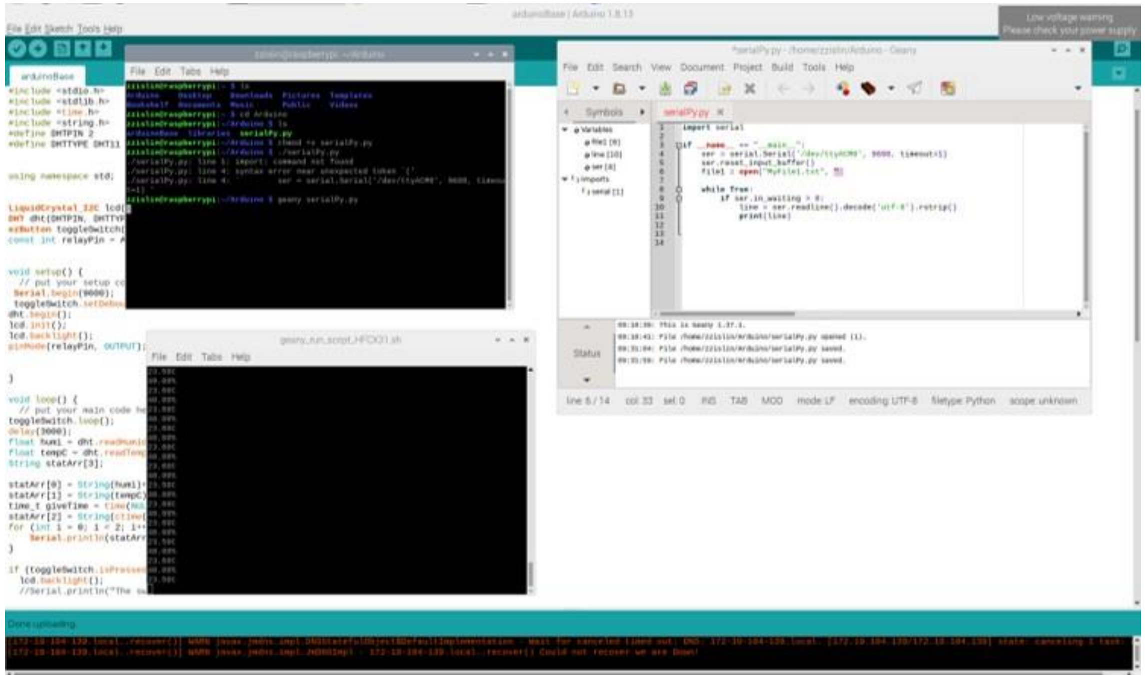


Code and Programming

C++ Arduino Script

- Data connection Between DHT11 sensor, the Arduino, and LCD screen
- Powered and hosted on Raspberry pi
- Python code on Raspberry pi monitors data

  from thermo-hygrometer
- Saves in continuously updating text file
- Will save to remote server



Engineering Tenets to Design a Product with a Digital Twin

Always start with a problem to solve.

- Requirements –
  - Complete missions of 8 hours of continuous use of balloon pump
- Critical Items
  - Items that allow for mission to complete.
  - For a Balloon pump, overheating or loss of battery power due to usage are the failure modes.
  - Overheating is small safety issue when changing out pumps.
- Select a Modeling Language – SysML is king right now but UML and AADL are still in use.
- Select a Tool- Cameo, Rhapsody, Sparks EA, Modelio etc..

Models Enable Communication, Collaboration and re-use

Engineering Tenets to Design a Product with a Digital Twin

- **Have support infrastructure in place – training (infrastructure people first), product lifecycle management tool, IT, Operations, Cyber, software scripting, hardware servers, storage to enable a digital thread**

- Select an Architecture Methodology and Framework

    - Methodology: Techniques, procedures, working concepts and rules how to solve a problem. i.e. Magic Grid, Object Oriented Software Engineering Method (OOSEM), Harmony, SysMOD, Functional Architecture of Systems, Architectural Development Method (ADM)

    - A framework is a picture or a model that guides you to understand which artifacts you should produce when. DoDAF, ToGAF, UAF, Zachman etc..

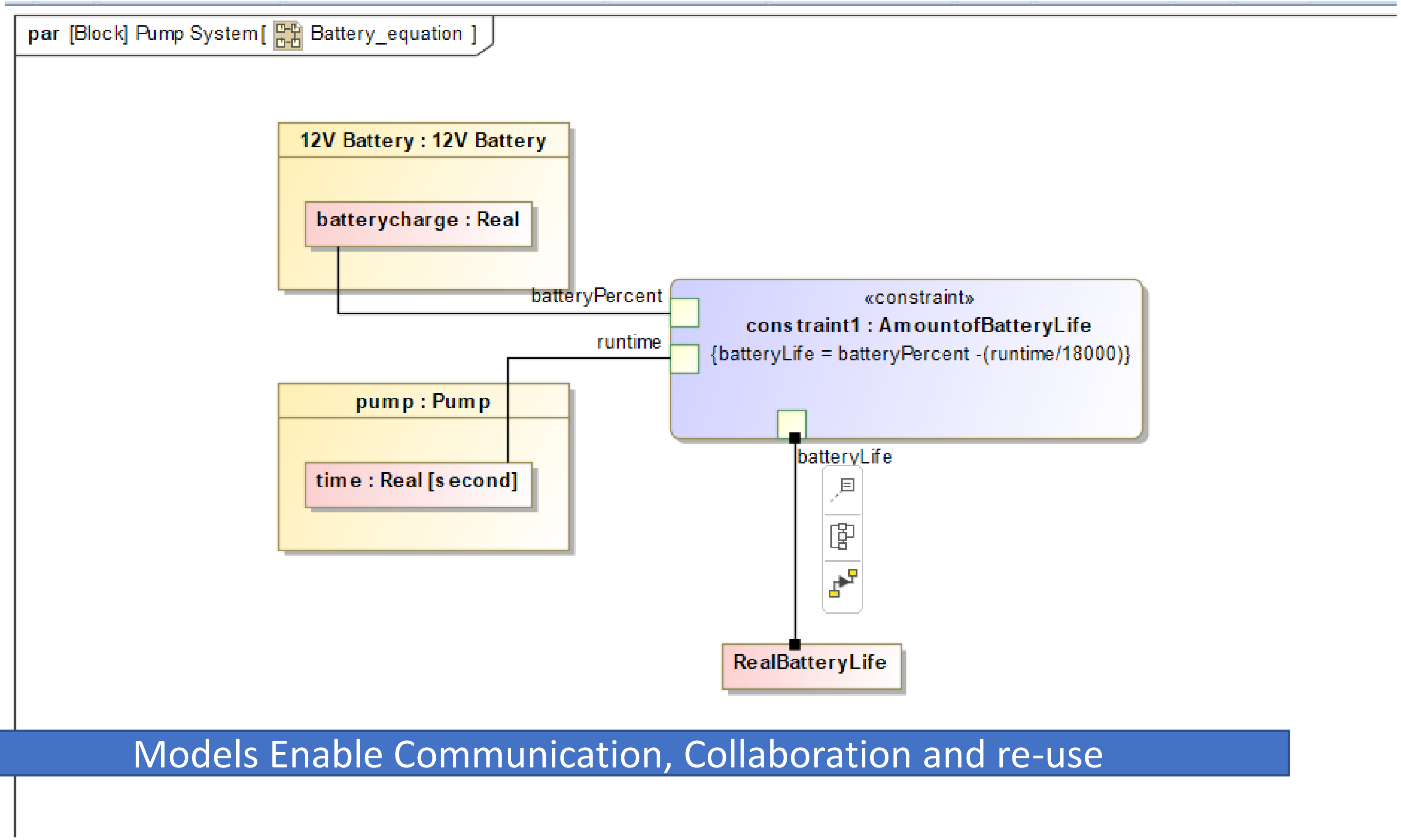
Models Enable Communication, Collaboration and re-use

Engineering Tenets to Design a Product with a Digital Twin

- Have a Model Template Warehouse
  - Mandate every program use the template / model warehouse instead of starting from scratch in the gating process
  - Create a Generic Layer between Physical and Logical architectures
- Have all parts of the lifecycle of a product involved at the start of the program to include Technical Manuals, Training, Reliability, Safety, Logistics Support Analysis.
  - Bi-directional communication between a model – no matter the tool and logistics data, technical manual work packages, reliability simulation and code is not only possible but has been done by companies and military branches. Kessel Run, Black Pearl, Altexsoft, Lockheed Martin, Raytheon, Intercax, Cameo Concept Modeler, IBM Rhapsody Jazz Suite
    - Making this happen requires infrastructure – hardware, software, IT, Cyber and people.
    - You can do it with FOSS or you can get a tool but the no matter the tool you have to still write code and negotiate with tool vendors

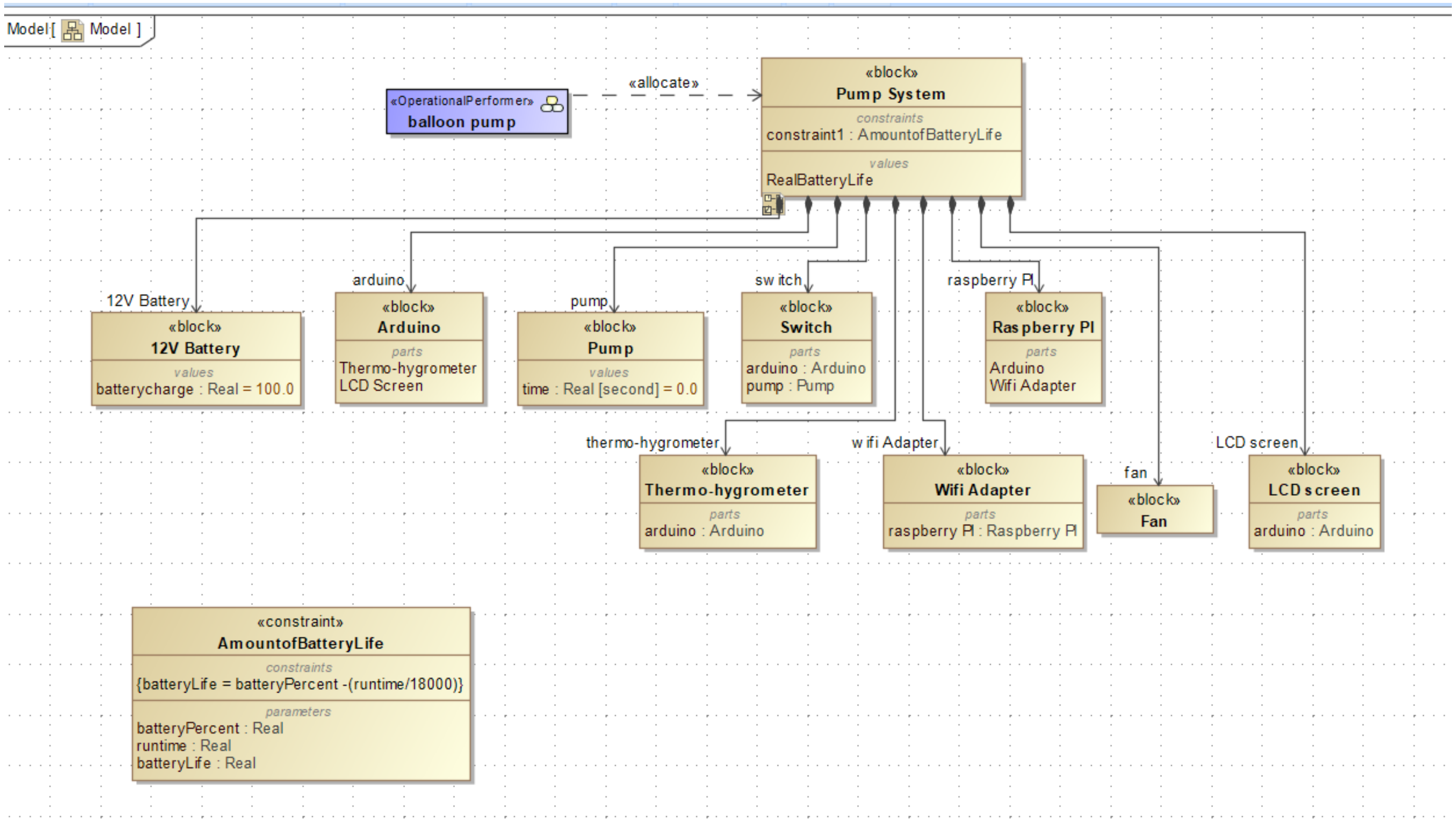
Models Enable Communication, Collaboration and re-use

Cameo Only Parametric



Models Enable Communication, Collaboration and re-use

Add a Hierarchy BDD of the System



Move to Magic Draw to Create Another Constraint for Reliability based on FMECA

Conclusions

- The tools allow for endless possibilities
- Think about Model Hierarchy as Project Usages
- Tying DoDAF, MoDAF, UAF, Zachman to actual components easily
- A lot of Connections are built in and dependent upon licensure and infrastructure
- **Have support infrastructure in place – training (infrastructure people first), product lifecycle management tool, IT, Operations, Cyber, software scripting, hardware servers, storage to enable a digital thread**

Working towards Dev*Ops - Tools that Enable Digital Twin

Open Systems Dev* Ops

What does an Open Systems Dev*Ops low code to no code infrastructure for low cost product development really look like?

- Containerization, Microservices, Continuous Improvement / Continuous Development (CI/CD) Pipeline
  - Why? What does it do for you?
    - Web page interfaces for tools instead of lengthy onboarding installations
    - Shared workspaces – one SME can facilitate good algorithm methodologies across teams of engineers. More confluence type work environment with gitlab or jupyterlab
    - SSO permissions
    - More efficiently utilize the SMEs to grow your workforce with good documentation and code not on someone's computer
    - Unit testing, CI testing, metrics automation for the purposes of software reliability, hardware spec testing, integration testing

Questions?