Engineered Resilient Systems:
Tradespace Tools

SERC SSRR – 3 December 2015

Santiago Balestrini-Robinson, PhD; Dane F Freeman, PhD; Valerie B Sitterle, PhD

PI: Tommer R Ender, PhD
Chief, Systems Engineering Research Division
Georgia Tech Research Institute
tommer.ender@gtri.gatech.edu

Simon R. Goerger, PhD
Director, Institute for Systems Engineering Research
Engineer Research & Development Center, US Army Corps of Engineers
simon.r.goerger@erdc.dren.mil

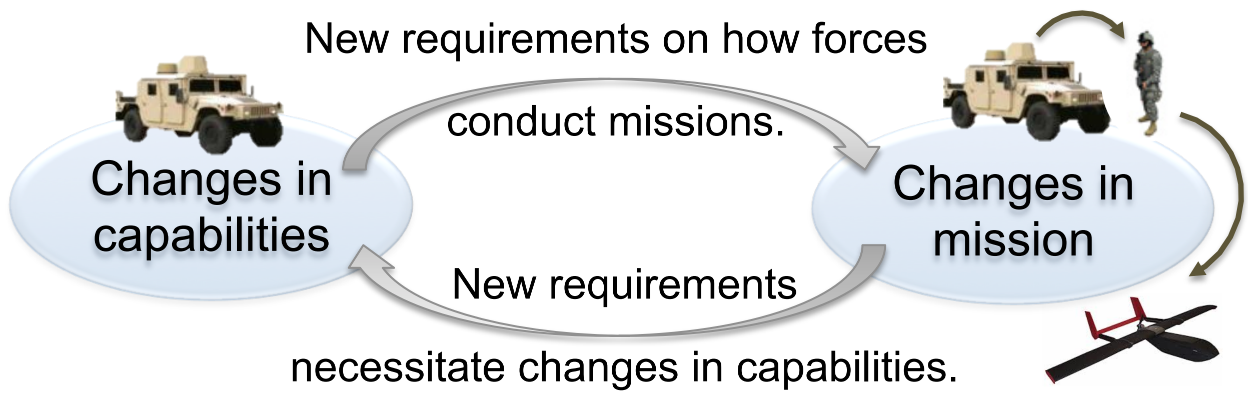
Georgia Tech Research Institute

ERDC Engineer Research & Development Center ISER

ERS & The Notion of a Tradespace

*"**ERS will empower Pre-materiel analysis** with significant impact on:*
- *Requirements Generation*
- *Analysis of Alternatives*
- *Lifecycle Intelligence"*

~ Holland, ERS Overview Dec 2013



What is a Tradespace?

- …the space spanned by completely enumerated design variables – it is the potential solution space
- …the set of program and system parameters, attributes, and characteristics required to satisfy performance standards

The enumeration of a large *tradespace* helps prevent designers from committing to limited point designs and allows them to recognize better design solutions

Why Tradespace Tools?

"In recent years the military services have been faced by **order of magnitude increases** on the severity of the **operational requirements** imposed by their missions and in the **complexity** of the equipment and operations necessary to meet these requirements."

… Though the **need for a systems approach** has been recognized… there has been a lack of tools of systems analysis for **linking together men and mechanisms** into an **integrated analytical framework**.

System models as analytical tools… have always played a vital role in analytical work and a sign of maturity in systems analysis will be the development of < integration across the various> physical models, abstract, and symbolic models.

Models are used as the **basis of evaluation tools** to answer … **which alternative best meets the performance requirements** of the system within the imposed constraints and with the given inputs?"

~ Albert Shapero & Charles Bates (1959)

SE as a Historically Imperative Process

Most state of the art SE analysis – and education – is imperative:

Surrogate Models

Design Variables → Link to Models as abstraction of System → Execute to produce output Design Space → Analyze

Object-Oriented design patterns

MCDA

Visualization

Conventional Engineering – Moving from abstract design to physical realization

Abstract

Concrete

Software Engineering – Moving from a real world problem to its expression and representation by virtualization and coding of software design

ERS TRADESPACE: Evolving Systems Engineering

How machines think	1G	Binary Machine Code	Directly Executable	Cognitive Load ↑
	2G	Assembly		
More abstract	3G	Most Languages	Imperative. Object oriented – encapsulates state and behavior	
	4G	Declarative	Domain Specific Languages	
How humans think	5G	Constraint Programming		

We aim to evolve how a systems engineer more effectively represents and/or expresses their problem in a computational environment.

ERS TRADESPACE:

- Create an open software architecture and integrated toolset that promote collaborative design and analysis, interoperability, and extensibility
- Create higher-level DSL abstractions for exploration of SE questions in an executable environment, quantitatively and traceably to support higher level decision making goals

Part of a Georgia Tech Research Institute (GTRI) and U.S. Army Engineer Research and Development Center (ERDC) collaborative effort

The "Big Picture" Process

Operational Requirements — Stakeholder expectation statements

MOEs — "Operational" measures of success related to the achievement of the mission or operational objective

KPPs — A critical subset of the performance parameters representing the most critical capabilities and characteristics

Architecture — High-level architecture definition for system designs

M&S — To refine and identify individual system performance needs

Performance

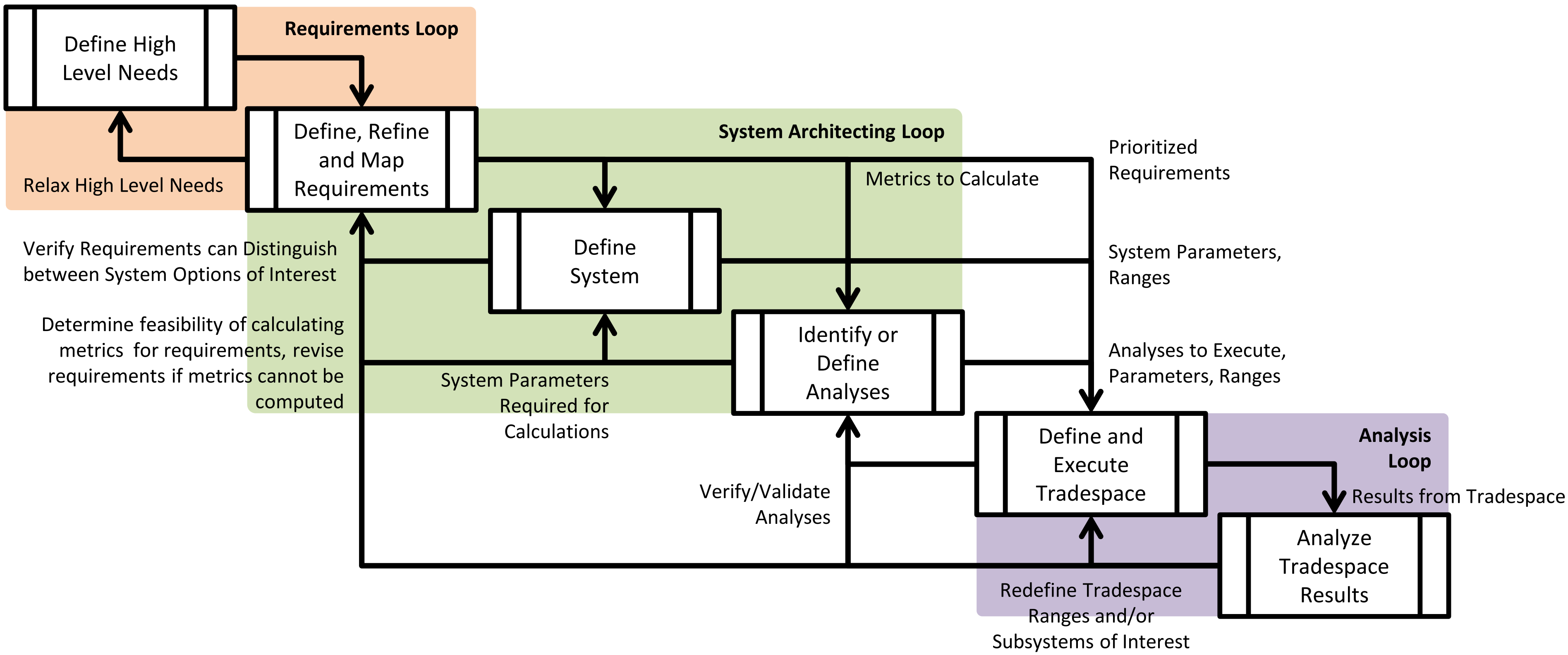
MOPs — Measures that characterize physical or functional attributes relating to the system operation

M&S — To refine and identify what attributes are needed to characterize a system design and evaluate its MOEs, MOPs, KPPs, etc.

System Design Variables — Attributes expressed as value properties that describe specific system design alternatives.

Networked Workflow through a Design Space Environment

Generalized Systems Engineering Workflow showing the set of all Systems Engineering Use Cases



A use case has a specific path through the networked workflow. Driving the tool development with the generalized workflow helps ensure we can meet the requirements of *future* use cases.

ERS TRADESPACE: Process Steps

- **Define**

  Users describe the needs, the analyses to assess whether or not the needs are met, and the system(s) being designed to satisfy those needs

- **Execute**

  Users set conditions for and manage/monitor the execution of the integrated engineering models

- **Analyze**

  Users assess the information generated by the execution of the models to improve their mental models of the problem and the system of interest

MBSE
Model Based Systems Engineering

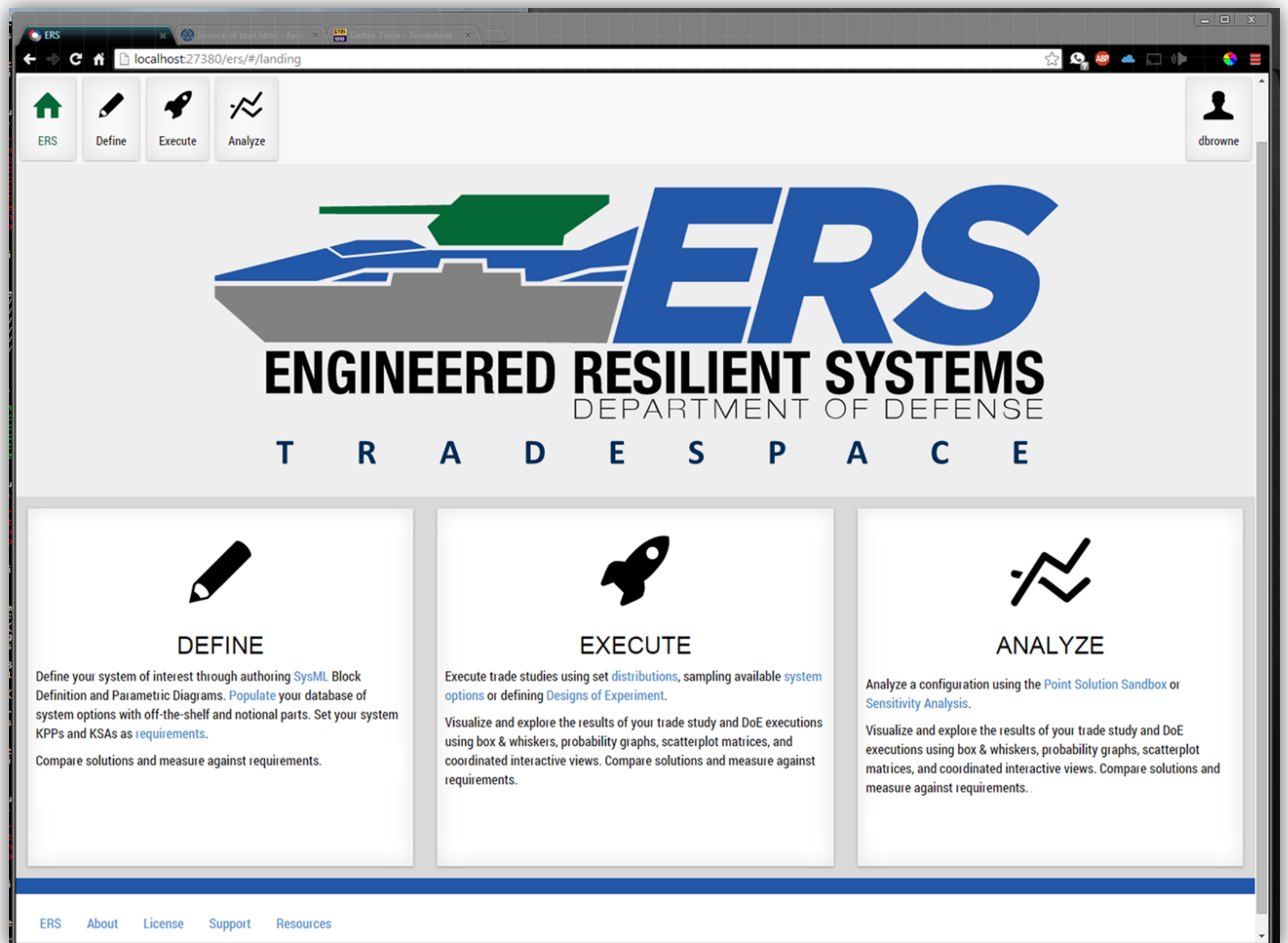
MDAO
Multidisciplinary Design Analysis and Optimization

MCDM
Multi Criteria Decision Making

Open framework allows Systems Engineers to insert themselves at various stages in the design process –

Interacting directly with system definition, models, Tradespace generation, and analytical exploration of the Design Space.

Web-enabled Collaborative Tradestudies



Developed by ERDC and GTRI as part of a larger ERS software program development effort

Software Architecture

- **Front End**

  Composed from a collection of Angular Modules and Javascript libraries

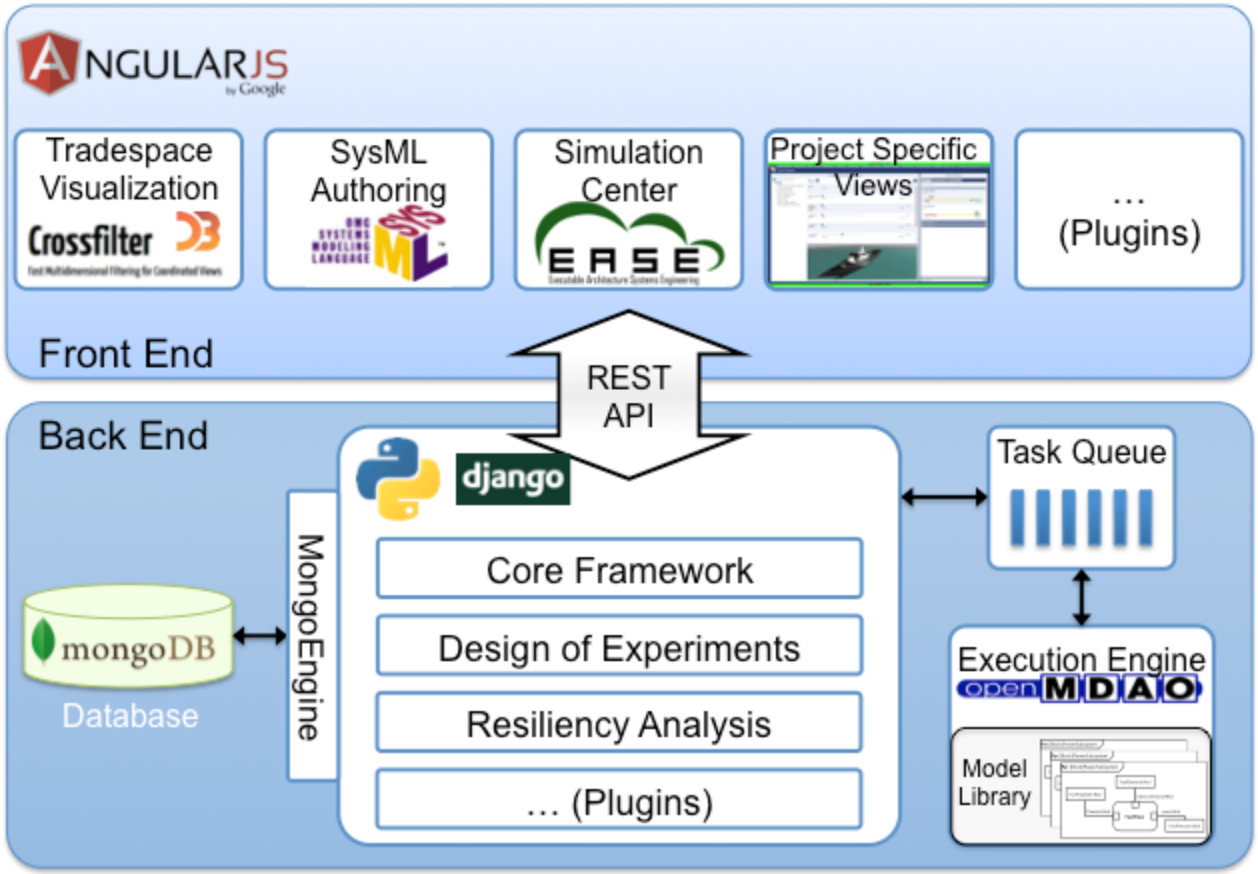
  (Workflow views, dependencies)

- **Backend**

  Composed from Django apps and other python libraries

  Using OpenMDAO to orchestrate the execution of linked constraints

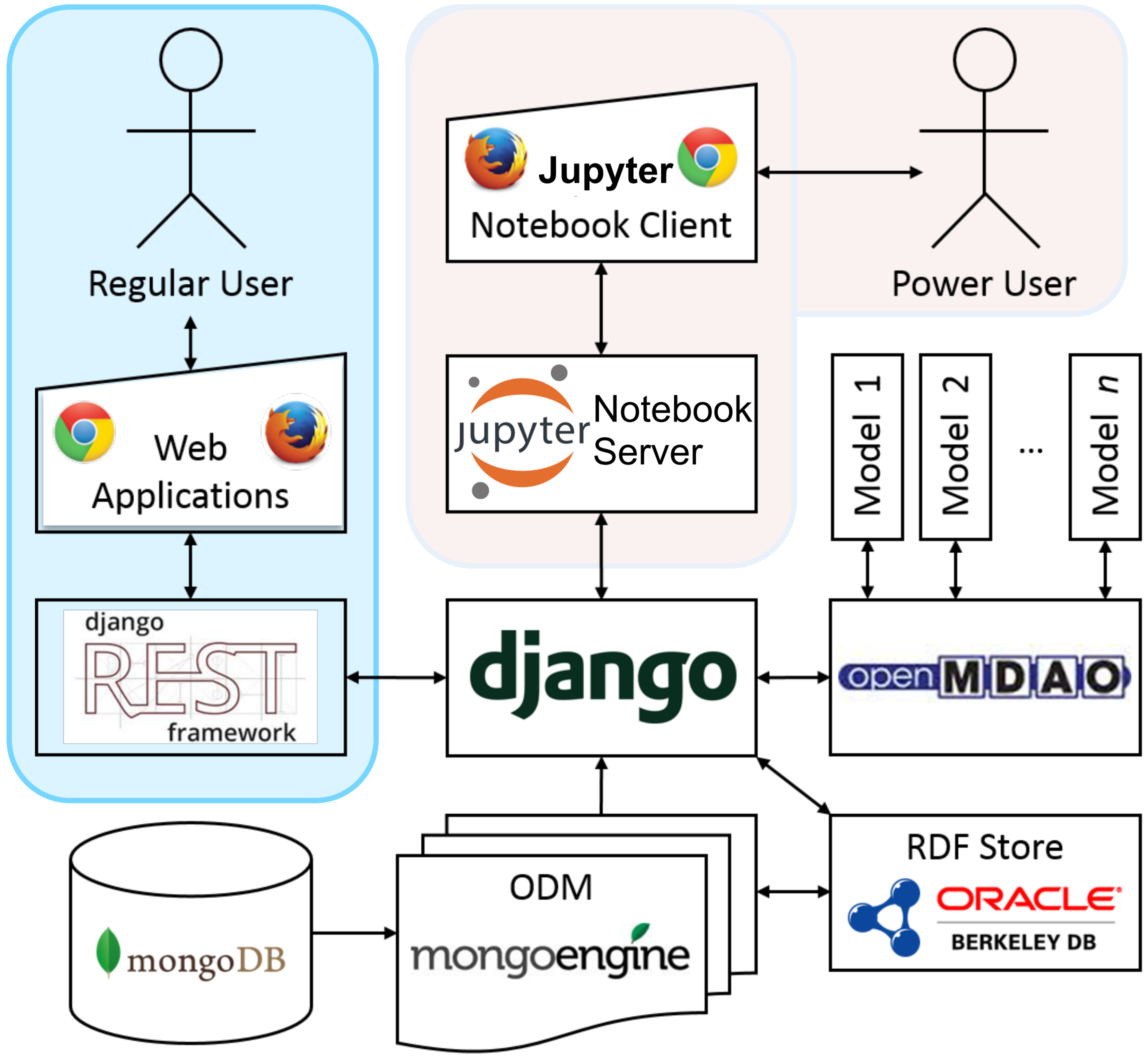
  (Data structure, pipelining, orchestration, & execution)



Modular approach to progressively layer in analysis capabilities and help to make code testable by focusing modules on a particular task

Support Two Disparate Users

- **Power Users**
  - Are experts that frequently do not need a GUI to quickly build a model
  - Can interact directly with the data through scripting environment (Jupyter notebooks)
- **Regular Users**
  - Are typically consuming views of the data (e.g., SysML diagrams, tradespace analysis visualizations)
  - Can interact via views customized for the type of actions needed



Development of a Tradespace Analysis

System Design
(Design Variables)

Value Properties that characterize various design alternatives.

Intrinsic to the system.

Operational Scenario
(Input Parameters)

Environment
&
Use

Parameters defining:

i. Physical environment factors (Operational Environment)
and
ii. How a system will be used (CONOPS)

that will effect performance characteristics (i.e., needed to evaluate MOPs.)

Transfer Functions
(Model Constraints)

Engineering or Data Models
&
Operational Simulations

Transfer functions that evaluate MOEs and MOPs from:

- System Design value properties,
- Operational Scenario parameters,
- Outputs from various other transfer function blocks or simulations.

Evaluation Measures
(Output Parameters)

MOEs,
MOPs,
KPPs,
etc.

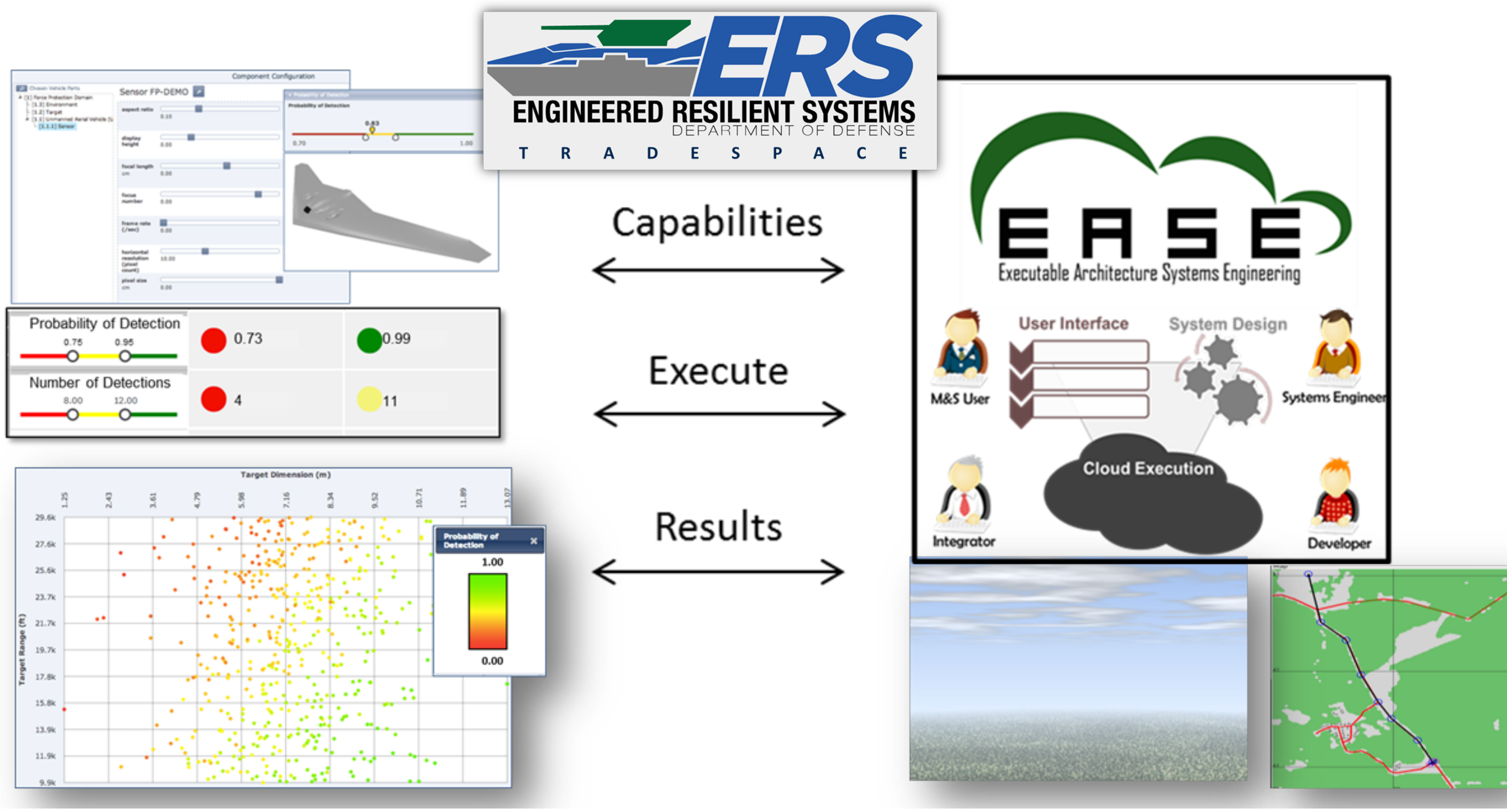
Various measures or parameters that together define the capabilities of the system and/or its effectiveness from the perspectives of the different stakeholders.

NOTE: The relationship between System Design variables and Performance Measures may be reversed depending on the nature of the analysis and transfer functions.

MOE –Measure of Effectiveness, MOP – Measure of Performance, KPP – Key Performance Parameter

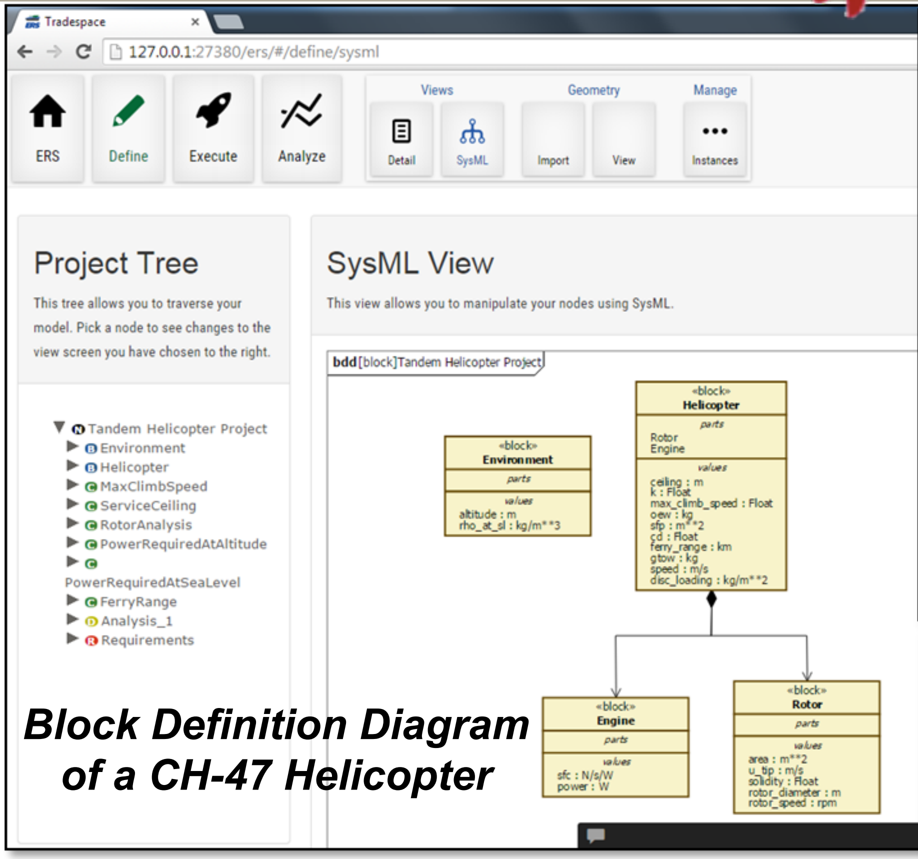
Interfacing Tradestudies with Simulation Operational Scenarios

- **Executable Architecture Systems Engineering (EASE)**
  - Links analytical, experimental, and training objectives with M&S
  - Explore **operational aspects** of the analytical questions in simulation
- ERS effort develops interface between MBSE/ Tradestudies and Army Research Lab investment in executable, cloud-computing resources

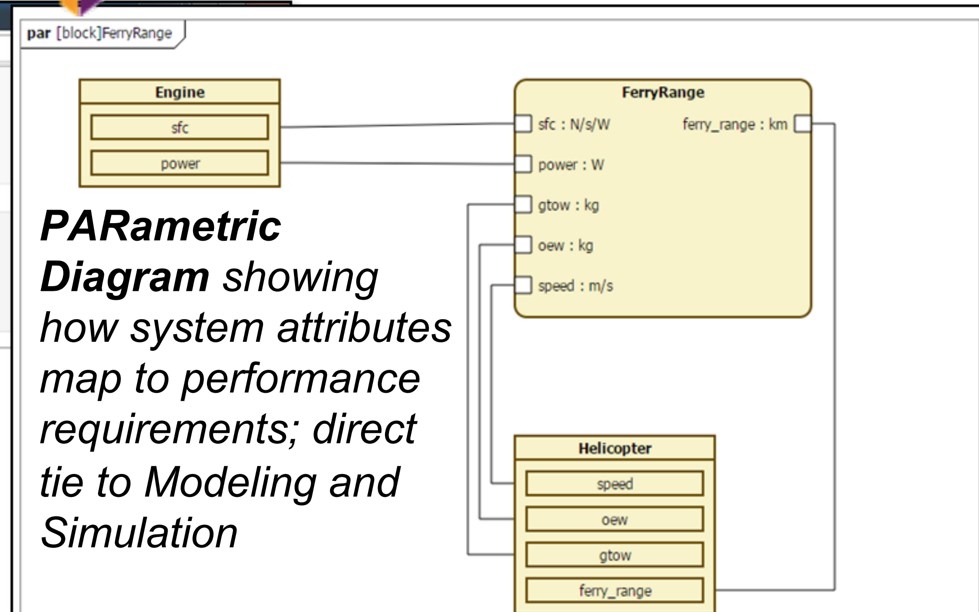


UNCLASSIFIED

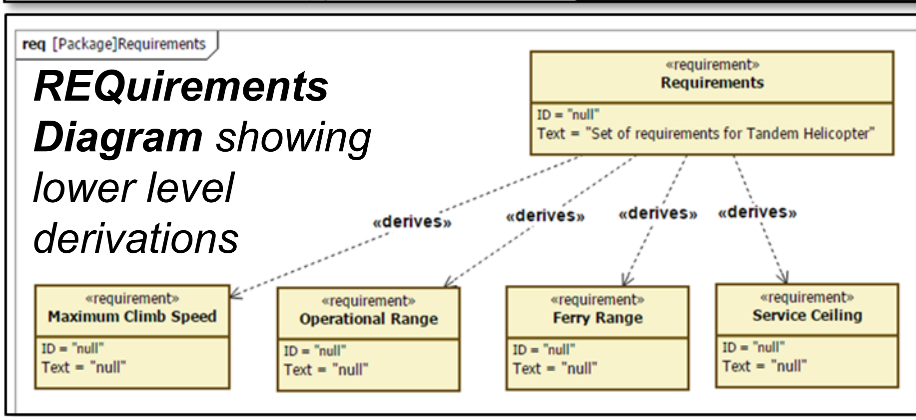
Collaborative SysML in the Browser



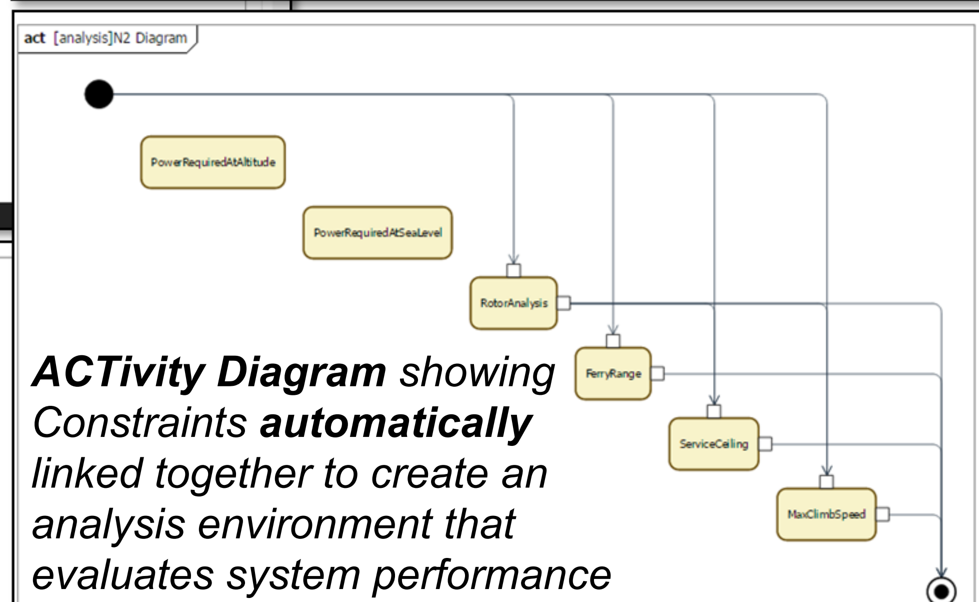
Block Definition Diagram of a CH-47 Helicopter



PARametric Diagram *showing how system attributes map to performance requirements; direct tie to Modeling and Simulation*



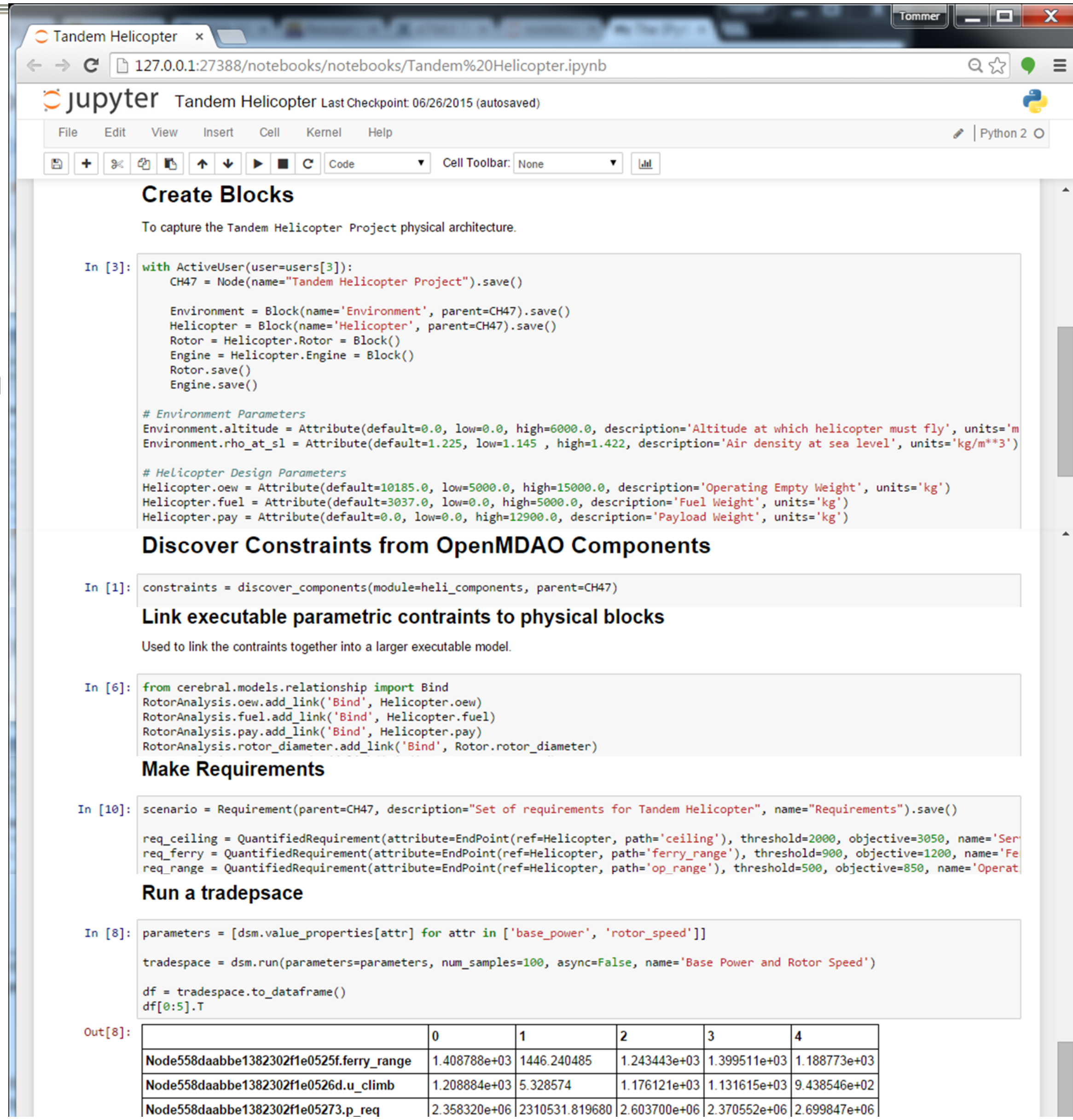
REQuirements Diagram *showing lower level derivations*



ACTivity Diagram *showing Constraints* ***automatically*** *linked together to create an analysis environment that evaluates system performance*

For the "Power User": Jupyter Notebook Interface*

- **Tandem helicopter** example shows how SysML blocks are created
- A block makes declarative statements about a system or component within a system) or attribute.
- Blocks are built off abstractions from a 3G language to build more of a 4G Domain Specific Language – how a systems engineer thinks.
- Tradespace can be executed directly



Tandem Helicopter ×

127.0.0.1:27388/notebooks/notebooks/Tandem%20Helicopter.ipynb

jupyter Tandem Helicopter Last Checkpoint: 06/26/2015 (autosaved)

File Edit View Insert Cell Kernel Help | Python 2

Code | Cell Toolbar: None

Create Blocks

To capture the `Tandem Helicopter Project` physical architecture.

In [3]:

```
with ActiveUser(user=users[3]):
    CH47 = Node(name="Tandem Helicopter Project").save()

    Environment = Block(name='Environment', parent=CH47).save()
    Helicopter = Block(name='Helicopter', parent=CH47).save()
    Rotor = Helicopter.Rotor = Block()
    Engine = Helicopter.Engine = Block()
    Rotor.save()
    Engine.save()

# Environment Parameters
Environment.altitude = Attribute(default=0.0, low=0.0, high=6000.0, description='Altitude at which helicopter must fly', units='m
Environment.rho_at_sl = Attribute(default=1.225, low=1.145 , high=1.422, description='Air density at sea level', units='kg/m**3')

# Helicopter Design Parameters
Helicopter.oew = Attribute(default=10185.0, low=5000.0, high=15000.0, description='Operating Empty Weight', units='kg')
Helicopter.fuel = Attribute(default=3037.0, low=0.0, high=5000.0, description='Fuel Weight', units='kg')
Helicopter.pay = Attribute(default=0.0, low=0.0, high=12900.0, description='Payload Weight', units='kg')
```

Discover Constraints from OpenMDAO Components

In [1]:

```
constraints = discover_components(module=heli_components, parent=CH47)
```

Link executable parametric contraints to physical blocks

Used to link the contraints together into a larger executable model.

In [6]:

```
from cerebral.models.relationship import Bind
RotorAnalysis.oew.add_link('Bind', Helicopter.oew)
RotorAnalysis.fuel.add_link('Bind', Helicopter.fuel)
RotorAnalysis.pay.add_link('Bind', Helicopter.pay)
RotorAnalysis.rotor_diameter.add_link('Bind', Rotor.rotor_diameter)
```

Make Requirements

In [10]:

```
scenario = Requirement(parent=CH47, description="Set of requirements for Tandem Helicopter", name="Requirements").save()

req_ceiling = QuantifiedRequirement(attribute=EndPoint(ref=Helicopter, path='ceiling'), threshold=2000, objective=3050, name='Ser
req_ferry = QuantifiedRequirement(attribute=EndPoint(ref=Helicopter, path='ferry_range'), threshold=900, objective=1200, name='Fe
req_range = QuantifiedRequirement(attribute=EndPoint(ref=Helicopter, path='op_range'), threshold=500, objective=850, name='Operat
```

Run a tradepsace

In [8]:

```
parameters = [dsm.value_properties[attr] for attr in ['base_power', 'rotor_speed']]

tradespace = dsm.run(parameters=parameters, num_samples=100, async=False, name='Base Power and Rotor Speed')

df = tradespace.to_dataframe()
df[0:5].T
```

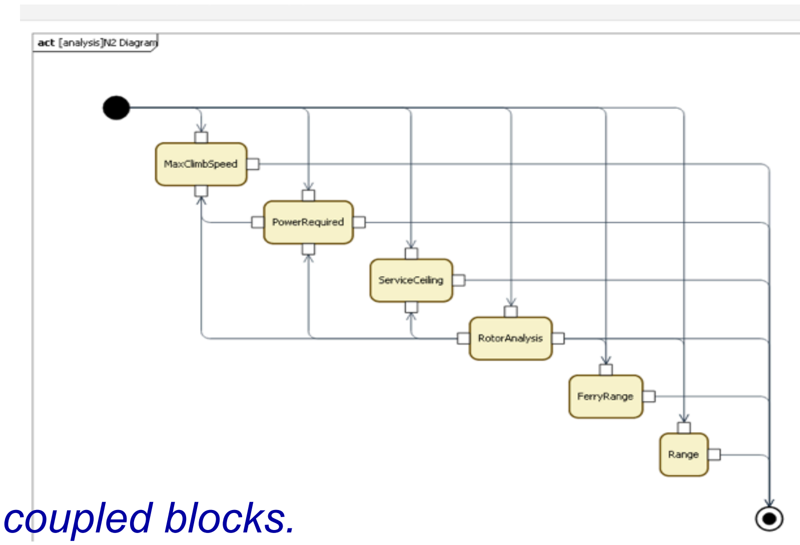
Out[8]:

	0	1	2	3	4
Node558daabbe1382302f1e0525f.ferry_range	1.408788e+03	1446.240485	1.243443e+03	1.399511e+03	1.188773e+03
Node558daabbe1382302f1e0526d.u_climb	1.208884e+03	5.328574	1.176121e+03	1.131615e+03	9.438546e+02
Node558daabbe1382302f1e05273.p_req	2.358320e+06	2310531.819680	2.603700e+06	2.370552e+06	2.699847e+06

*formerly iPython Notebook

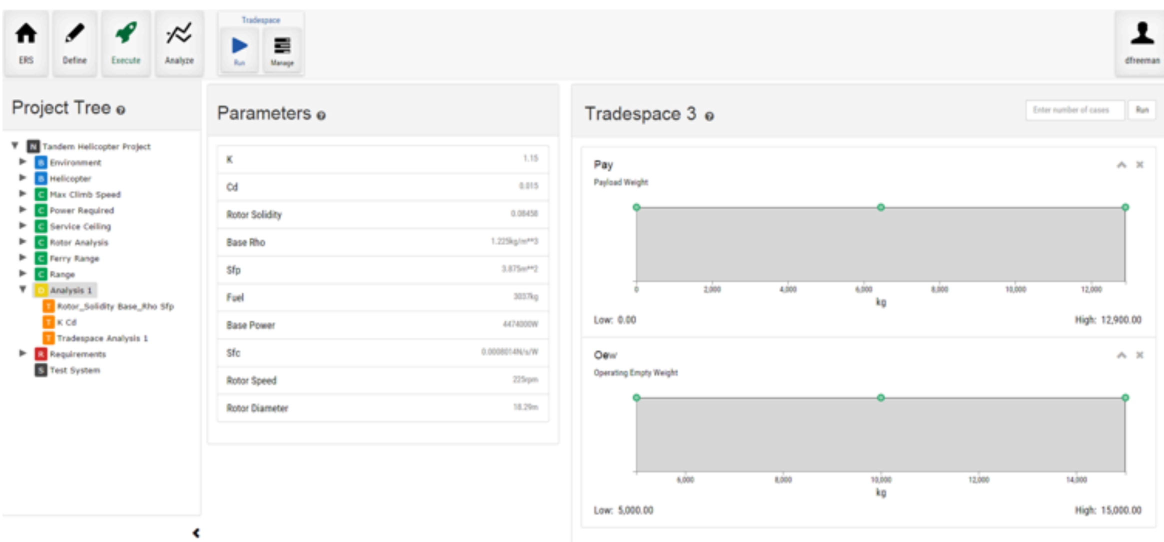
Tradespace Generation

M&S architected in and of itself to support scalable and efficient Tradespace generation



Orchestration and execution via openMDAO

Supports modularity while still solving as tightly coupled blocks.



A Tradespace can be generated by varying independent parameters in the analysis.

Generating a Tradespace needs to become smarter.

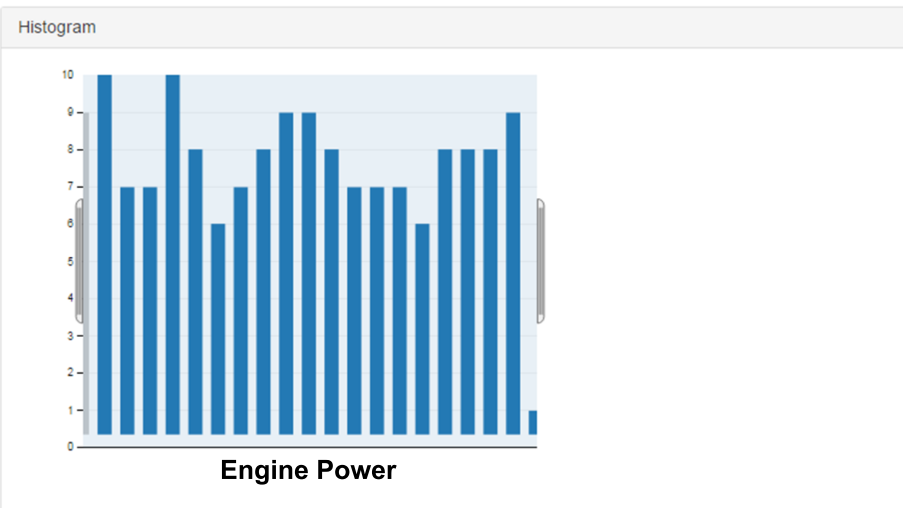
Maturing toward more efficiently using resources in generation process.

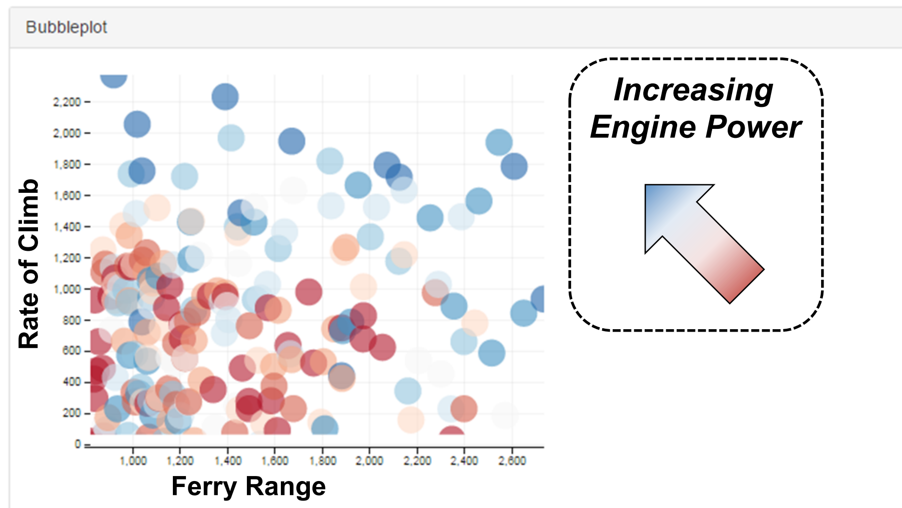
Interactive and Dynamic Analysis

Dynamic Tradespace exploration needs to be **intelligently driven** by how design space data maps to the systems model(s), design variables, constraints, requirements, stakeholder perspectives, etc.

Analyses supported by **interactive visualization can be constructed** – tailored by an analyst – to explore design space regions and dimensions of resiliency of interest.

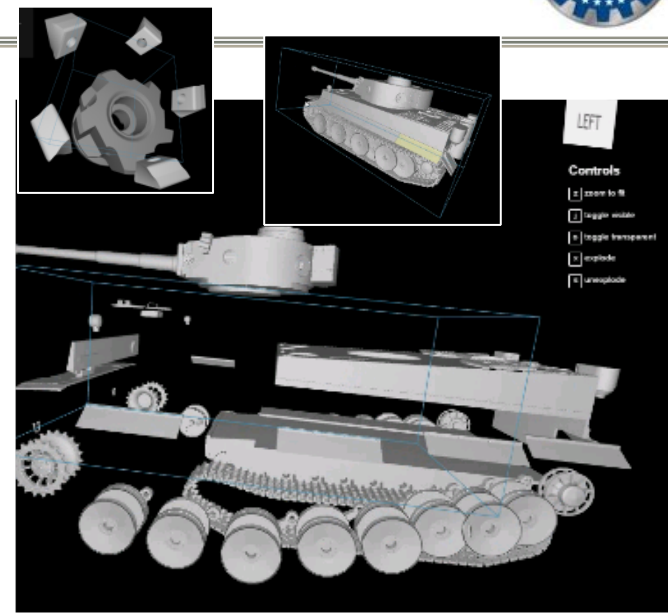
Dynamic analyses driven by interactive capability to **specify and infer relationships across whole, integrated body of metadata**.





- Able to select regions of interest within a tradespace
- Note that as ***Engine Power*** is increased, a tradeoff emerges between ***Rate of Climb*** and ***Ferry Range***

Next Steps

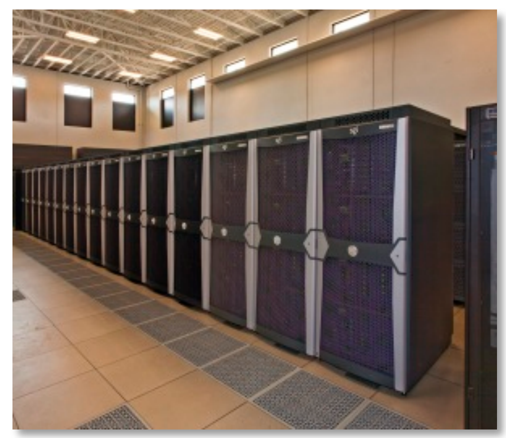


- Extend a limited "**CAD in the browser**" capability
- Integration with **High Performance Computing** assets at DoD HPC Centers

> Beyond Big Data analysis – How do we architect to exploit HPC resources?

- Application to **DoD acquisition programs**
- **Tradespace exploration** has become a **living narrative**

> Analyses and their results must be captured in the context of the foundational assumptions and exploration to be understood and effectively conveyed.



http://itl.erdc.usace.army.mil/featurecenter

Parting Thoughts…

- **Systems Engineering for tradestudies requires an evolution of how components, processes, and design space exploration symbiotically support each other.**
- **Tradepsace generation and analyses are realized through a holistic integration of MBSE, MDAO, and MCDM –**

  All of which are supported by interactive modeling, execution, and exploration.
- **Tradespace exploration for ERS is grounded on Big Data and information analyzed and presented in a holistic view, all while preserving the emergent and dynamic narrative.**
- **Critical program decisions are often made based on the outcomes of trades defined by multiple types and quantities of data and information.**
- **Consistency across treatments and processes is vital to produce meaningful (actionable) insights.**
- **Methods, Processes, & Tools (MPTs) should be engineered together to promote transparent, intuitive, rational, and quantifiably traceable foundations for resiliency analyses.**
- **Preserving modularity and orchestration of analytical components helps support an integrated frontend-to-backend architecture, and …**

  Enables interactive visualization for analytical construction and tradespace exploration.

Parting Thoughts…

The overarching goal is to support DoD Leadership in keeping with the vision for ERS:

- Through more effective evaluation, definition, and maturation of systems architectures and requirements alongside diverse stakeholders needs and expectations (present and future)
- Enabling operationally relevant analyses of Resiliency that better support the Acquisitions process across a system's lifecycle.

Acknowledgements

The authors wish to thank the following for their valuable contributions to the content of this work

Georgia Tech Research Institute

- **Erika Brimhall**
- **Daniel Browne**
- **Robert Kempf**
- **L. Drew Pihera**
- **Melissa Rost**

ERDC Engineer Research & Development Center

- **Dr. Owen Eslinger**
- **Alex Baylot**
- **Daniel Chausse**
- **David Stuart**
- **Dr. Andrew Strelzoff**

Portions of this material are based upon work supported, in whole or in part, by the United States DoD through the Systems Engineering Research Center (SERC) under Contract HQ0034-13-D-0004. SERC is a federally funded University Affiliated Research Center managed by Stevens Institute of Technology. The views and conclusions are those of the individual authors and participants, and should not be interpreted as necessarily representing official policies, either expressed or implied, of the DoD, any specific US Government agency, or the US Government in general.