A Systems Engineering Approach to Requirements Analysis

By
Prof. Dimitri Mavris

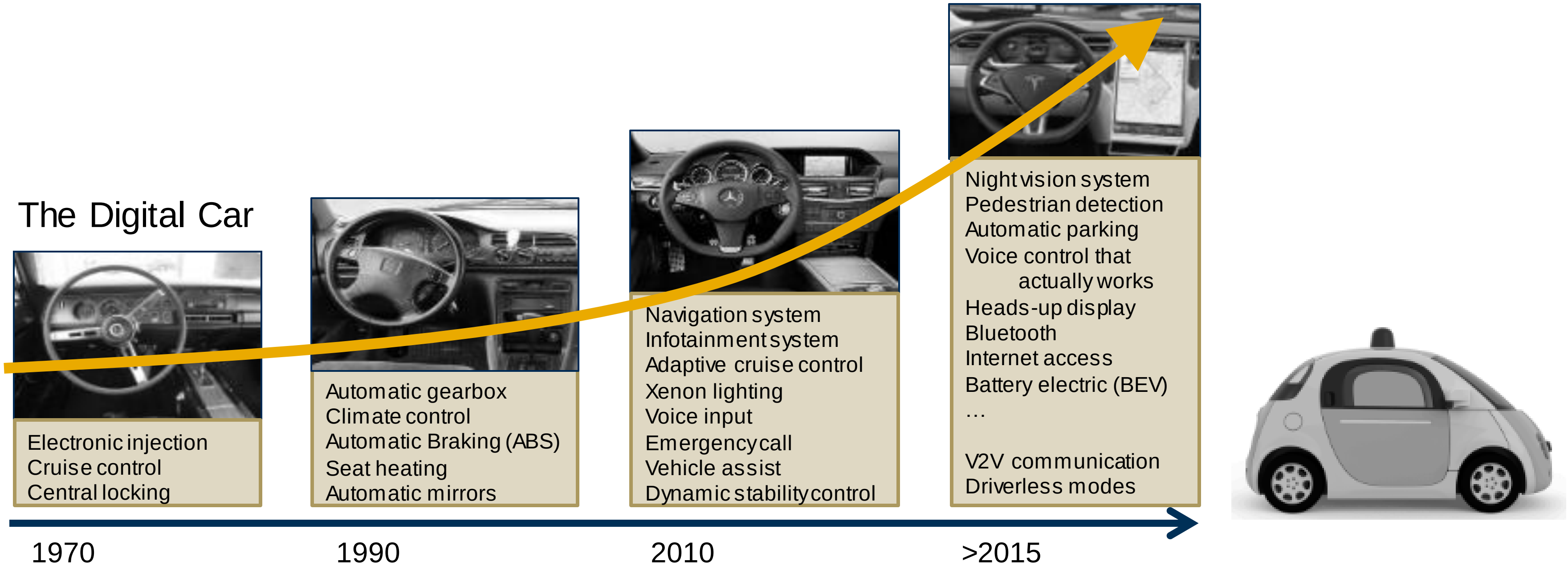
Director

Aerospace Systems Design Laboratory

Boeing Regents Professor in Advanced Aerospace Systems Analysis
NIA NASA Langley Distinguished Regents Professor
School of Aerospace Engineering
Georgia Institute of Technology

Engineered Complex Systems

- Complexity is often a By-Product of our Desire for Functionality
- **Increasingly Rapid Expansion of Functionality**

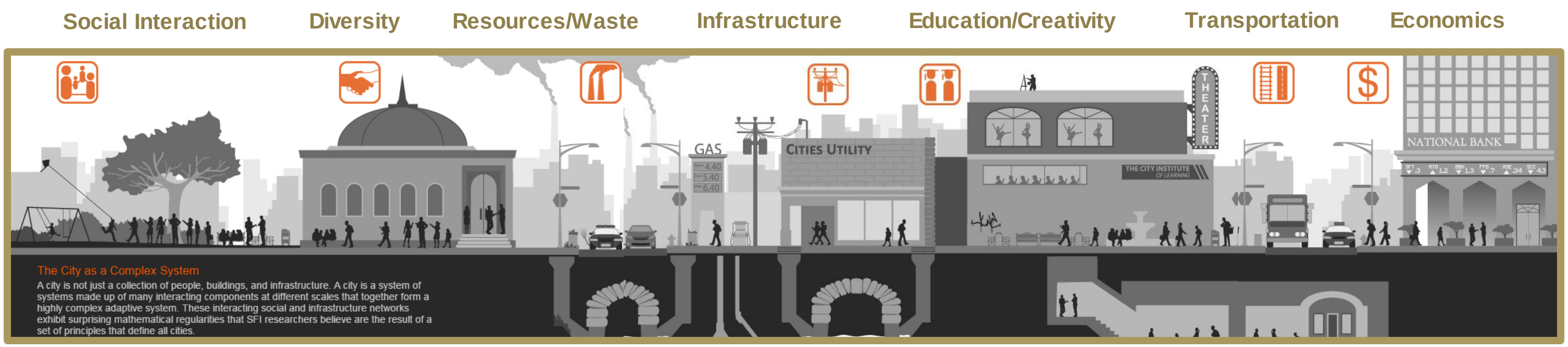


Dr. Dimitri N. Mavris, Director ASDL
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150
dimitri.mavris@ae.gatech.edu

Gigatechnologies – 10^{+9} Systems

(i.e. how would you model and design the largest engineered systems that humans create?)

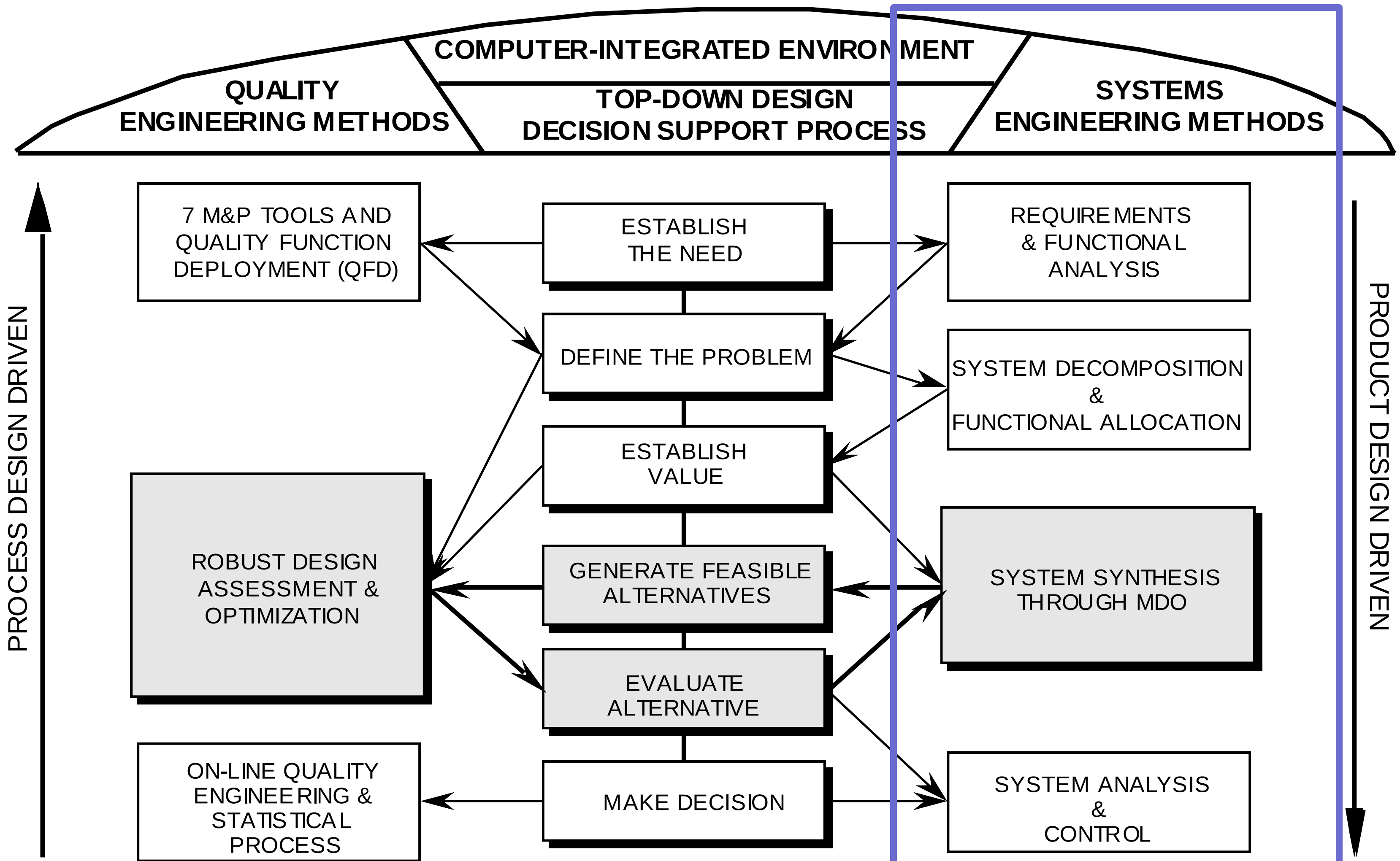
- There is no single perspective or strategy that can change the shape of a city, only shared knowledge and collaboration combined with foresight, shared long-term goals, coordinated short term actions, and continuous measurement and feedback over many eras of change



- "*Make no little plans. They have no magic to stir men's blood and probably themselves will not be realized. Make big plans; aim high in hope and work, remembering that a noble, logical diagram once recorded will never die, but long after we are gone will be a living thing, asserting itself with ever-growing insistency*." - Daniel Burnham, City Planner, 1903

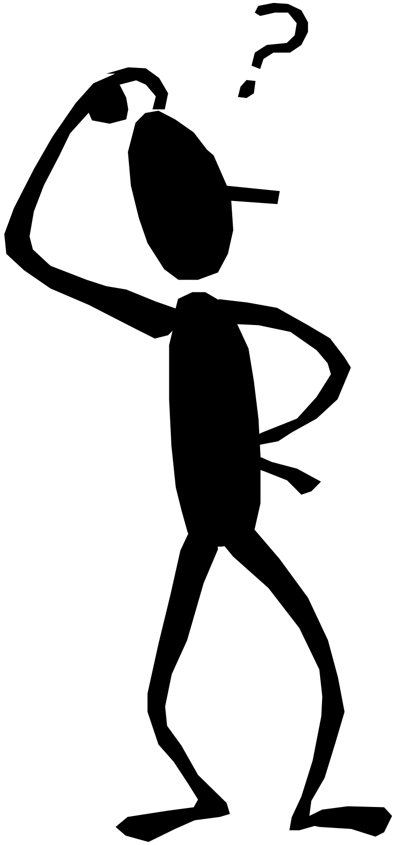
Dr. Dimitri N. Mavris, Director ASDL
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150
dimitri.mavris@ae.gatech.edu

A Basis from Which to Begin: Systems Engineering Methods



Dr. Dimitri N. Mavris, Director ASDL
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150
dimitri.mavris@ae.gatech.edu

How Would You Define SE?



Dr. Dimitri N. Mavris, Director ASDL
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150
dimitri.mavris@ae.gatech.edu

Definitions: Systems Engineering

INCOSE (International Council on Systems Engineering)

- Systems Engineering is an interdisciplinary approach and means to enable the realization of successful systems
- It focuses on defining customer needs and required functionality early in the development cycle, documenting requirements, then proceeding with design synthesis and system validation while considering the complete problem: Operations, Cost & Schedule, Performance, Training & Support, Test, Disposal, and Manufacturing
- Systems Engineering integrates all the disciplines and specialty groups into a team effort forming a structured development process that proceeds from concept to production to operation
- Systems Engineering considers both the business and the technical needs of all customers with the goal of providing a quality product that meets the user needs

In short, Systems Engineering is a way of doing business that is focused on providing holistic, integrated solutions to the customer

Systems Engineering Focus

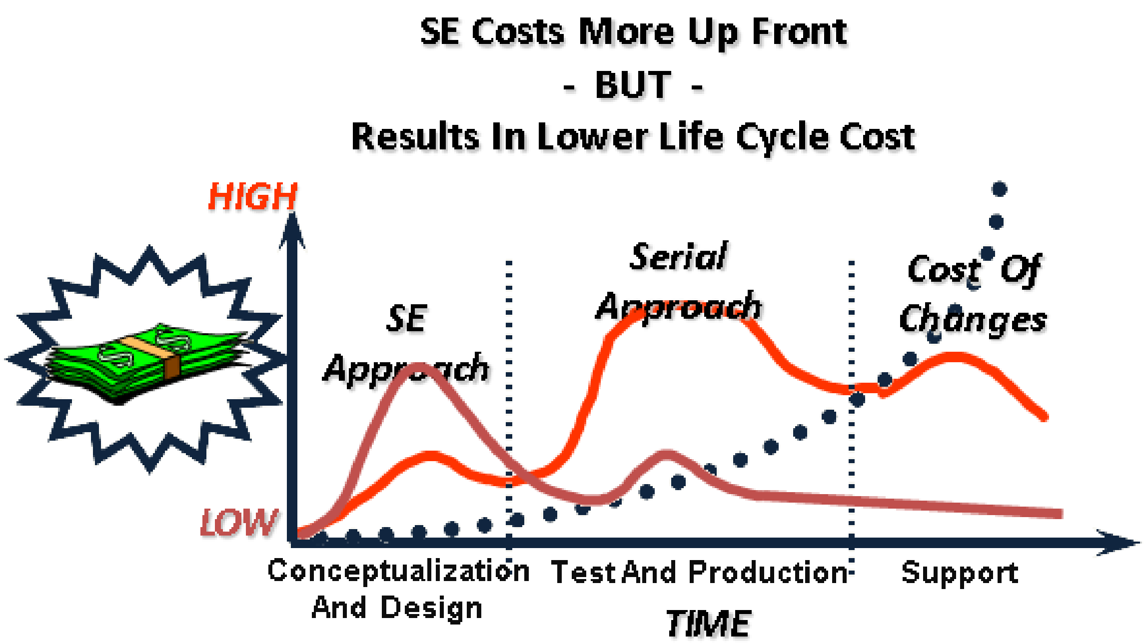
- **Defining customer needs** and required functionality **early** in the development cycle
- Developing and managing requirements and interfaces
- Synthesizing designs and validating system
- Considering the complete problem to be solved, including:
  - Acquisition Approach and Management
  - External environment/influences
  - Stakeholders
  - Requirements
  - Performance
  - Cost and Schedule
  - Technology
  - Manufacturing
  - Test
  - Training and Support
  - Operations and Maintenance
  - Disposal
- Considering both the business and the technical needs of all customers with the goal of providing a quality product that meets the user needs

Why Systems Engineering?

Dr. Dimitri N. Mavris, Director ASDL
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150
dimitri.mavris@ae.gatech.edu

Why use an SE approach?

- The application of systems engineering to the design, development, and support of complex systems has been shown to reduce cost and schedule overruns

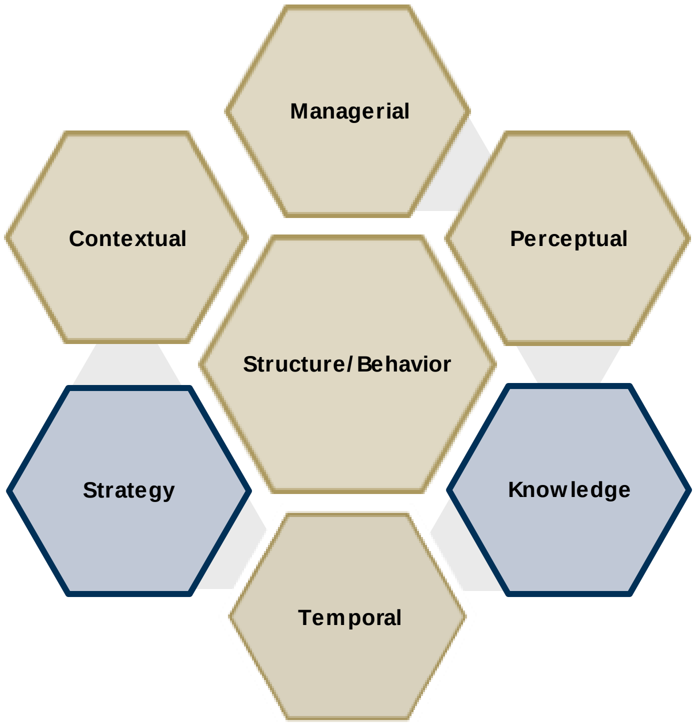


Systems Engineering is an overarching discipline, to achieve the best overall product and/or service that meets requirements and does so within budget and schedule constraints

Dr. Dimitri N. Mavris, Director ASDL
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150
dimitri.mavris@ae.gatech.edu

Definition of a System

- A system is made up of a set of **entities** and **processes** that have a defined structure
- The system has a unique **purpose**, different than those of its individual entities and processes
- The system has a set of **behaviors** determined by its functions, interactions, and activities
- The system has a **boundary**, internal capabilities, and an external environment
- The system exists within an external **context** and a **time**
- The system has a **domain** which aligns the **language** we use to describe the system & its external context
- The system has a form of **managerial control** determined by a set of stakeholder goals
- The usefulness of the system includes various stakeholder **perspectives**
- Systems encode **knowledge** and **strategy**



Dr. Dimitri N. Mavris, Director ASDL
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150
dimitri.mavris@ae.gatech.edu

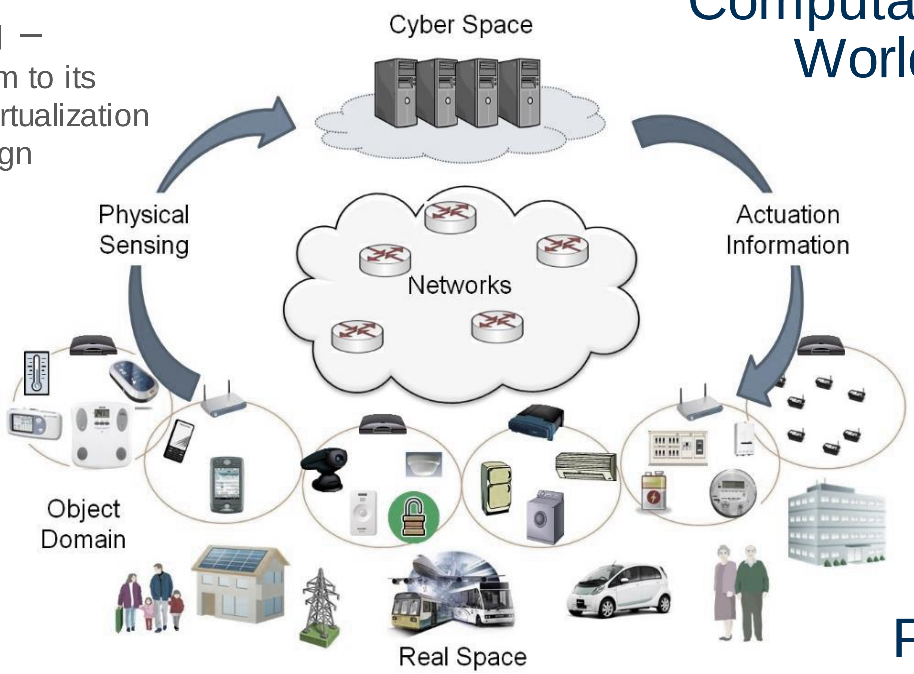
Three Classes of Systems

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

Abstract

Concrete

Conventional Engineering –
Moving from abstract design to physical realization



Computational World

Cyber-Physical World

Physical World

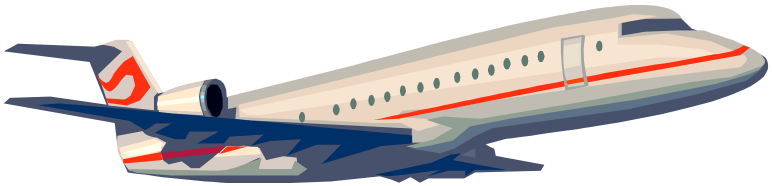
Reference: Wang, Y., Software Engineering Foundations: A Software Science Perspective.
Image: http://www.jaist.ac.jp/is/labs/lim-lab/image/4.CPS.jpg

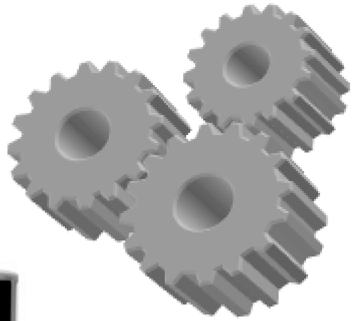
Systems can be ordered in complexity

- **Simple systems**
  - Small number of components which act according to well understood laws
- **Complicated systems**
  - Large number of components which have well-defined roles and are governed by well understood laws
- **Complex systems**
  - Large number of components capable of interacting with each other and the environment
  - Rules for the components may change over time and may not be well understood
  - Properties of the system as a whole cannot necessarily be determined from the sum of the parts

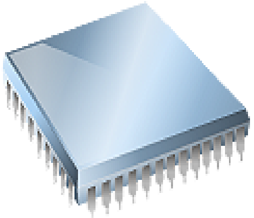
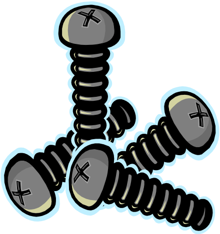
[Amaral and Ottino. "Complex Networks: Augmenting the Framework for the Study of Complex Systems." Eur. Phys. J. B 38, 147-162 (2004)]

Which of these are systems?



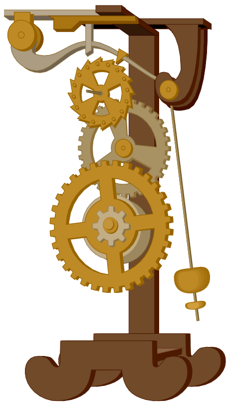





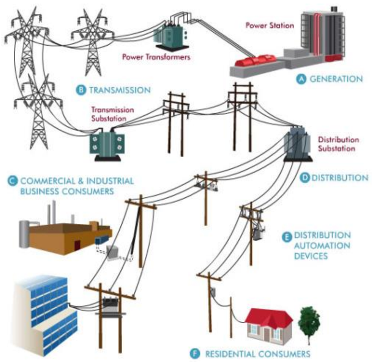




Dr. Dimitri N. Mavris, Director ASDL
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150
dimitri.mavris@ae.gatech.edu

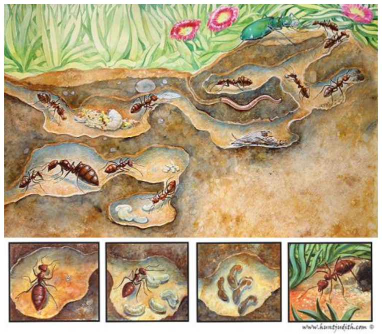
Which are simple, complicated, or complex?



Pendulum

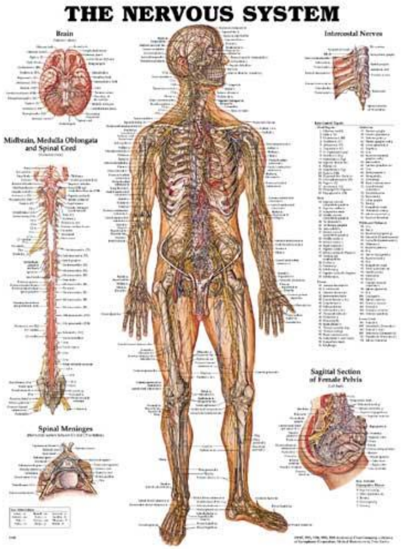


Electricity Grid



Ant Colony

Car



Aircraft

Pen

Dr. Dimitri N. Mavris, Director ASDL
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150
dimitri.mavris@ae.gatech.edu

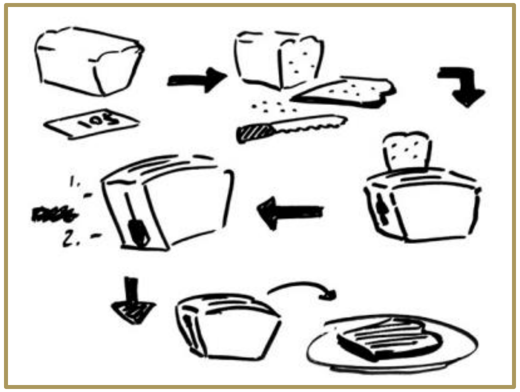
Complex systems have interesting properties

- Large number of variables or elements
- Rich interactions among elements
- Difficulty in identifying attributes and emergent properties
- Loosely organized (structured) interaction among elements
- Probabilistic, as opposed to deterministic, behavior in the system
- System evolution and emergence over time
- Purposeful pursuit of multiple goals by system entities or subsystems (pluralistic)
- Possibility of behavioral influence or intervention in the system
- Largely open to the transport of energy, information, or resources from/to across the system boundary to the environment
- Diverse in technology, context, operation, geography, and conceptual frame

Sample Exercise: In search of the perfect Toast

- Prepare
- Question
- Draw
- Reflect
- Analyze
- Make Meaning

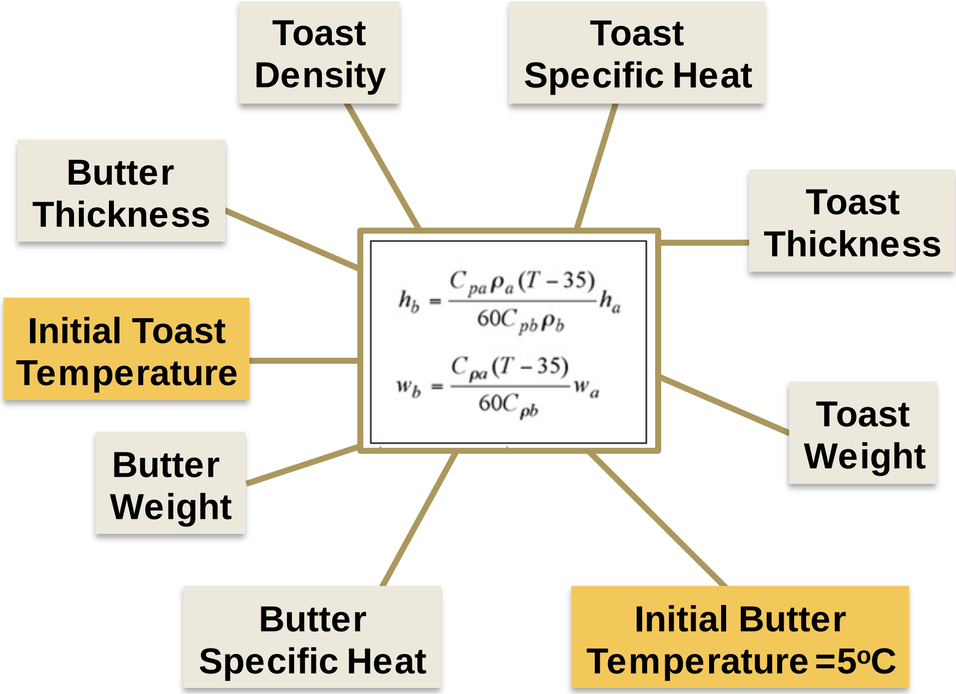
https://www.ted.com/talks/tom_wujec_got_a_wicked_problem_first_tell_me_how_you_make_toast?language=en#t-381429



Source: www.drawtoast.com

Dr. Dimitri N. Mavris, Director ASDL
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150
dimitri.mavris@ae.gatech.edu

The Perfect Toast



Scientists create 'perfect' toast

Food scientists at the UK's University of Leeds have developed a formula for making the perfect piece of toast.

$$h_b = \frac{C_{pa}\rho_a(T-35)}{60C_{pb}\rho_b}h_a$$

$$w_b = \frac{C_{pa}(T-35)}{60C_{pb}}w_a$$

The secret to toast heaven

The equation - which details butter and toast temperature - took three months and cost £10,000 to develop.

Researchers' found that people think the perfect piece of toast should have partly melted butter patches on it, improving its taste and texture.

For this to work, the butter should be applied at fridge temperature of five degrees Celsius, the equation shows.

The formula was developed following research commissioned by the butter brand Lurpak made by Leeds-based Arla Foods.

They wanted to know how the properties of melting butter affects the taste of toast.

Spreading the word

The equation relates to the critical amount of butter, applied at fridge temperature (approximately five degrees Celsius or 41F), required to produce this effect.

H represents thickness, Cp the specific heat, P density, T the initial temperature of the toast, w the weight, and subscript a and b toast and butter respectively.

The Leeds University food scientist Professor Bronek Wedzicha translated the equation into practical terms for toast lovers.

"To produce the patches of butter most people said they preferred, the bread needs to be heated to at least 120°C, and the butter should be used straight from the fridge, applied unevenly within two minutes of the bread coming out of the toaster.

> **The amount of butter should be about one-seventeenth the thickness of the bread**
> Professor Bronek Wedzicha

"The amount of butter should be about one-seventeenth the thickness of the bread," he said.

Source: http://news.bbc.co.uk/2/hi/uk_news/3248822.stm

Dr. Dimitri N. Mavris, Director ASDL
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150
dimitri.mavris@ae.gatech.edu

Problem Formulation

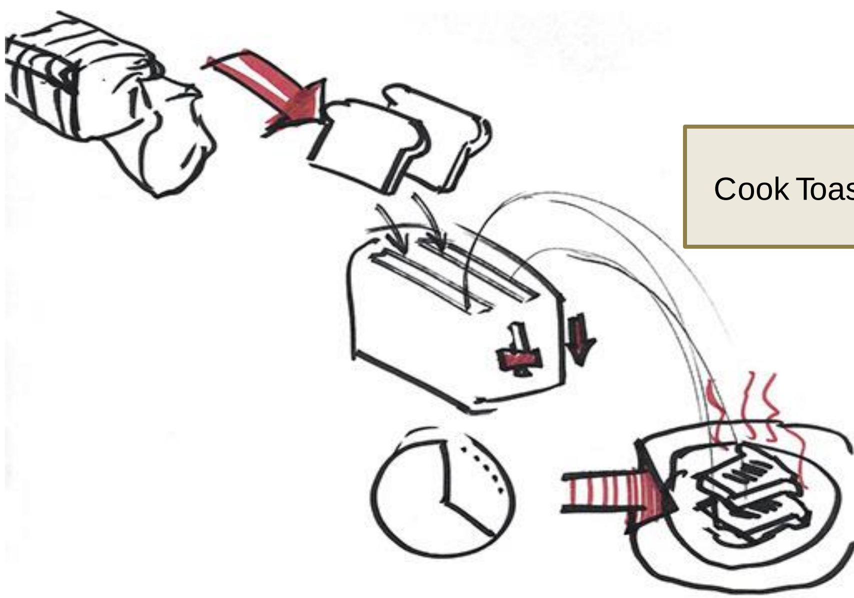
What are the Primary Questions? Assumptions? Constraints?

- How can we automate 154 C and “12:1 exterior/interior crispiness ratio”?
  - What is the ideal toaster temperature?
  - What is the ideal cooking time?
- Is a horizontal slice load or vertical slice load better?
  - What is the horizontal distance/speed for perfect toast?
- How many slices per minute need to be cooked?
- How is the system calibrated for its lifetime?
- What is the optimal balance of automation and price?
- What if “perfect” in one city is not perfect in another?

Dr. Dimitri N. Mavris, Director ASDL
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150
dimitri.mavris@ae.gatech.edu

Visualize: Back of the Napkin Diagram

Insert Toast



Cook Toast

Retrieve
& Butter Toast

Source: www.drawtoast.com

Problem formulation - The Perfect Toast (Literature search)

“It has the look of 'builder's tea' and, crucially, the outside is 12 times crunchier than the middle.

The result is achieved by setting the toaster dial to 'five out of six' on a typical 900-watt appliance to produce a temperature of 154 degrees Celsius, the study revealed.

During his exhaustive study, commissioned by bread maker Vogel's, Dr Lane used a complex formula to help determine the toasting required to produce the perfect level of crispiness.

He found that 216 seconds was the exact amount of time needed to toast the outside to the desired level before the golden 12:1 exterior/interior crispiness ratio was lost.

He also discovered it was best to use a pale, seeded, loaf, taken fresh from the fridge at a temperature of 3 degrees Celsius.

Both sides of the bread should be cooked at the same time, using a toaster rather than a grill, to help 'curtail excessive moisture loss'.

It should then be buttered as soon as it pops up, before the slice loses the heat required to melt the spread.”

The perfect piece of toast: Scientists test 2,000 slices and find 216 seconds is the optimum time

By Daily Mail Reporter
UPDATED: 03:03 EST, 22 July 2011

Share 136 View comments

Scientists today revealed the mathematical formula for a perfect slice of toast, showing that it is best cooked for exactly 216 seconds.

A team of researchers carried out a study which found the optimum thickness is 14mm and the ideal amount of butter is 0.44 grams per square inch.

The recommended cooking time gives the slice a 'golden-brown' colour and the 'ultimate balance of external crunch and internal softness'.

Source: http://www.dailymail.co.uk/sciencetech/article-2017338/The-perfect-piece-toast-Scientists-test-2-000-slices-216-seconds-optimum-time.html

Narrative: Describe the Requirements and the Objectives

Stakeholders

Mission

The **Perfect Toast Academy** is looking for a system that will **reliably produce the perfect piece of toast** for installation in **Just Eggs & Toast** restaurants across the state. Just Eggs & Toast specifies the slice to be a **'golden-brown' color** and the **'ultimate balance of external crunch and internal softness'**.

Requirements

Both sides of the bread should be cooked at the same time, using a toaster rather than a grill, to help 'curtail excessive moisture loss'.

The result is achieved by setting the toaster dial to 'five out of six' on a typical 900-watt appliance to produce a temperature of 154 degrees Celsius.

He found that 216 seconds was the exact amount of time needed to toast the outside to the desired level before the golden **12:1 exterior/interior crispiness ratio** was lost.

It should then be buttered as soon as it pops up, before the slice loses the heat required to melt the spread."

> "The amount of butter should be about one-seventeenth the thickness of the bread"
>
> Professor Bronek Wedzicha

Source: Powell, S. G., Blatt, R. J., *Modeling for Insight: A Master Class for Business Analysts*, John Wiley and Sons, 2008.

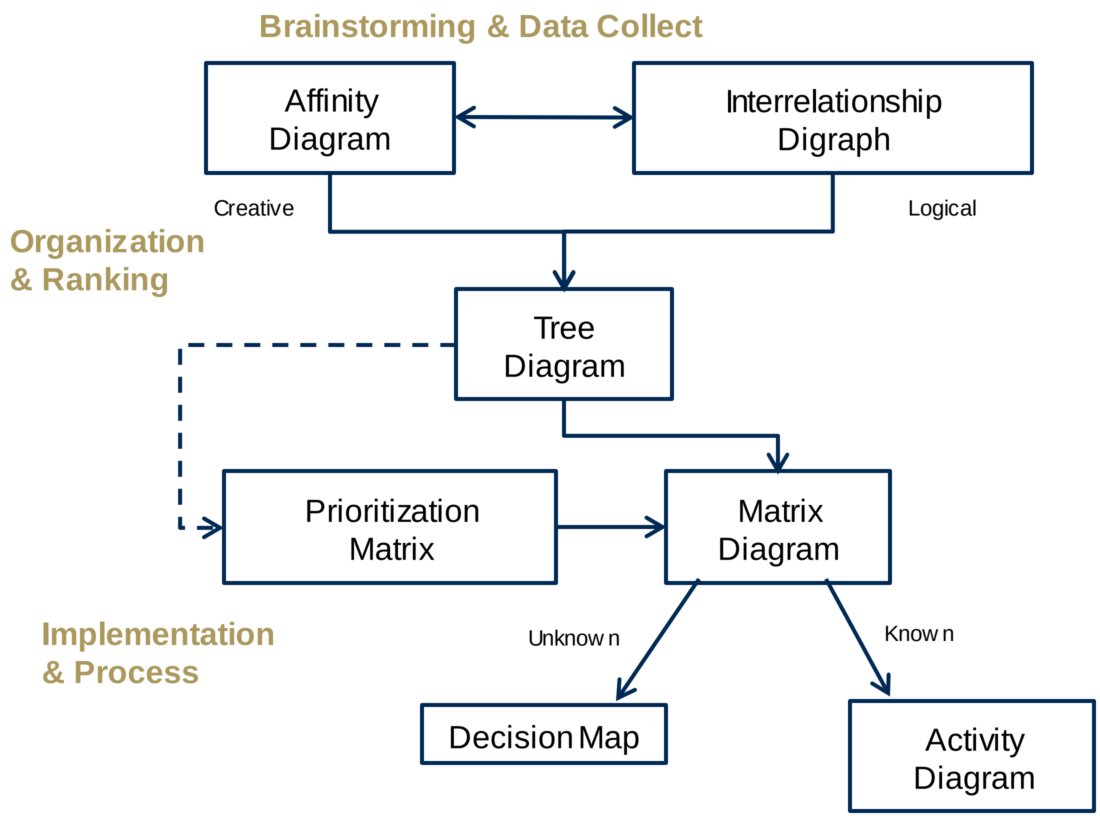
Modeling the System

- Systems thinkers build a model structure as a backbone for their thinking. In other words, they focus on the relationships between variables rather than on the values of those variables. They use the structure they create to ask questions, such as:
  - Have I formulated the right problem?
  - Will the approach I'm taking work out?
  - Will this model help me answer the client's questions?
- Can this approach lead to a practical course of action?

Source: Powell, S. G., Blatt, R. J., *Modeling for Insight: A Master Class for Business Analysts*, John Wiley and Sons, 2008.

Dr. Dimitri N. Mavris, Director ASDL
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150
dimitri.mavris@ae.gatech.edu

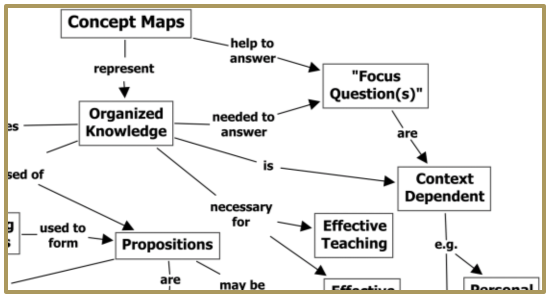
Employ the 7 Management and Planning Tools



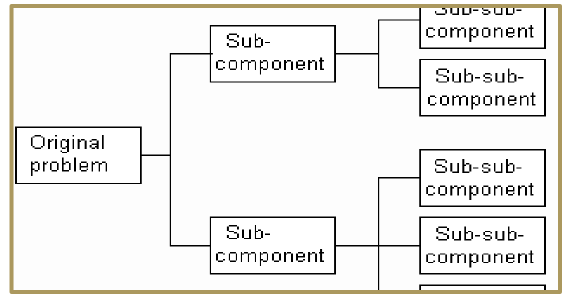
Source: Derived from The Memory Jogger Plus+, Brassard, 1996

Diagram the Problem

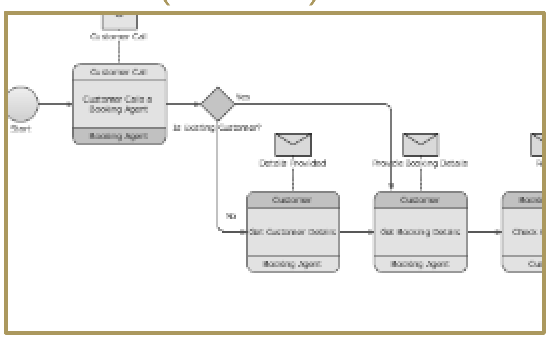
Concept (knowledge)



Tree (structure)



Process (behavior)

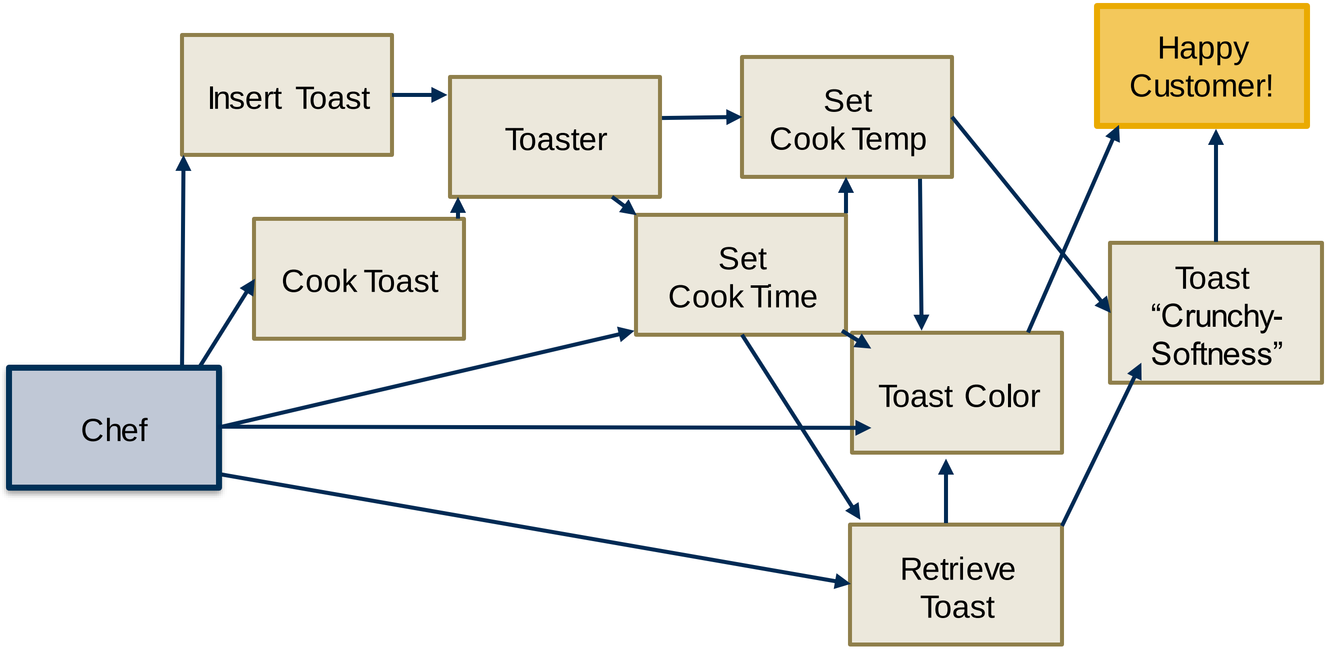


Network (patterns)

Dr. Dimitri N. Mavris, Director ASDL
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150
dimitri.mavris@ae.gatech.edu

Model: Interrelationship Digraph

- Also known as an Acyclic Graph
- Linear model of node & link relationships

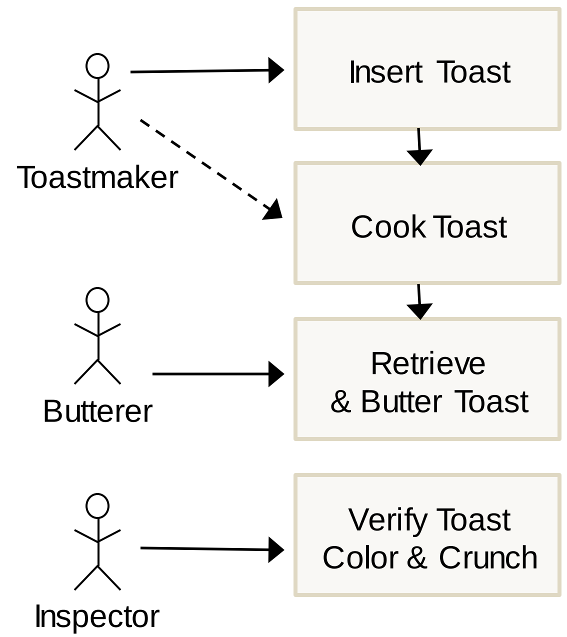


Model: Mission Functions & Use Cases

Functional Decomposition

1. Toastmaker inserts toast
2. Toaster cooks toast to at least 120 ^{o}C and ideally 154^{o}C
3. Toaster cooks toast to “12:1 exterior/interior crispiness ratio”
4. Butteror retrieves toast
5. Inspector verifies golden-brown color to reference photo
6. Inspector verifies balance of external crunch and internal softness with sample tests
7. Butterer butters toast with prepackaged 1/17 thickness pats from refrigerator

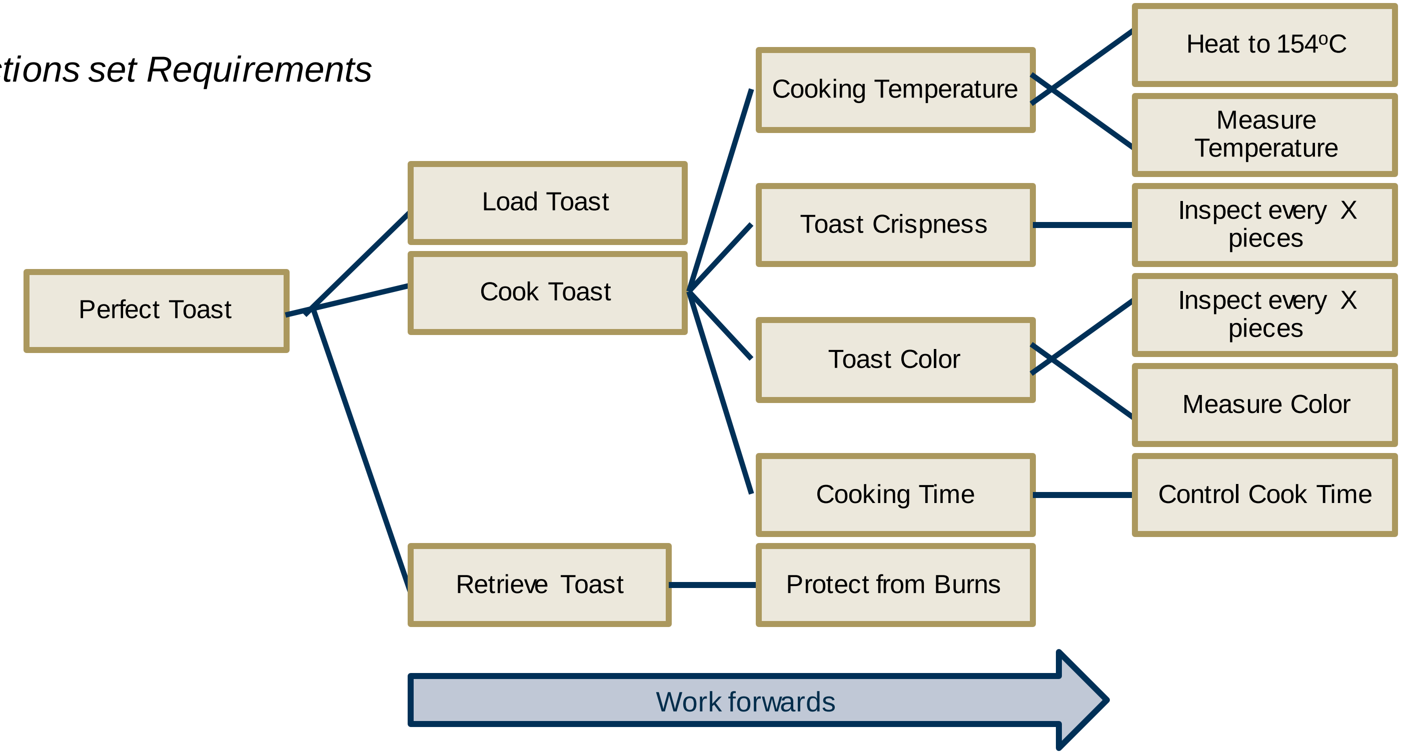
Use Cases



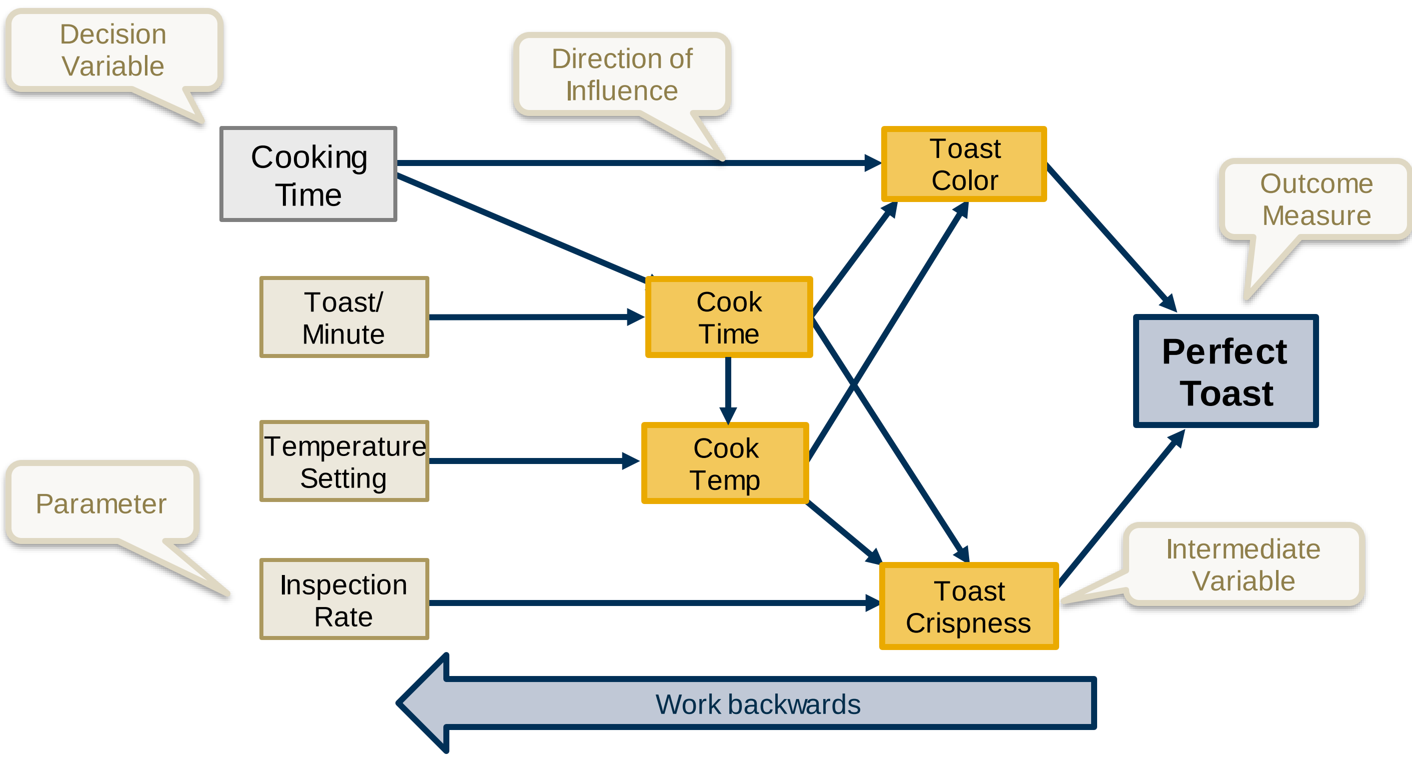
Dr. Dimitri N. Mavris, Director ASDL
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150
dimitri.mavris@ae.gatech.edu

Structure: Hierarchical Decomposition

Functions set Requirements



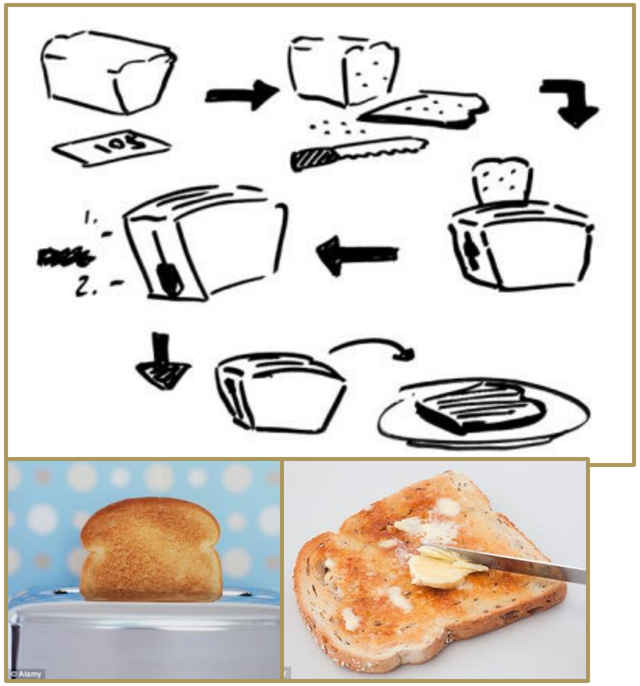
Measurement: Influence Diagram



Dr. Dimitri N. Mavris, Director ASDL
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150
dimitri.mavris@ae.gatech.edu

Summary – Toaster Automation

- Prepare
- Question
- Draw
- Reflect
- Analyze
- Make Meaning

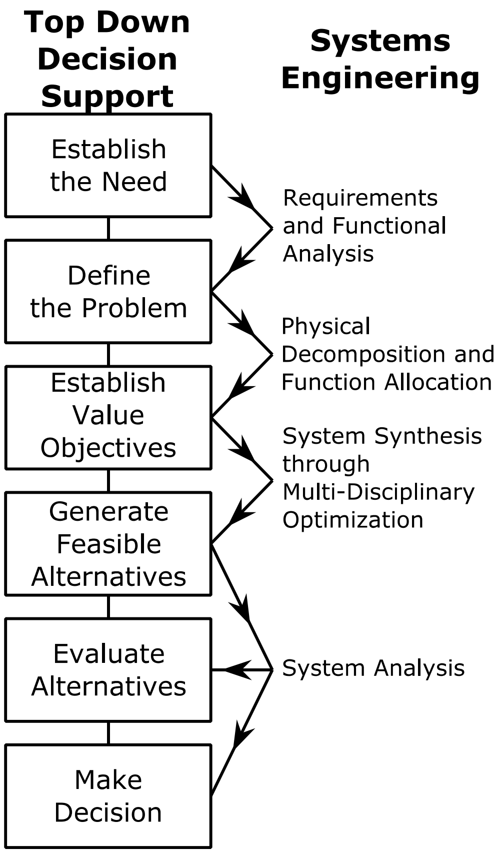


Source: www.drawtoast.com

Dr. Dimitri N. Mavris, Director ASDL
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150
dimitri.mavris@ae.gatech.edu

Systems Engineering Process

Systems Engineering - the design process



- Requirements must be translated into functions
- Those functions must then be allocated to systems, subsystems, components
- All parts must be integrated into alternative feasible & viable designs
- An optimal system must be chosen as the final design.

Systems Engineering Processes

- Systems Engineering Processes: Logical, systematic, comprehensive, iterative problem solving activities tailored and used to accomplish systems engineering tasks and generate work products

Scheduling

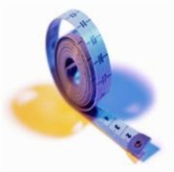
Information Management

System Architecture

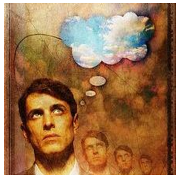
Quality

Risk Management

Requirements Development & Management



Measurement & Analysis



Decision Analysis

Integration, Verification, Validation, & Transition

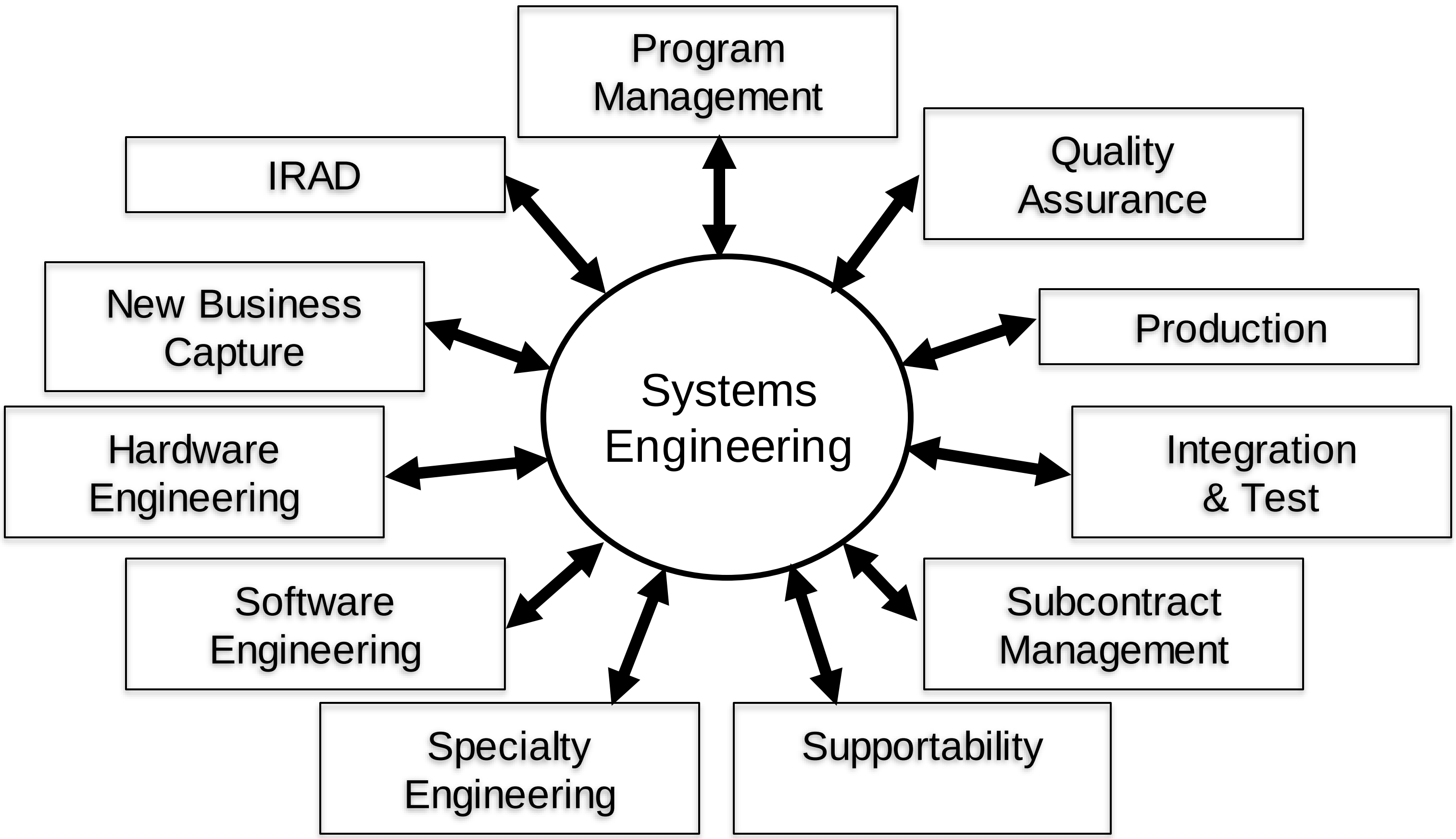
Configuration Management

Cost Estimation

Integrated System Security

Dr. Dimitri N. Mavris, Director ASDL
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150
dimitri.mavris@ae.gatech.edu

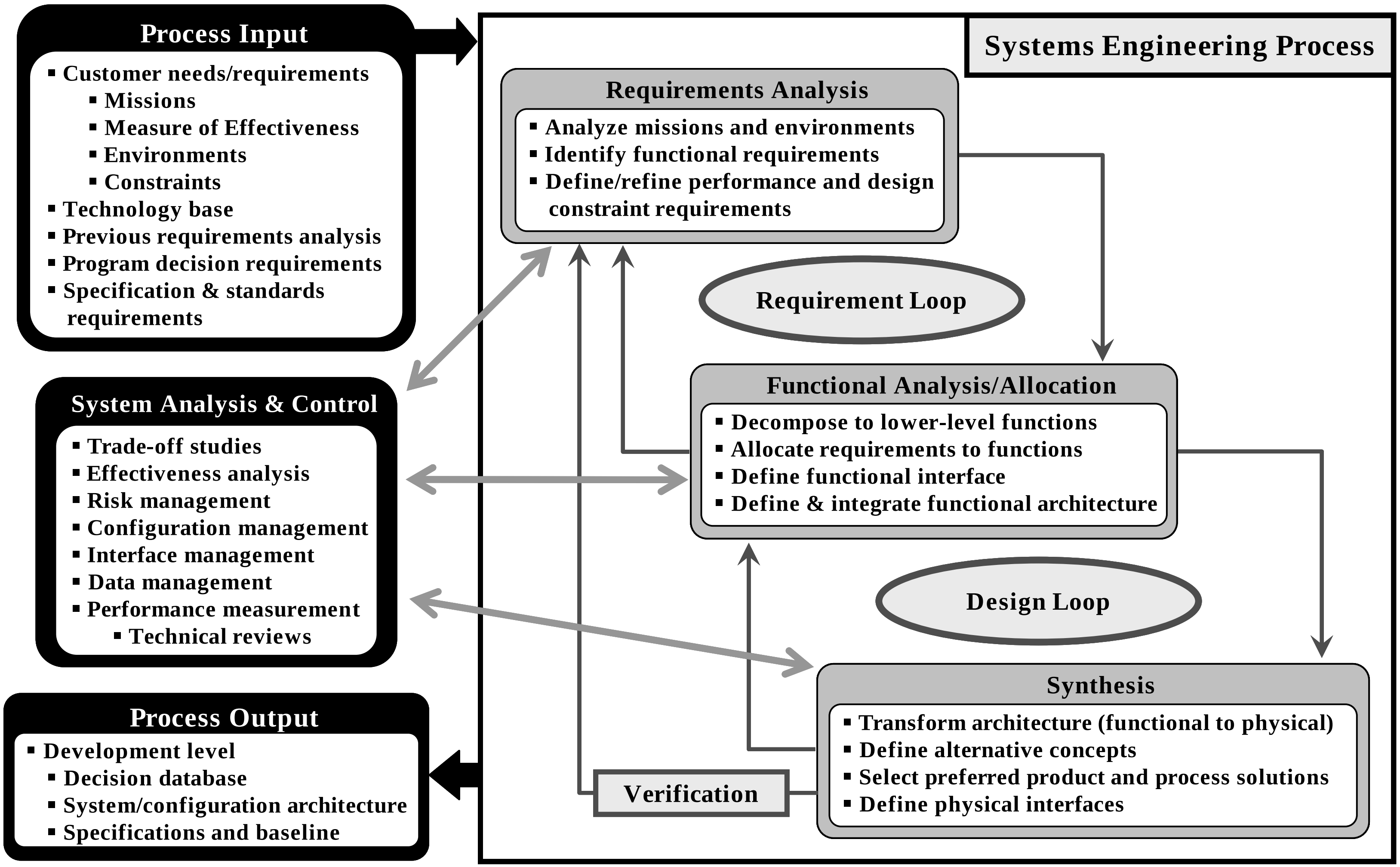
Interfaces for Engineering of a System



Systems Engineering Process Integrates
All these Functions Together for a given Program

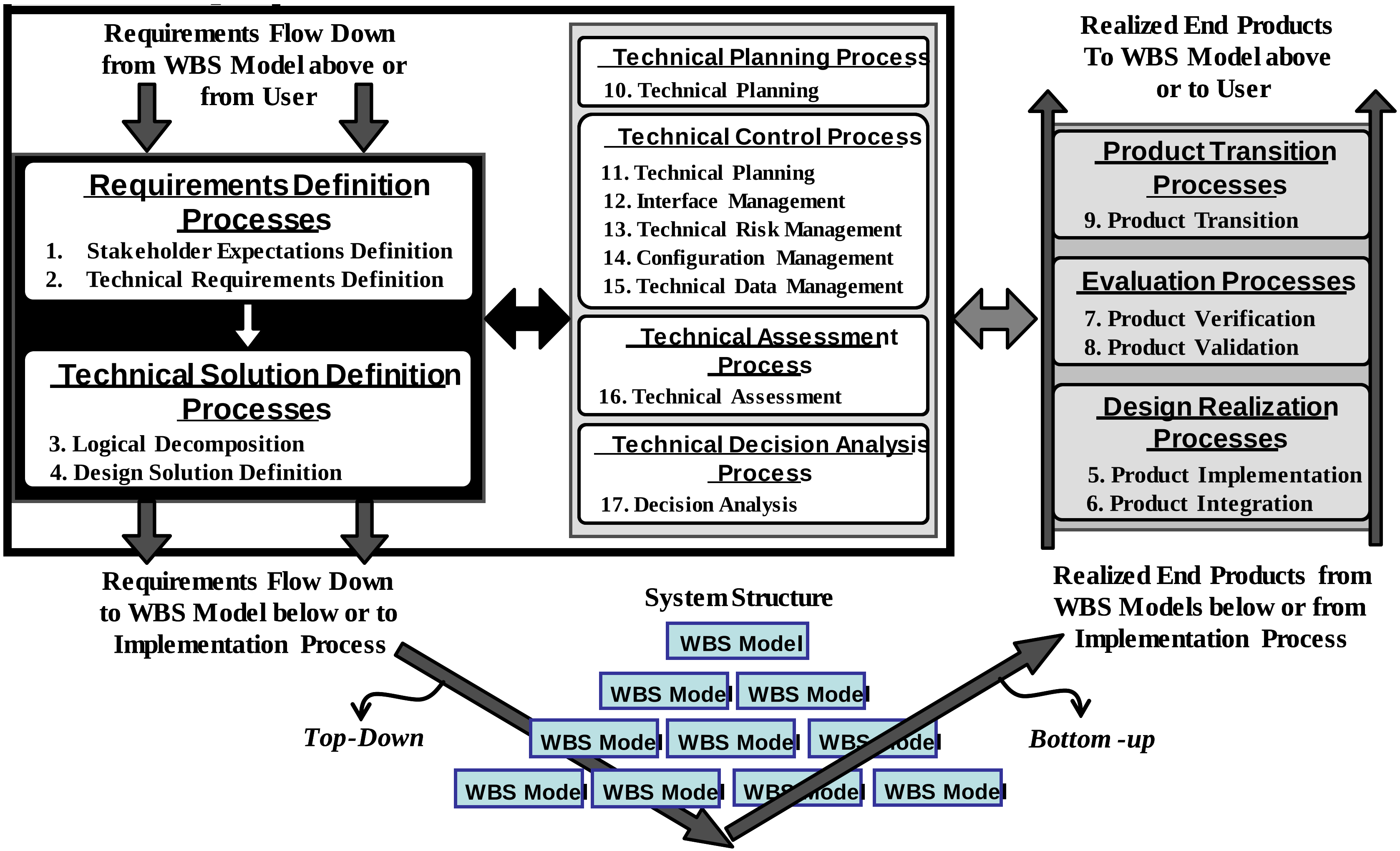
Dr. Dimitri N. Mavris, Director ASDL
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150
dimitri.mavris@ae.gatech.edu

DoD – Systems Engineering Process



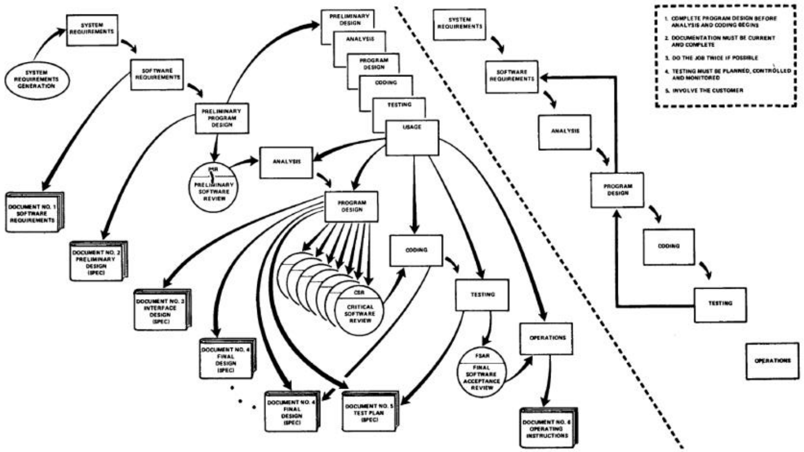
Dr. Dimitri N. Mavris, Director ASDL
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150
dimitri.mavris@ae.gatech.edu

NASA-Systems Engineering Process

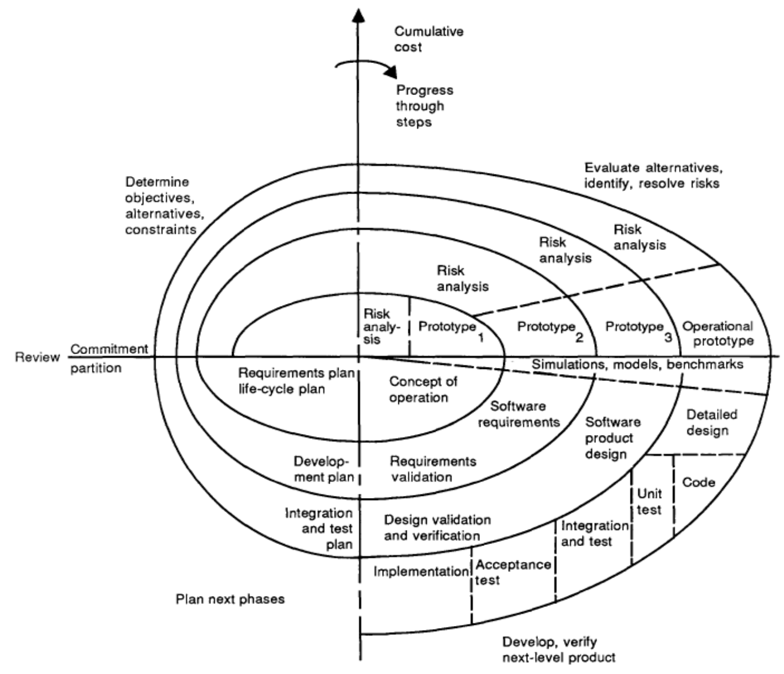


Dr. Dimitri N. Mavris, Director ASDL
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150
dimitri.mavris@ae.gatech.edu

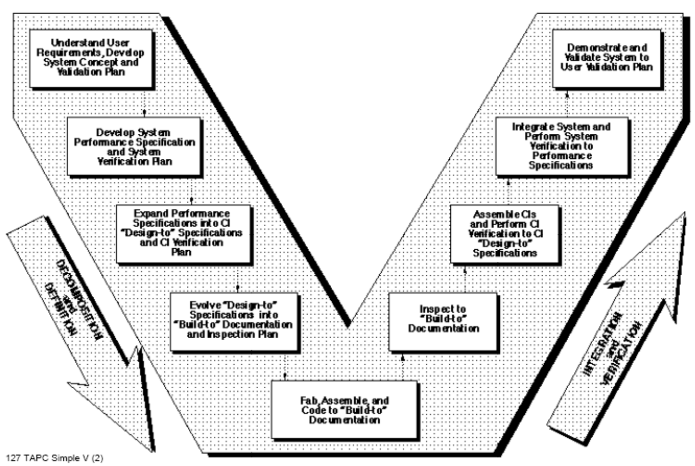
Common Systems Engineering Models



Royce's Waterfall Model, 1970



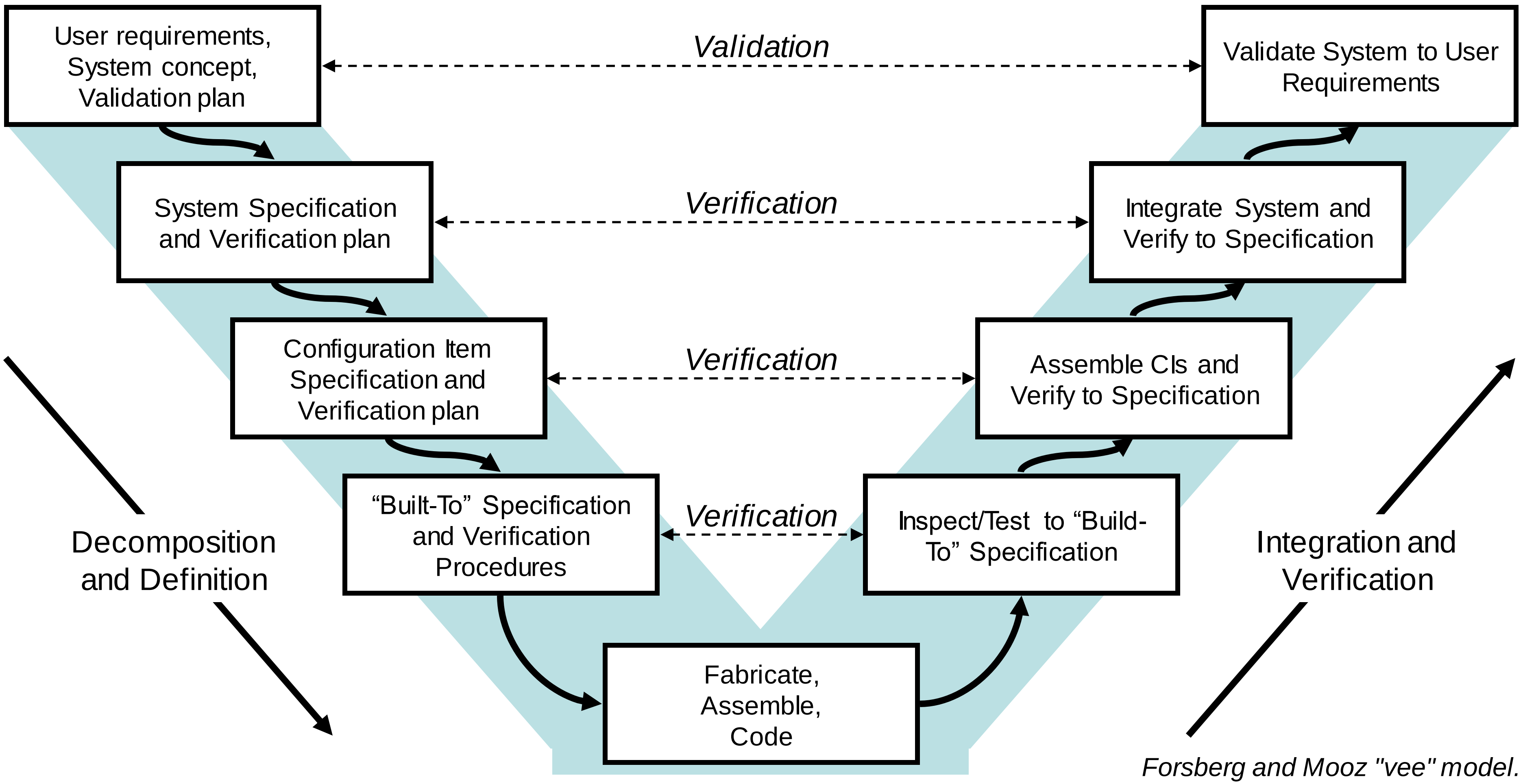
Bohem's Spiral Model, 1988



Forsburg and Mooz's Vee Model, 1995

Dr. Dimitri N. Mavris, Director ASDL
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150
dimitri.mavris@ae.gatech.edu

The "V" Model



Forsberg and Mooz "vee" model.

Dr. Dimitri N. Mavris, Director ASDL
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150
dimitri.mavris@ae.gatech.edu

Concepts and Principles of the V approach

- **Concepts**
  - System concepts are decomposed into requirements
  - Decomposition and Definition result in specification at the build and computer code level
  - Agreement is reached at every level and controlled
  - Realization of the solution is accomplished through Integration, Verification, and Validation (IV&V)

- **Principles**
  - The Vee model is interdisciplinary
  - Decomposition and definition are repeated until achieving specification of system parts and lines of code
  - Analysis is on the right of Vee and synthesis is on the left
  - Planning correlates the left of the Vee to the right side

Dr. Dimitri N. Mavris, Director ASDL
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150
dimitri.mavris@ae.gatech.edu

Elements of the SE Process

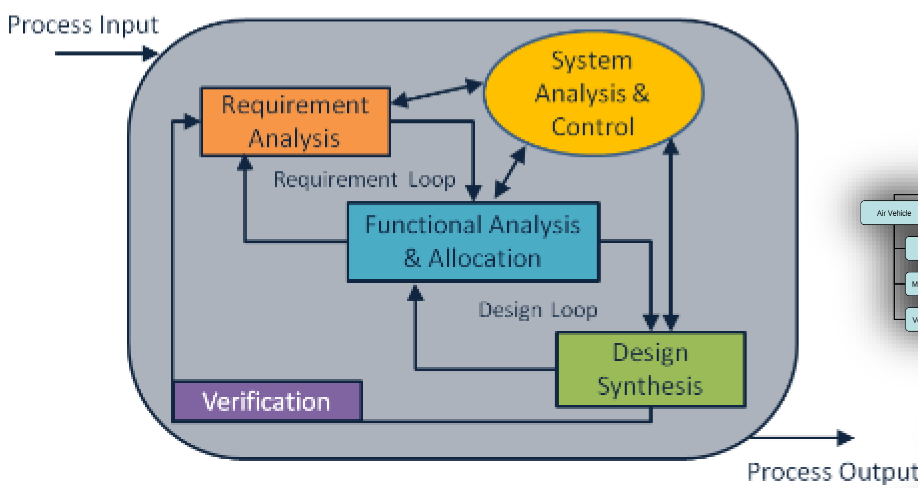
- As developed for the Department of Defense (DoD) the **Systems Engineering Process** includes three major elements:
  - ***Requirements Analysis***
  - ***Functional Analysis and Allocation***
  - ***Synthesis***
- This includes the techniques and tools to analyze and control the Systems Engineering Process
- The Systems Engineering Process is applied to each stage of life cycle development, one level at a time

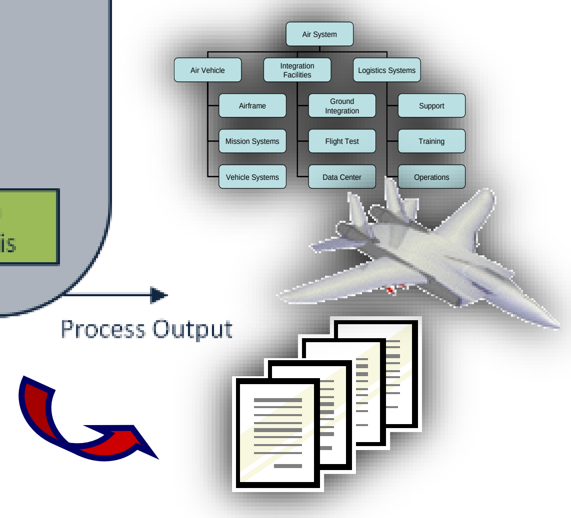
Dr. Dimitri N. Mavris, Director ASDL
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150
dimitri.mavris@ae.gatech.edu

Systems Engineering approach

The SE approach provides a structured but flexible process that transforms requirements into specifications, architectures, and configuration baselines

The discipline of this process provides the control and traceability to develop solutions that meet customer needs





The SE process may be repeated one or more times during any phase of the development process

Dr. Dimitri N. Mavris, Director ASDL
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150
dimitri.mavris@ae.gatech.edu

REQUIREMENTS ANALYSIS

Doctor Analogy Example

- Imagine that you have just moved to a new area. Among other tasks that you face, you need to ***choose a personal doctor*** from among those in your community
- What are the criteria that you would use to make this decision?

____ ☐ is knowledgeable and up-to-date
____ ☐ has experience in his/her field
____ ☐ makes conversations pleasant and enjoyable
____ ☐ discusses multiple treatment options
____ ☐ considers treatments other than medicine
____ ☐ is a clear and thorough communicator
____ ☐ is affiliated with a top-notch hospital
____ ☐ is located conveniently for appointments
____ ☐ will be in current location for many years
____ ☐ handles the paperwork for insurance, etc.
____ ☐ is a particular age or gender
____ ☐ respects and empathizes with my concerns
____ ☐ strives to minimize my out-of-pocket costs
____ ☐ makes it easy to schedule appointments
____ ☐ doesn't rush through examinations; takes appropriate time
____ ☐ has an excellent reputation
____ ☐ focuses on wellness in addition to sickness
____ ☐ can explain medical situations in lay terms
____ ☐ is courteous and respectful
____ ☐ has a competent and friendly office staff
____ ☐ prompt on appointments and returning phone calls
____ ☐ makes recommendations based on the pros and cons of alternatives to me
____ ☐ can schedule emergencies soon
____ ☐ knows many quality specialists for referrals
____ ☐ ______________________
____ ☐ ______________________
____ ☐ ______________________
____ ☐ ______________________
____ ☐ ______________________
____ ☐ ______________________
____ ☐ ______________________

- Now imagine if this list was provided
  - Does this make it easier?
  - How many of your criteria were on this list as well?

Dr. Dimitri N. Mavris, Director ASDL
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150
dimitri.mavris@ae.gatech.edu

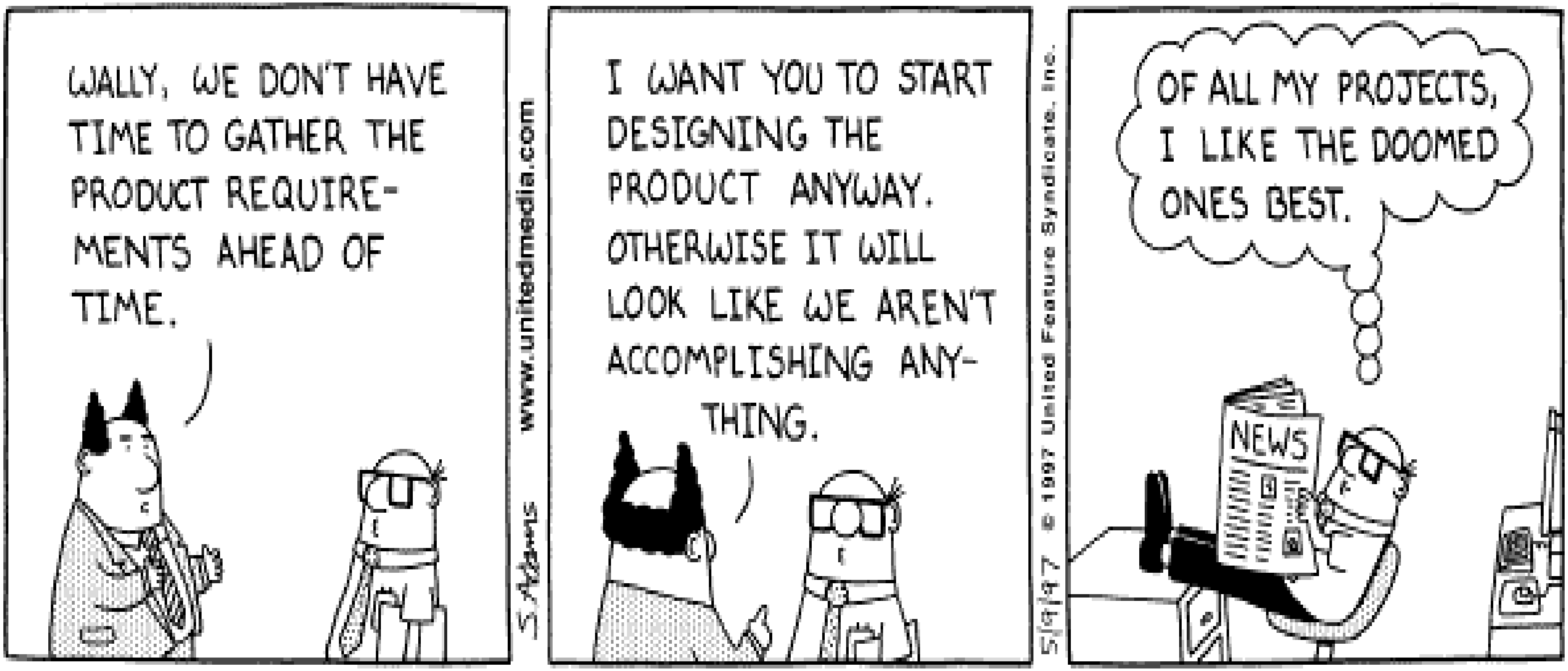
Georgia Institute of Technology
Aerospace Systems Design Laboratory

Objectives

- Learn more about the requirements analysis process
  - How to create “good” requirements?
  - How to map requirements with operational and system perspectives?
  - What are the current requirements analysis methodologies?

Dr. Dimitri N. Mavris, Director ASDL
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150
dimitri.mavris@ae.gatech.edu

Introduction to Requirements Analysis



Dr. Dimitri N. Mavris, Director ASDL
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150
dimitri.mavris@ae.gatech.edu

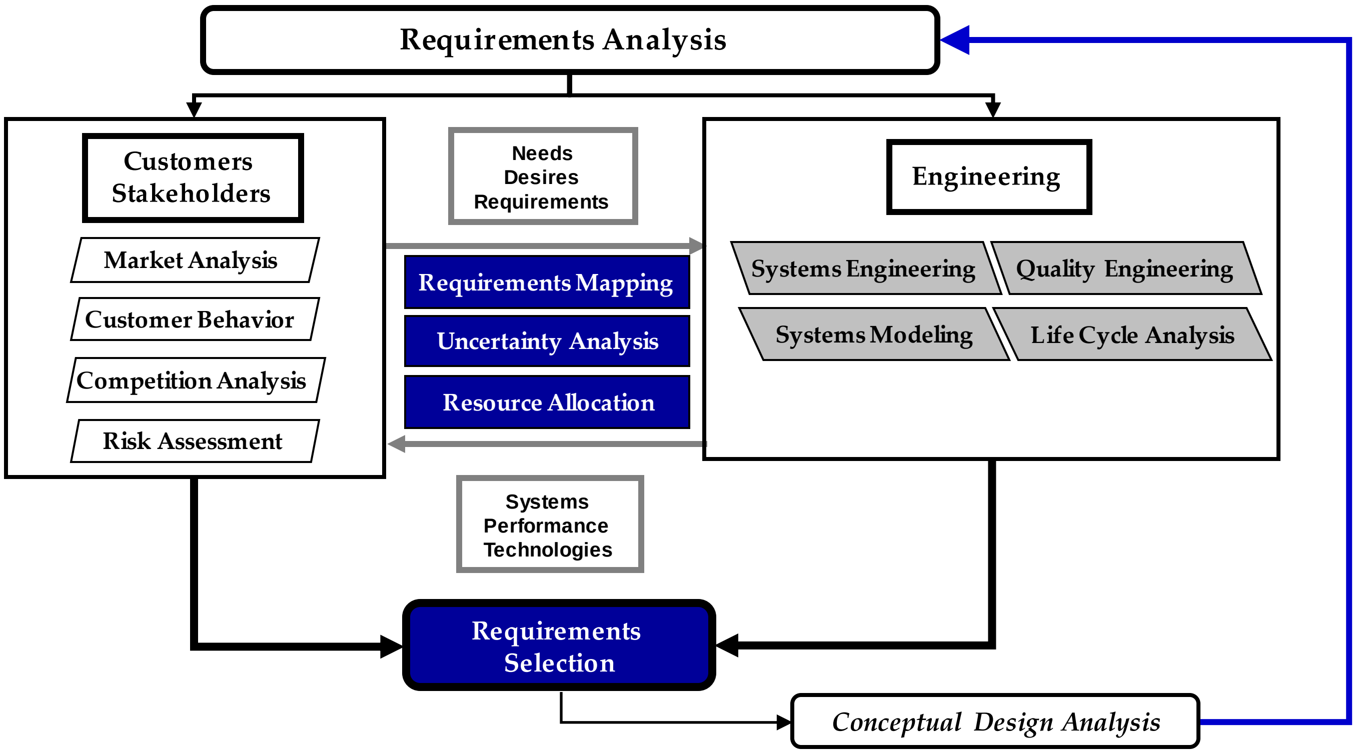
Some Useful Definitions

- **Requirement**
  - "A **statement** that identifies a system, product or process characteristic or constraint, which is unambiguous, can be verified and is deemed necessary for the stakeholder acceptability."
- **Well-formed requirement**
  - Includes the system functionality (capability), attributes (measurable conditions), and is bounded by constraints.
- **Constraint**
  - The boundary conditions, externally or internally imposed, for the system of interest within which the organization must remain when the executing the processes during the concept and development stage.

Dr. Dimitri N. Mavris, Director ASDL
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150
dimitri.mavris@ae.gatech.edu

Overview

- **Requirements**
  - Where do they come from?
  - Who do they affect?
  - What do they affect?



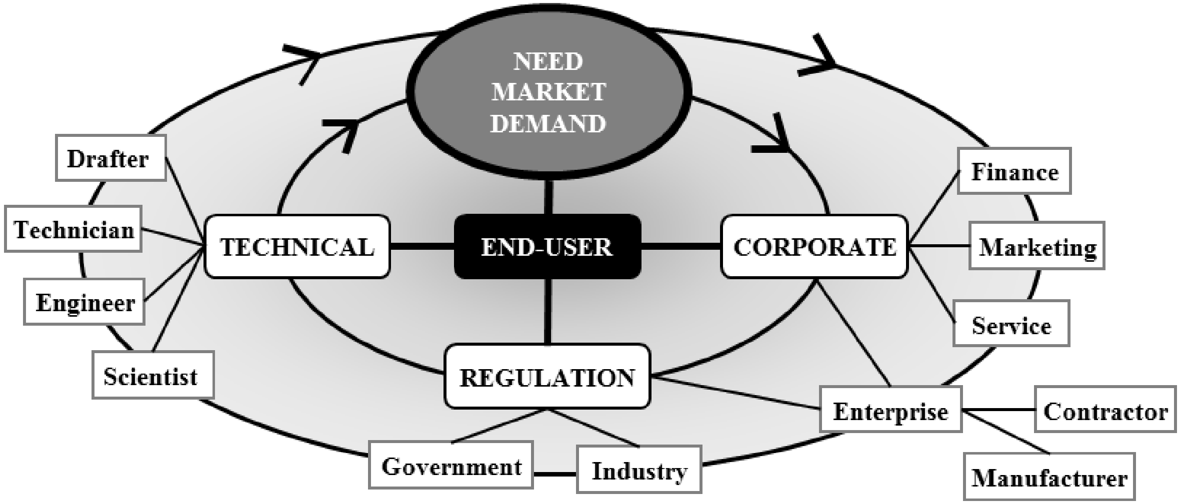
Sources of Requirements

- Interviews with users
- The system's environment
- Business objectives
- Marketing requirements
- Contractual obligations
- Working in user environment
- Analogous or existing systems
- Change suggestions, problem reports
- User modifications
- User group meetings
- Workshops
- Prototypes
- Studies
- Innovation work
- Questionnaires
- Consultants & Trainers
- Observation / Fieldwork

Dr. Dimitri N. Mavris, Director ASDL
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150
dimitri.mavris@ae.gatech.edu

Who are the Stakeholders?

- **Stakeholder:** Anyone who has a right to impose requirements on the system
  - End users, operators, bill payers, owners regulatory agencies, victims, sponsors, producers…
- **Stakeholders for a to-be-built aircraft**
  - End users: passengers that fly on the airplane
  - Operators: flight crews and mechanics
  - Bill payers: airline companies
  - Owners: stockholders in these companies
  - Regulators: FAA, EPA, aviation authorities
  - Victims: people living near the airports and competing companies
  - Sponsors: A/C manufacturer corporate headquarters, DoD



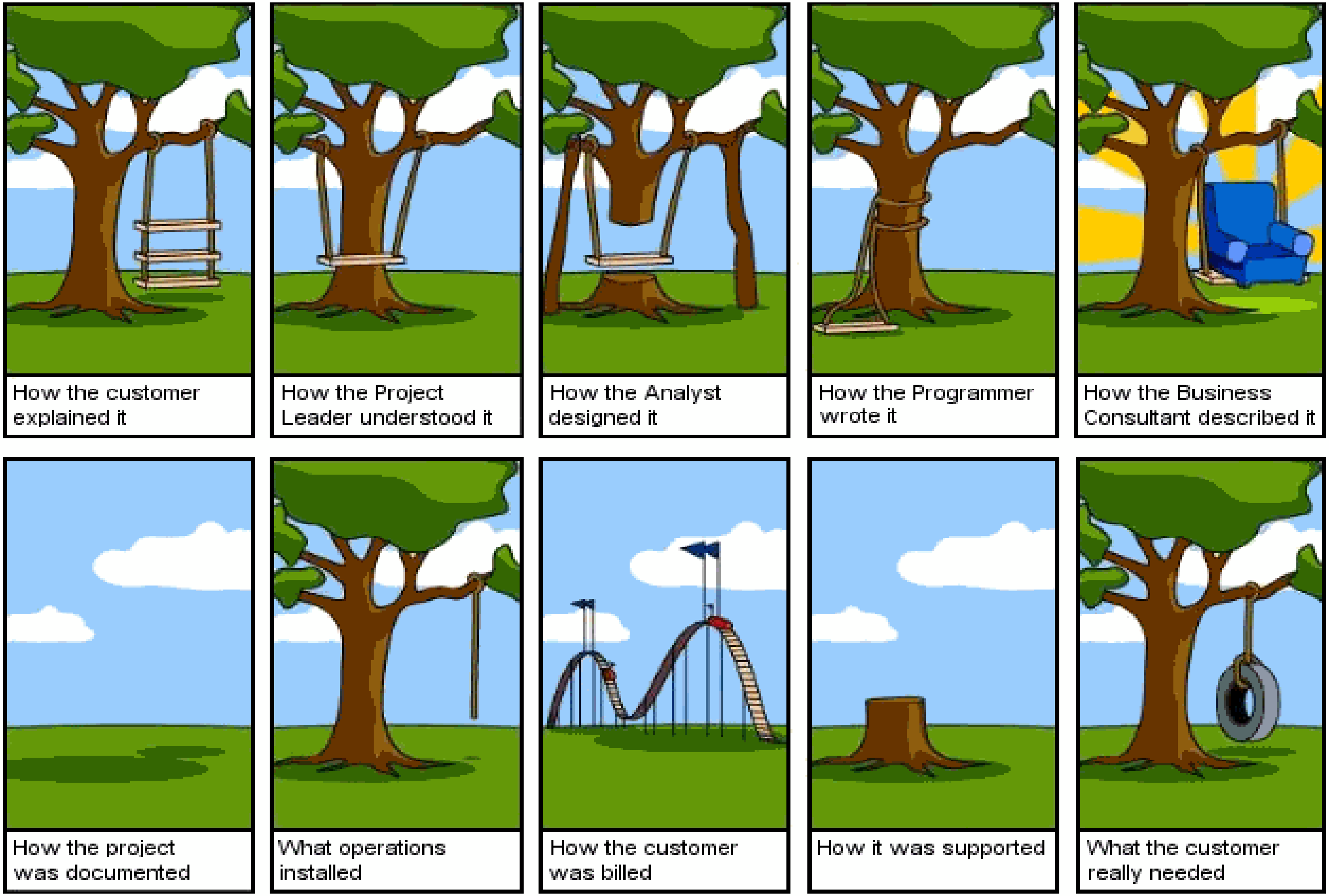
Dr. Dimitri N. Mavris, Director ASDL
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150
dimitri.mavris@ae.gatech.edu

Typical Number of Requirements

	Complexity				
	1. Tiny	2. Small	3. Medium	4. Large	5. Huge
Design professionals (number)	3	15	75	400	2,000
Duration (months)	12	24	36	48	60
Geographical area	Building	City	Multi-city	Country	World
States/modes (number)	3	30	300	2,000	10 000
States/modes (type)	Well defined	Deterministic		Stochastic	
Interface params (no.)	9	90	900	6,000	30 000
Physical environment of product	Indoor	In/outdoor 1 location	Outdoor global	Global	Space, Ocean
Requirements (no. of text pages)	15	150	500	2 000	50000
Requirements (approximate no)	150	1 500	5 000	20 000	500 000
Number of specs generated	1	3	10	200	1000
Examples	Electric kitchen mixer; low budget art film	Housing development; TV Set; grand opera	Night viewing system; highway system	Aircraft	Space vehicle; Battleship & armament

Dr. Dimitri N. Mavris, Director ASDL
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150
dimitri.mavris@ae.gatech.edu

Why Are Requirements Important?



Clearly Communicating Requirements is Essential

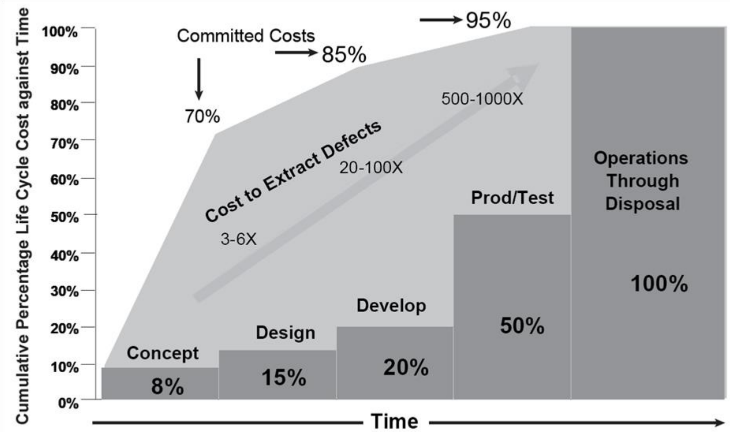
Dr. Dimitri N. Ma
School of Aerospa
Georgia Institute
Atlanta, GA 303
dimitri.mavris@

Propulsion Centric View

Importance of Requirements

- A large portion of the project cost are committed in conceptual Design

> "…the ability to define the problem is the most important and difficult task in engineering."
> *[Suh, 1990] – The Principles of Design*



Source: INCOSE

- What is the importance of requirements in a project success?
  - **More than 22 %** of project failure can be related to requirements

Reasons for Project Failure (*Software*)

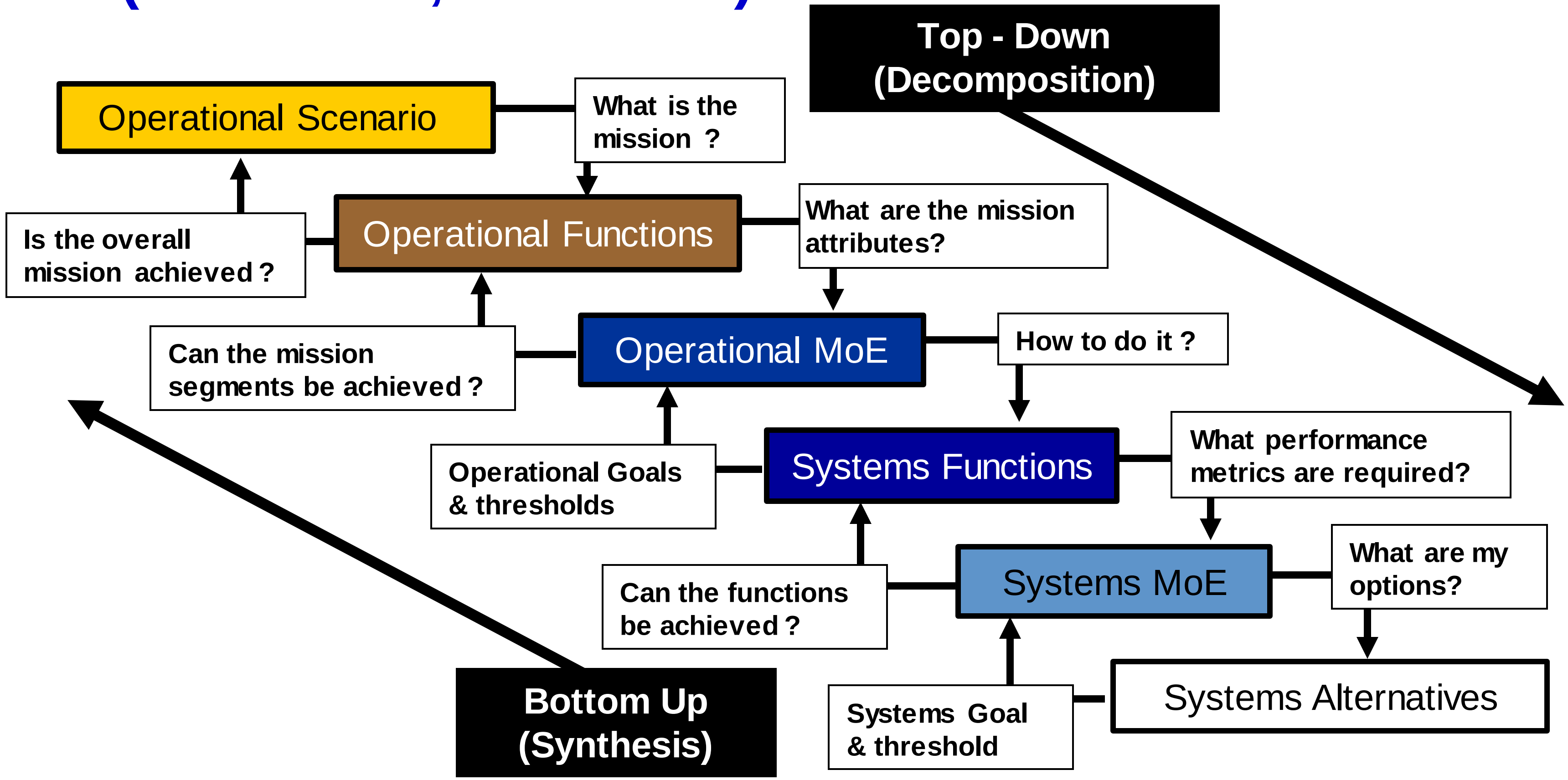
Incomplete requirements	**13.1%**
Lack of resources	**12.4%**
Unrealistic expectation	**10.6%**
Lack of executive support	**9.9%**
Changing requirements/specifications	**8.7%**
Didn't need it any longer	**7.5%**

The Chaos Report 1994 (Standish Group)

Dr. Dimitri N. Mavris, Director ASDL
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150
dimitri.mavris@ae.gatech.edu

From Operations to System Alternatives

- There are Requirements at every level of the design process
  - The requirements information needs to be **TRACEABLE (flow-down, allocation)**



Dr. Dimitri N. Mavris, Director ASDL
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150
dimitri.mavris@ae.gatech.edu

Characteristics of Good Requirements

- Describes what, not how
- Is single purpose
- **Unique**
- Documented & accessible
- Identifies its owner
- Approved & dated
- **Traceable**
- Necessary
- Complete
- States its rationale
- Avoids synonyms & homonyms
- Is not always written
- Quantifiable & testable
- Identifies applicable states
- States assumptions
- Use shall, should, and will
- Avoids use of optimize, maximize, & minimize
- Might vary in level of detail
- Respects the Media

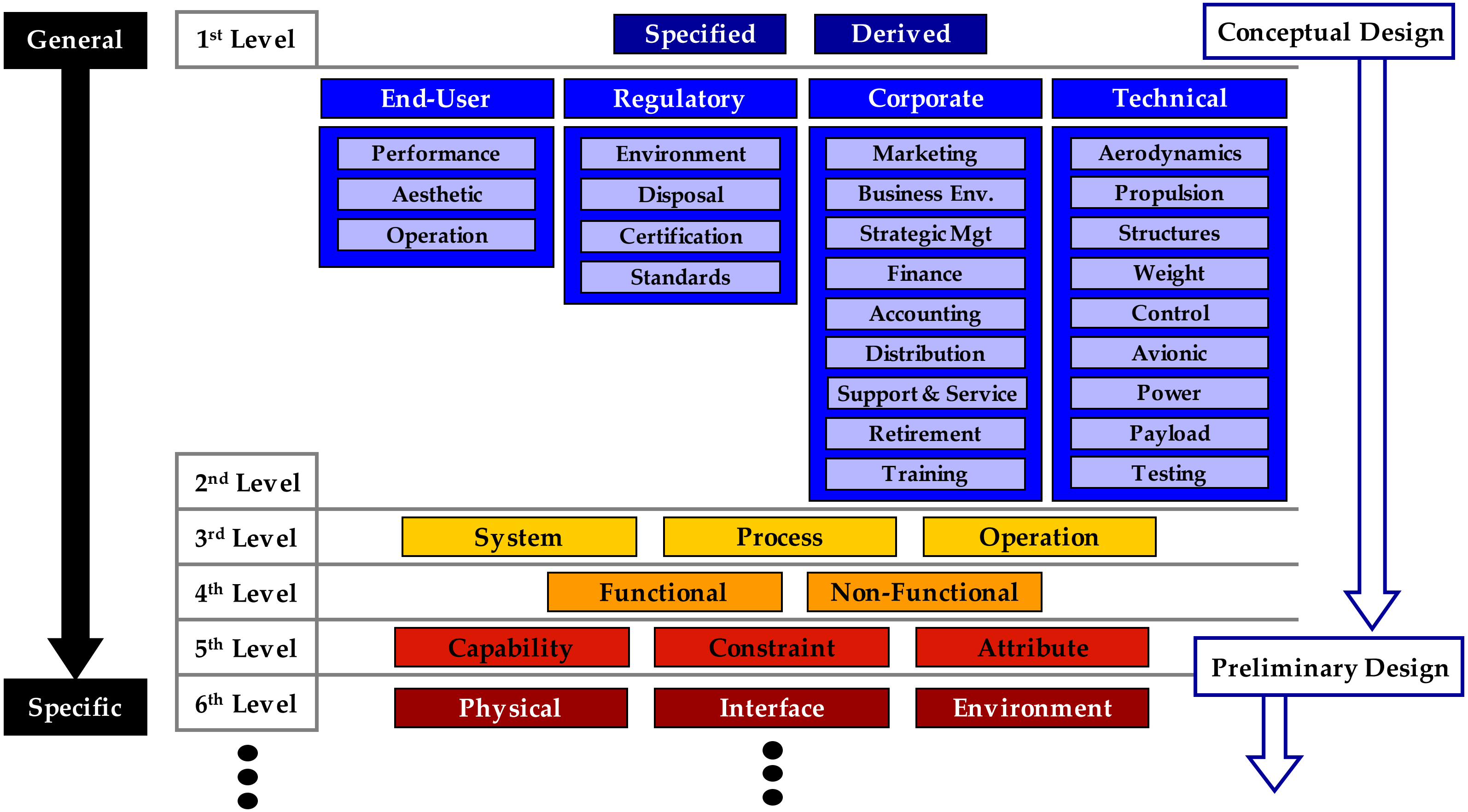
Requirements Validation

- Validation: check that the requirements define the system that the Client really wants
- Requirements validation techniques
  - Manual reviews (careful readers from Client and Contractor)
  - Demonstration of prototype
  - Test case generation out of requirements - failure to design test cases may reveal problems in the requirement

Requirements Checkpoints

- Consistency: Are there any requirements conflicts?
- Completeness: Are all functions required by the customer (and stakeholders) included?
- Realism: Can the requirements be implemented given available budget and technology
- Verifiability: Is the requirement realistically testable (in implementation)?
- Traceability: Is the source of the requirement clearly stated (in case of change)?

Requirements Classification: Taxonomy

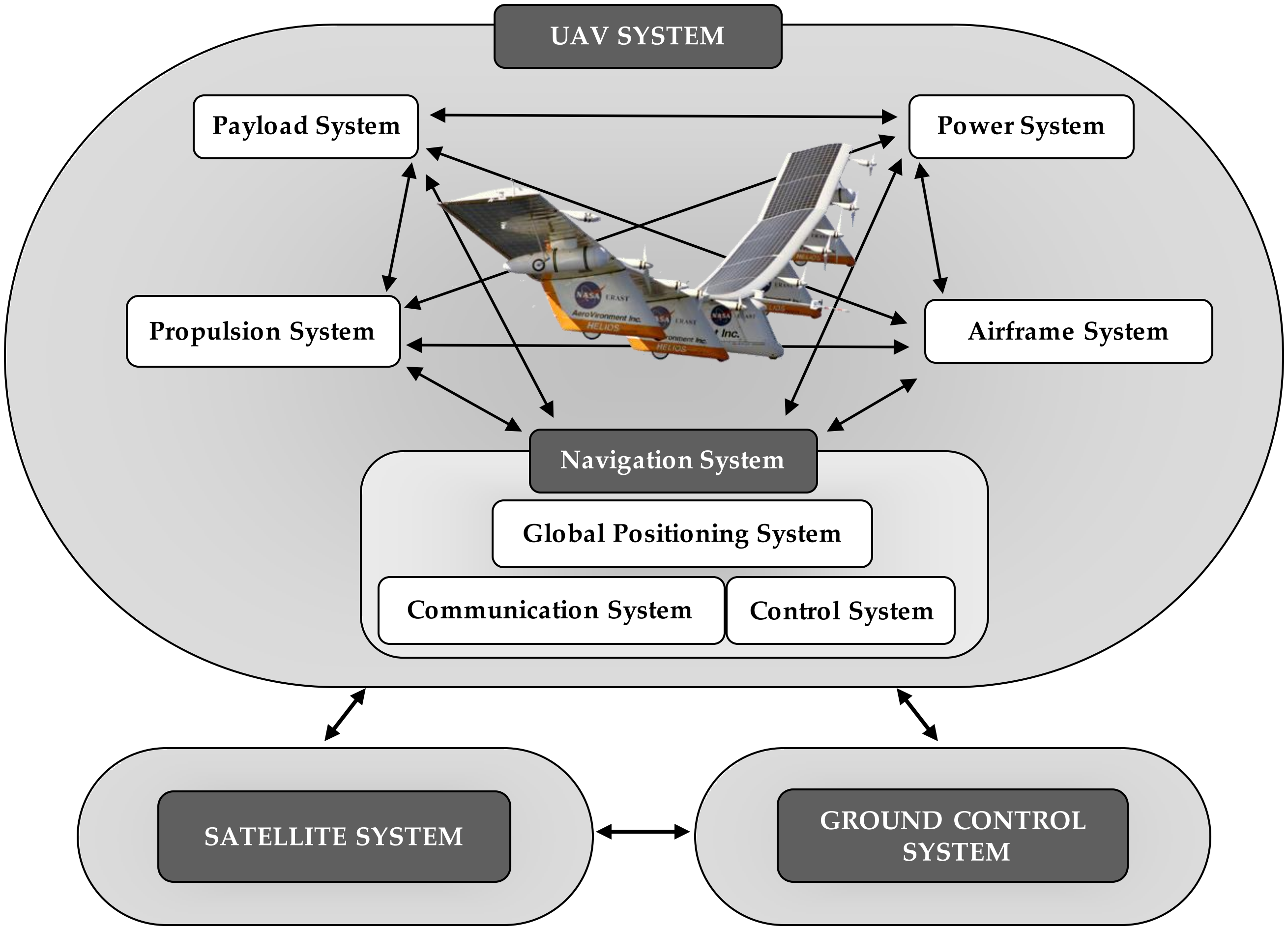


Dr. Dimitri N. Mavris, Director ASDL
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150
dimitri.mavris@ae.gatech.edu

REQUIREMENTS ANALYSIS EXAMPLE

Complex Systems Analysis

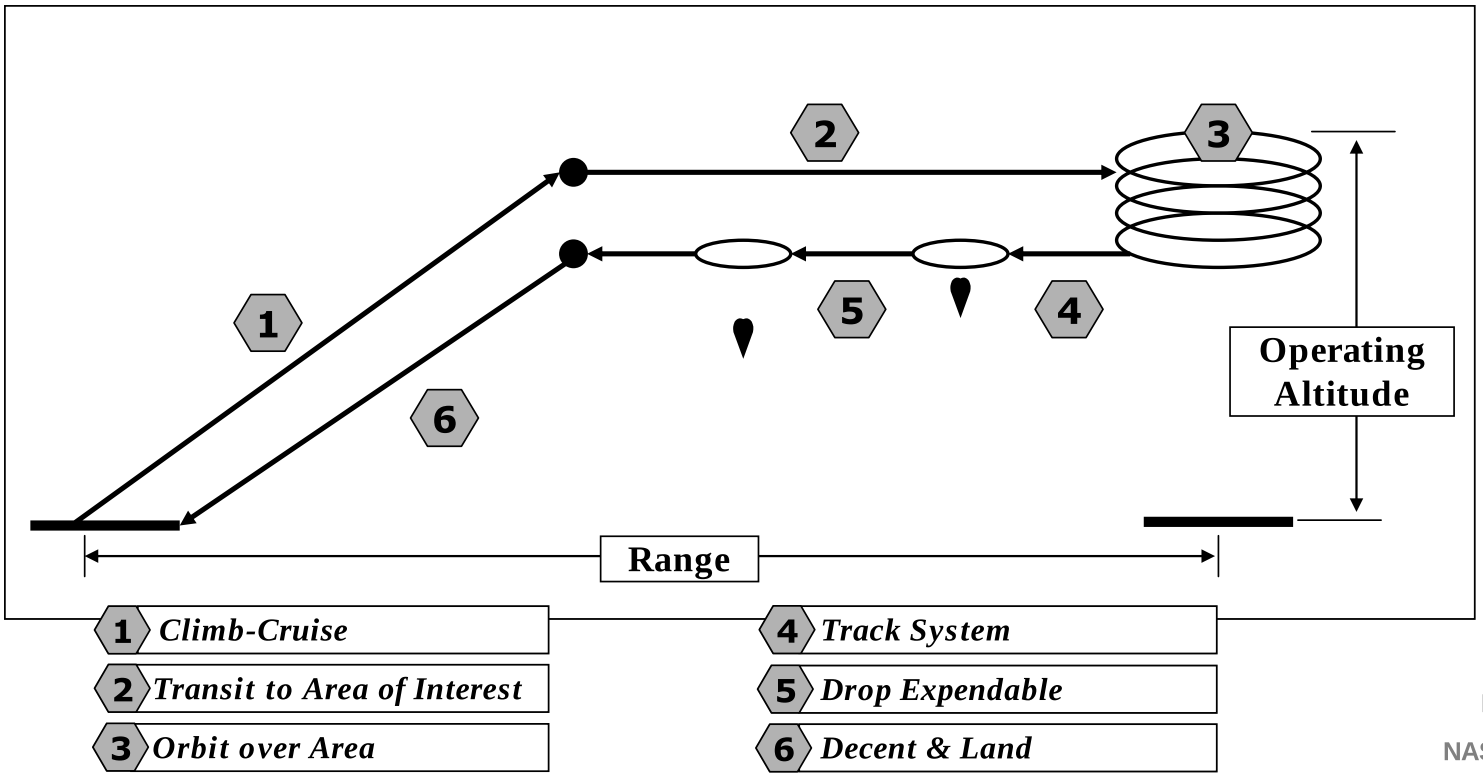
Large Number of Requirements: Where to start ?



Dr. Dimitri N. Mavris, Director ASDL
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150
dimitri.mavris@ae.gatech.edu

Hurricane Tracker UAV

- Define the Concept of Operations (CONOPS)
- Define the system architecture using a matrix of alternatives



Modified from:

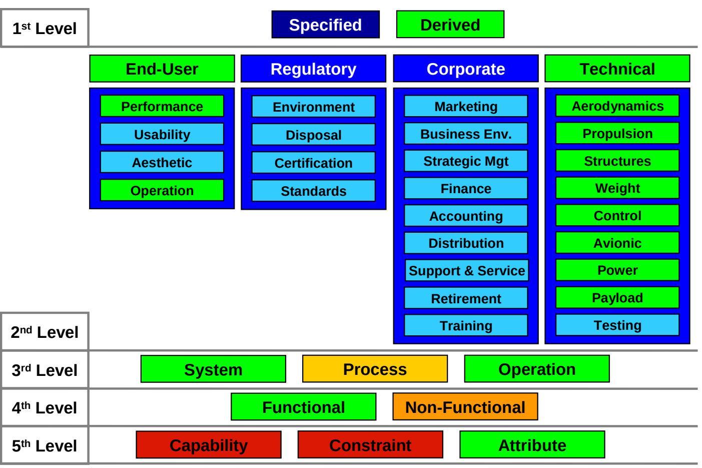
NASA TP-2007-21486

Dr. Dimitri N. Mavris, Director ASDL
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150
dimitri.mavris@ae.gatech.edu

Georgia Institute of Technology
Aerospace Systems Design Laboratory

Operations and Systems Decomposition

- Use the taxonomy to classify the systems and operational requirements



MISSION SEGMENT	FUNCTION (*actor-verb-object*)	MEASURE OF EFFECTIVENESS		
1 - Takeoff	UAV has to takeoff	1 - Runway dimension		
2 - Climb	UAV climbs to optimum altitude	1 - Climb rate	2 - Climb altitude	3 - Structural stress
3 - Transit to area of interest	UAV transits to area of interest	1 - Cruise Speed	2 - Cruise altitude	3 - Range
4 - Orbit over area	UAV orbits over area of interest	1 - Time on station	2 - Endurance altitude	3 - Endurance speed
		4 - Latitude Range	5 - Rate of Sink	6 - Collect Data
		7 - Store Data		
5 - Track storm system	UAV tracks object of interest	1 - Tracking speed	2 - Tracking altitude	
	UAV modifies flight path	3 - Max turn rate		
6 - Drop expendables	UAV drops expendable	1 - Drop speed	2 - Drop altitude	
Return to base	UAV transits from area of interest	*1 - Cruise Speed*	*2 - Cruise altitude*	*3 - Range*
7 - Descend	UAV descent to base	1 - Descend rate		
8 - Land	UAV lands at base	1 - Landing speed	2 - Runway dimension	

SYSTEM	FUNCTION (*actor-verb-object*)	MEASURE OF EFFECTIVENESS		
1 - Airframe	Airframe provides sufficient lift	1 - L/D	2 - $C_{L\,MAX}$	3 - Wing Span
		4 - Wing Area	5 - Wing Aspect Ratio	
	Airframe carries payload	1 - Weight	2 - Internal volume	3 - Wing loading
2 - Propulsion	Energy is converted into mechanical work	1 - Prop. Specific energy	2 - Conversion efficiency	3 - Energy consumption
3 - Fuel	Fuel provides energy to systems	1 - Fuel specific energy		
4 - Power	Power systems provides electrical energy	1 - Power Specific energy		
5 - Avionics & CTRL	Avionics determine vehicle state variables	1 - Positioning accuracy		
	Avionics stores information	1 - CPU speed		
6 - Communication	Comm. transmit and receive information	1 - Comm. power	2 - Comm. Bandwidth	3 - Antenna size
		4 - Comm. Availability	5 - Comm. robustness	
7 - Payload	Payload sustains environment	1 - Operating conditions		
	Payload collect data	1 - Payload power	2 - Sensor resolution	3 - Acquisition rate
		4 - Field of view		
8 - Ground Station (GS)	GS monitor the vehicle	1 - Line of Sight range	2 - Satellite range	
9 - Expendable Payload	Expendable collect vertical measurements	1 - Number of expendable units		

Dr. Dimitri N. Mavris, Director ASDL
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150
dimitri.mavris@ae.gatech.edu

AN EXAMPLE WITH AMBIGUOUS REQUIREMENTS

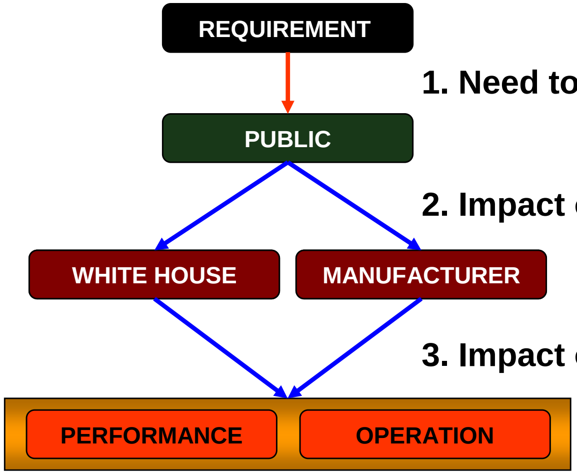
U.S. Presidential Helicopter

- Marine One is most photographed helicopter on the planet
  - The new helicopter needs to carry the same image

Explicit Requirement: The helicopter should look Presidential

Source: www.history.navy.mil

[R] → **The helicopter shall look "*Presidential* "**



1. Need to satisfy ?

2. Impact on Other Stakeholders ?

3. Impact on the System?

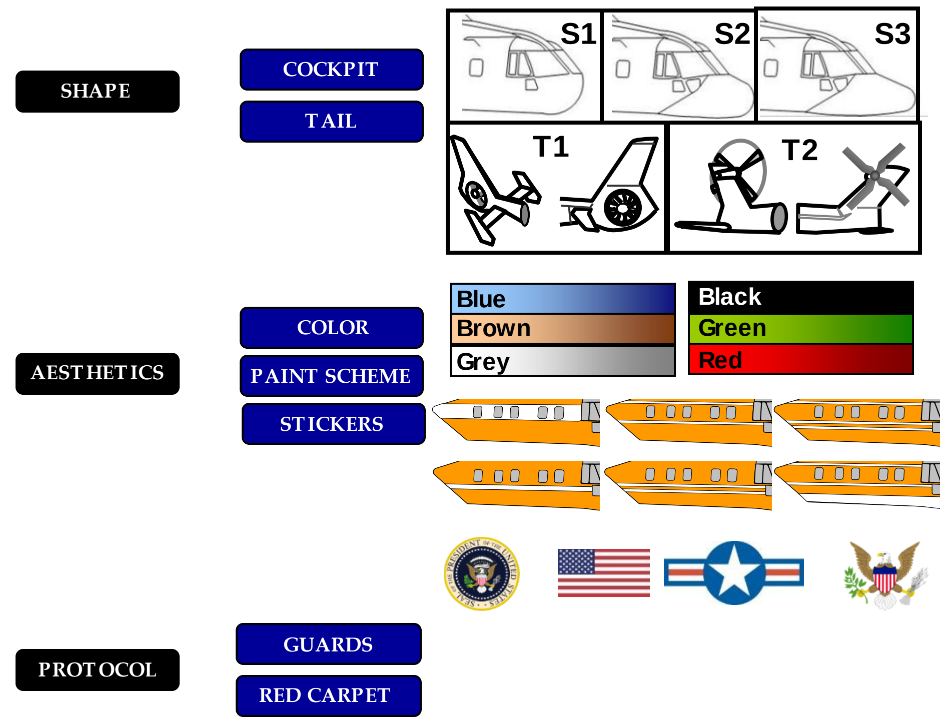
Dr. Dimitri N. Mavris, Director ASDL
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150
dimitri.mavris@ae.gatech.edu

Taxonomy Applied to Presidential Helicopter

- Taxonomy can be used as an initial framework for brainstorming exercises
- The information can be classified through the taxonomy, and used to structure the requirements mapping

End-User	Regulatory	Corporate	Technical
Performance	Environment	Marketing	Airframe
Aesthetic	Disposal	Business Env.	Propulsion
Operation	Certification	Strategic Mgt	Structures
	Standards	Finance	Weight
		Accounting	Control
		Distribution	Avionic
		Support & Service	Power
		Retirement	Payload
		Training	Testing

2nd Level



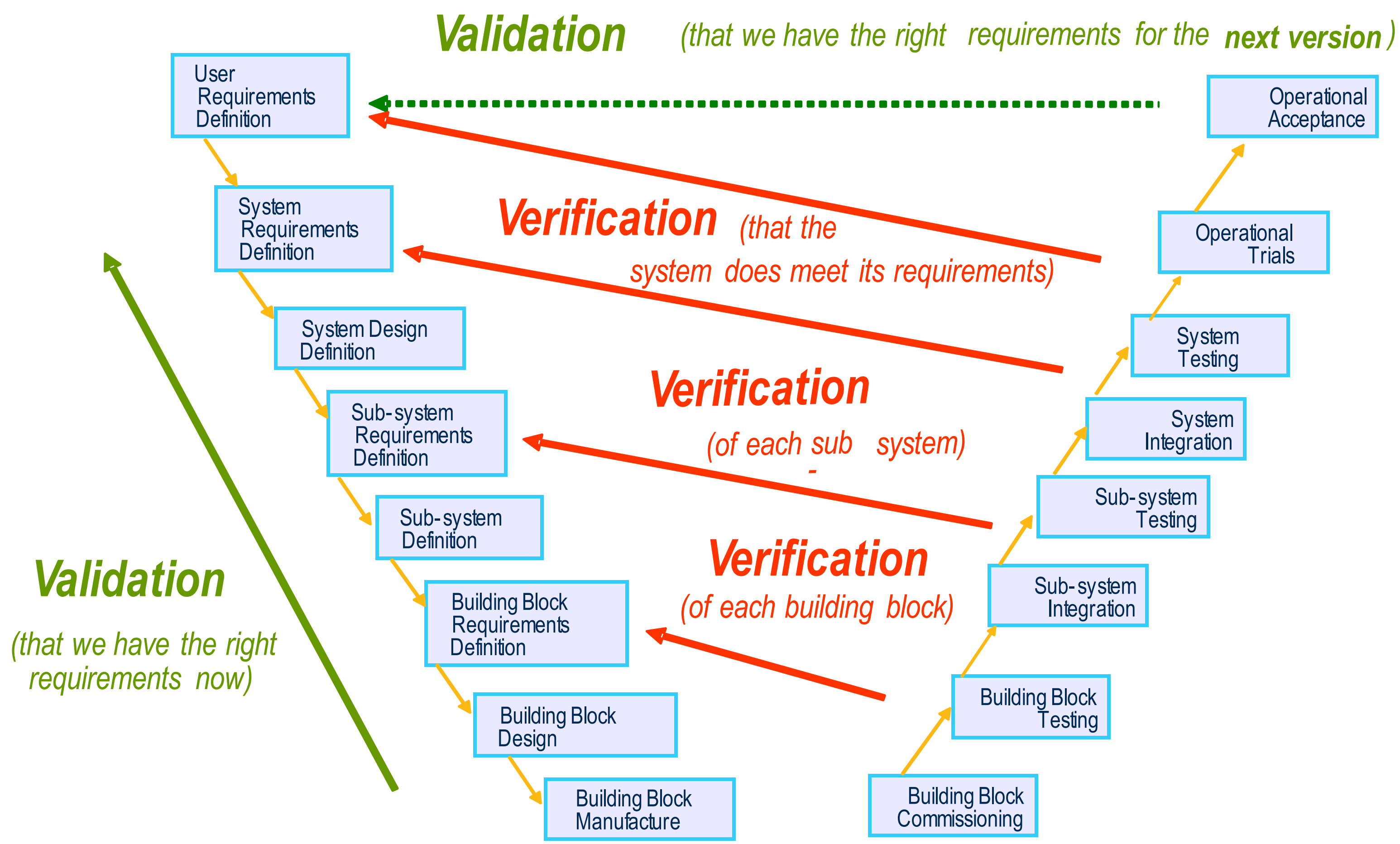
Dr. Dimitri N. Mavris, Director ASDL
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150
dimitri.mavris@ae.gatech.edu

REVIEW REQUIREMENTS

Review Requirements

- Involve the stakeholders!
  - Stakeholders should be convinced that if the requirements are met, then the system will do what it needs to do
    - Verbatim requirements → Agreed requirements
- Ask, why each requirement is needed:
  - Helps to eliminate unneeded requirements
  - Helps to reveal the requirements behind the state requirements
    - It may be easier to satisfy the requirements behind the requirements, than the stated requirements themselves
- Check correctness, completeness, consistency, ranking…
- Perform formal requirements reviews, document agreement and distribute the results

And Requirements are the Foundation of V&V!



Dr. Dimitri N. Mavris, Director ASDL
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150
dimitri.mavris@ae.gatech.edu

Key points

- Requirements define what the **system should provide** and define system constraints
- Requirements **problems lead to late delivery and change** requests after the system is in use
- Requirements engineering is concerned with **eliciting**, **analyzing**, and **managing** the system requirements
- Requirements needs to match with the **available resources**

Summary Requirements Engineering

Effectively generating High Quality Requirements

- *correct*
- *consistent*
- *complete*
- *modifiable*
- *traceable*
- *verifiable*
- *non-ambiguous*
- *understandable*
- *annotated*

Requirements Engineering

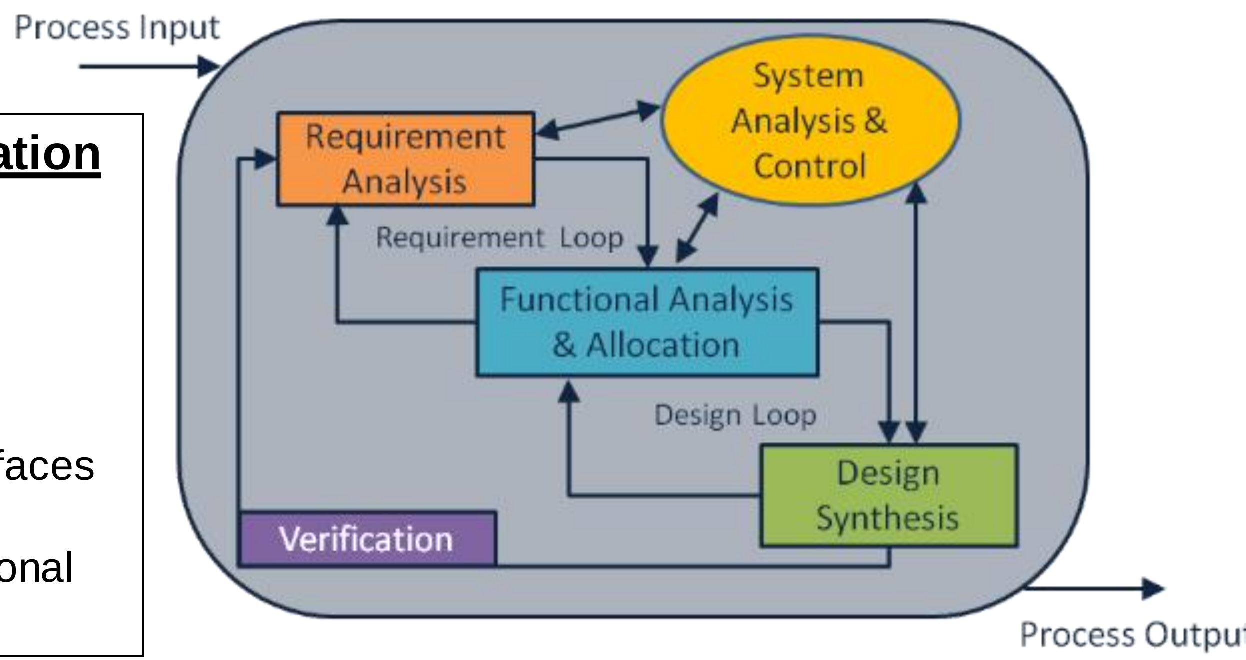
- **Requirements Development**
  - Elicitation
  - Analysis
  - Modeling & Specification
  - Verification & Validation
- **Requirements Management**
  - Change Control
  - Version Control
  - Tracing & Impact Analysis
  - Status Tracking

Best Practices, Methods,Tools and Processes for Requirements Engineering

Dr. Dimitri N. Mavris, Director ASDL
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150
dimitri.mavris@ae.gatech.edu

FUNCTIONAL ANALYSIS & ALLOCATION

What is Functional Analysis Allocation?



<u>Functional Analysis Allocation</u>

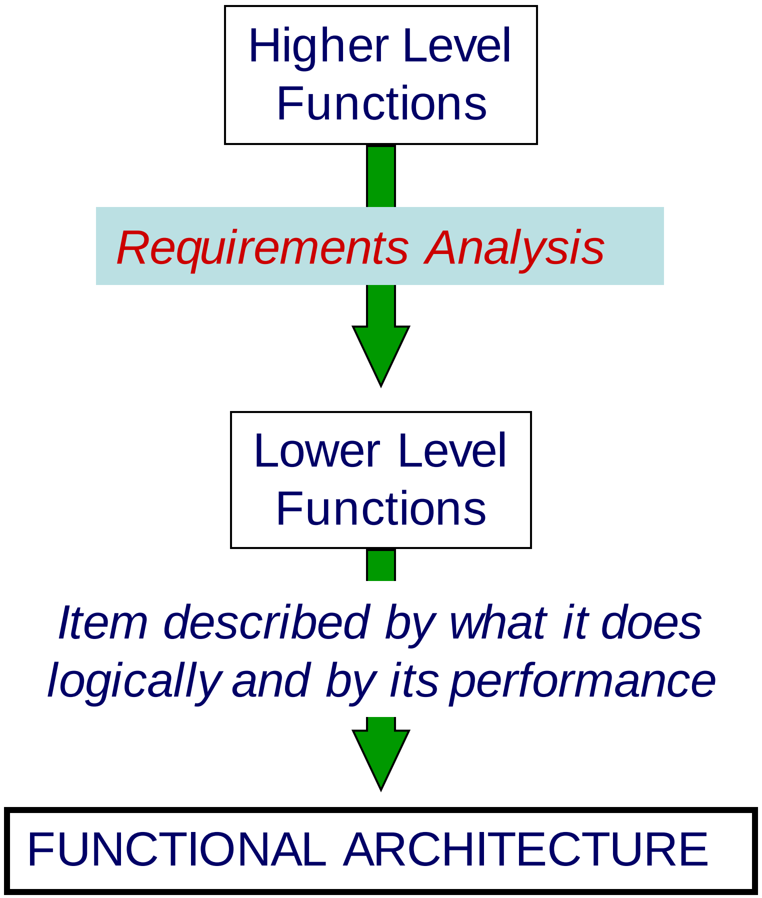
- Decompose to Lower-Level Functions
- Allocate Performance & Other Limiting Requirements to All Functional Levels
- Define/Refine Functional Interfaces (Internal/External)
- Define/Refine/Integrate Functional Architecture

Source: Systems Engineering Fundamentals, DoD Guide

Functional Analysis Allocation

Functions are analyzed by DECOMPOSING higher-level functions into lower-level functions through Requirements Analysis

Functional Analysis and Allocation allows for a better understanding of what the system has to do, in what ways it can do it, and often the priorities and conflicts associated with lower-level functions



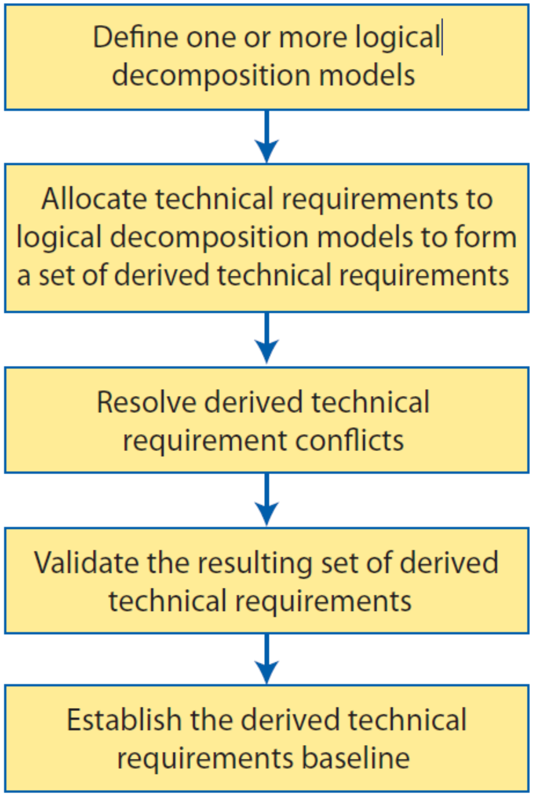
Source: Systems Engineering Fundamentals, DoD Guide

Dr. Dimitri N. Mavris, Director ASDL
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150
dimitri.mavris@ae.gatech.edu

Logical Decomposition

- The process for creating the detailed functional requirements that enable the designed system to meet the stakeholder expectations

- Utilizes functional analysis to create a system architecture and to decompose top-level (or parent) requirements and allocate them down to the lowest desired levels of the system

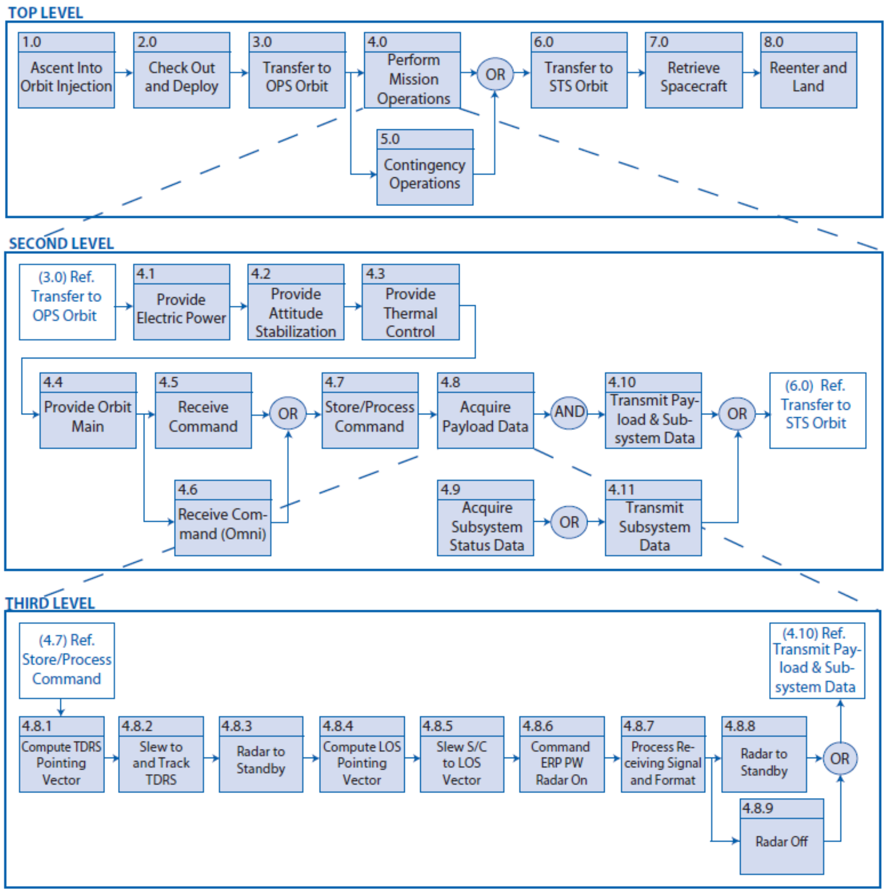


Logical Decomposition Process [NASA SE Handbook]

Dr. Dimitri N. Mavris, Director ASDL
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150
dimitri.mavris@ae.gatech.edu

Functional Analysis Techniques

- **Functional Flow Block Diagrams**
  - To indicate the sequential relationship of all functions that must be accomplished by a system
  - Show the entire network of actions that lead to the fulfillment of a function
  - Identify "what" must happen and must not assume a particular answer to "how" a function will be performed
    - The "how" is then defined for each block at a given level by defining the "what" functions at the next lower level necessary to accomplish that block

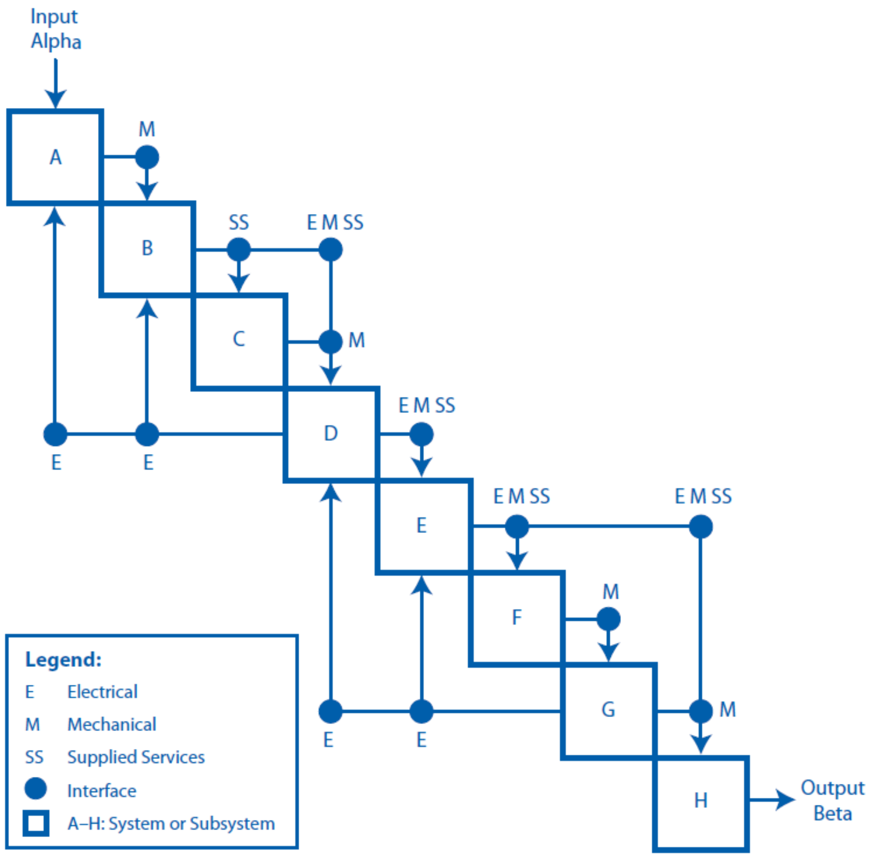


Example of a FFBD [NASA SE Handbook]

Dr. Dimitri N. Mavris, Director ASDL
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150
dimitri.mavris@ae.gatech.edu

Functional Analysis Techniques

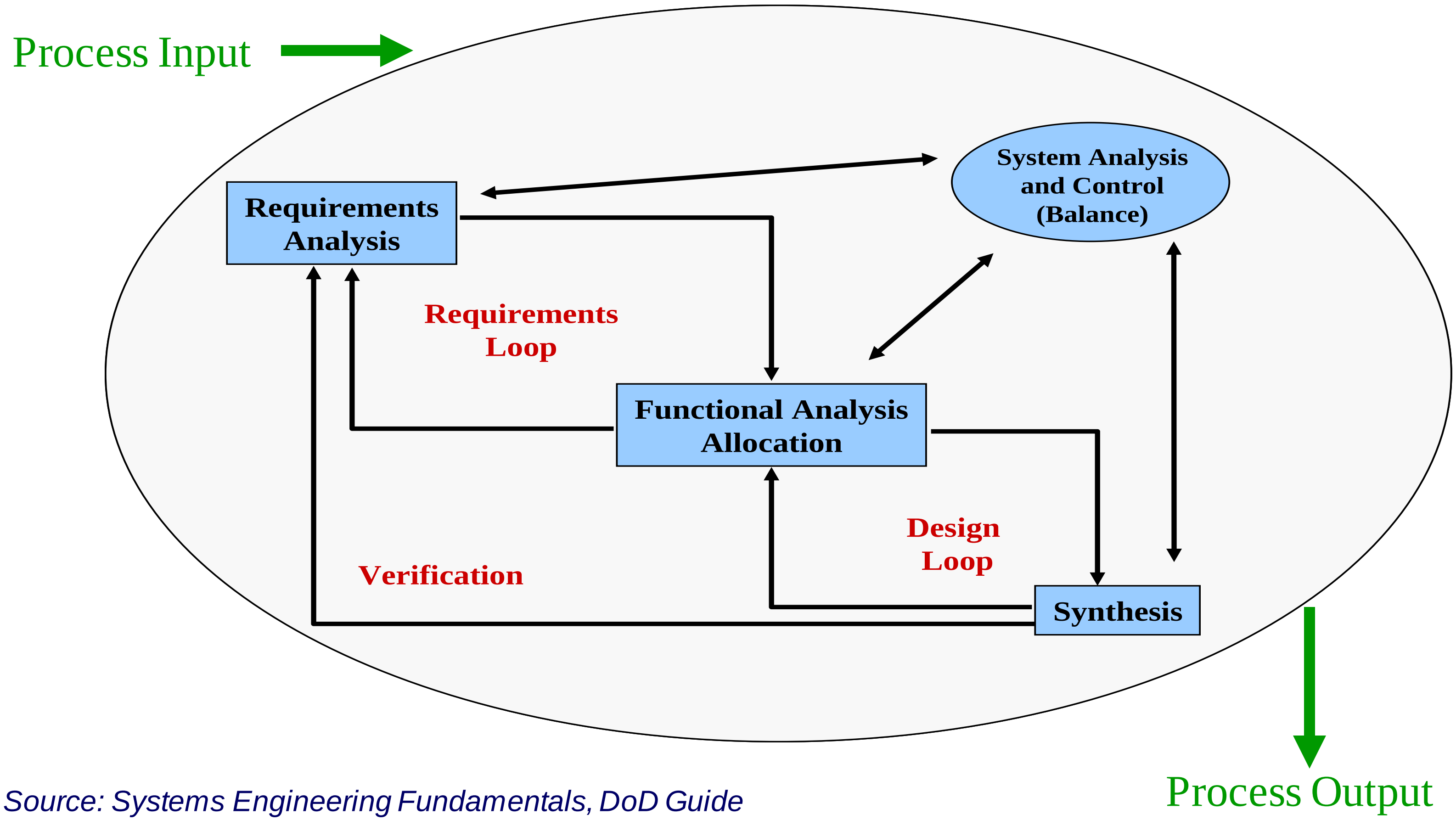
- **N2 Diagrams**
  - To develop system interfaces
  - The system components or functions are placed on the diagonal; the remainder of the squares in the N x N matrix represent the interface inputs and outputs



Example of an N2 Diagram [NASA SE Handbook]

What is Synthesis?

- Transform Architectures (Functional to Physical)
- Define Alternative System Concepts, Configuration Items & System Elements
- Select Preferred Product & Process Solutions
- Define/Refine Physical Interfaces (Internal/External)

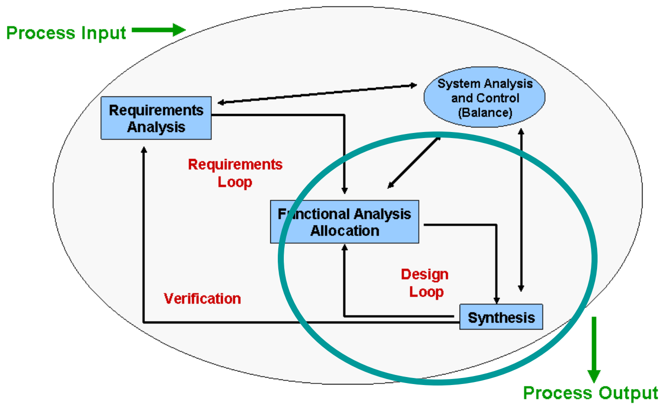


Source: Systems Engineering Fundamentals, DoD Guide

Dr. Dimitri N. Mavris, Director ASDL
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150
dimitri.mavris@ae.gatech.edu

Synthesis

Design synthesis is the process of defining the product or items in terms of the physical and software elements which together make up and define the item.



PHYSICAL ARCHITECTURE

- Each part must meet at least one
  - •Functional requirement
  - •Any part may support many functions
- Basic structure for generating specifications and baselines

The Design Loop

Similar to the requirements loop, it is the process of revisiting the functional architecture to verify that the physical design synthesized can perform the required functions at the required levels of performance

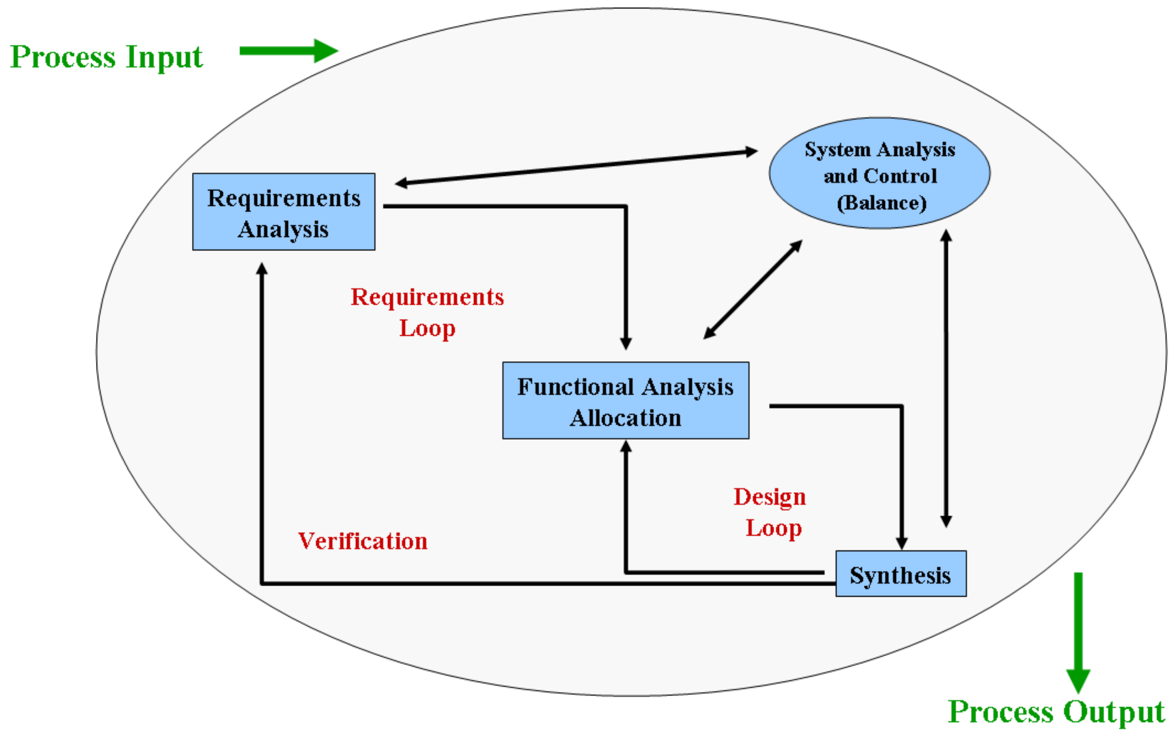
Source: Systems Engineering Fundamentals, DoD Guide

Verification

VERIFICATION is the process of comparing the solution to the requirements. This must occur at each level of development.

Appropriate methods of verification include:

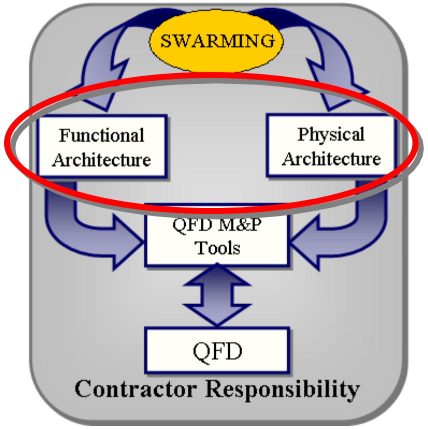
- *Examination*
- *Demonstration*
- *Analysis (including modelling and simulation)*
- *Testing*



Source: INCOSE Systems Engineering Handbook

System Architecture Definition

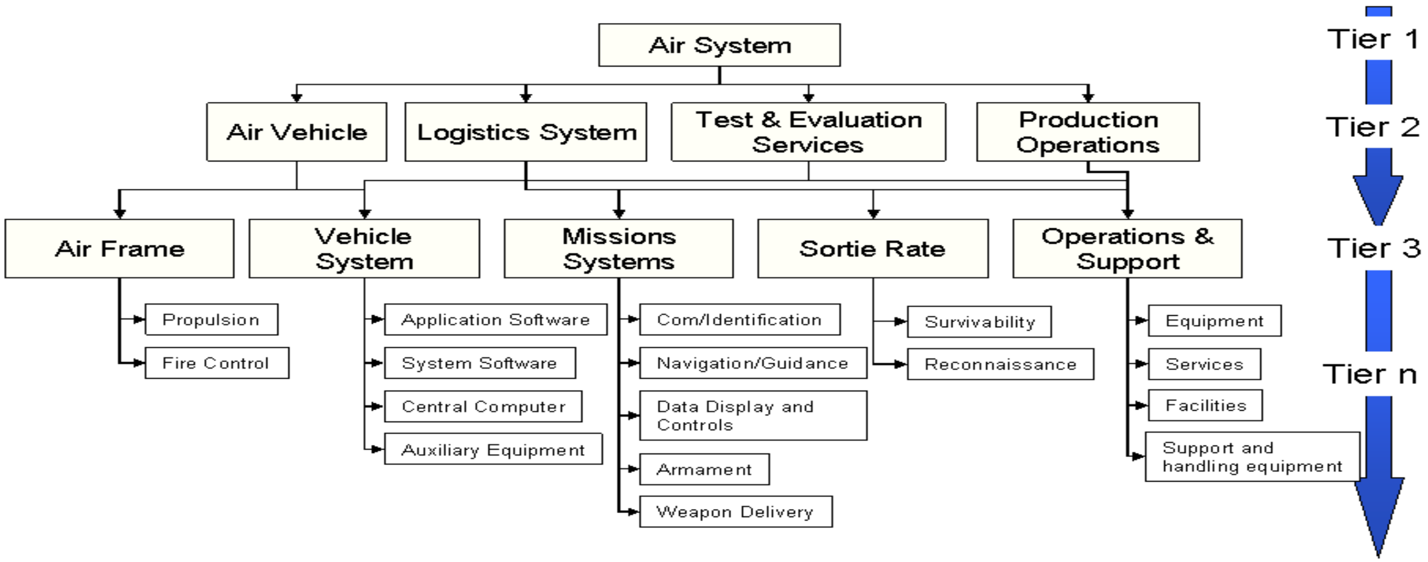
- The ***System Architecture*** identifies
  - All the products (including enabling products) that are necessary to support the system and, by implication, the processes necessary for development, production/construction, deployment, operations, support, disposal, training, and verification
- The elements can be either
  - **Functional** or **Physical**
  - Physical can be hardware, software, or both



Dr. Dimitri N. Mavris, Director ASDL
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150
dimitri.mavris@ae.gatech.edu

System Architecture Decomposition

- Systems can be decomposed **physically** as well as **functionally**
  - Decompose the systems and associated equipment required to accomplish functions
  - Indicate resources and tier level of system and equipment



Rule of Thumb:

Make sure system architecture is decomposed to accountable levels of lower tiers

Dr. Dimitri N. Mavris, Director ASDL
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150
dimitri.mavris@ae.gatech.edu

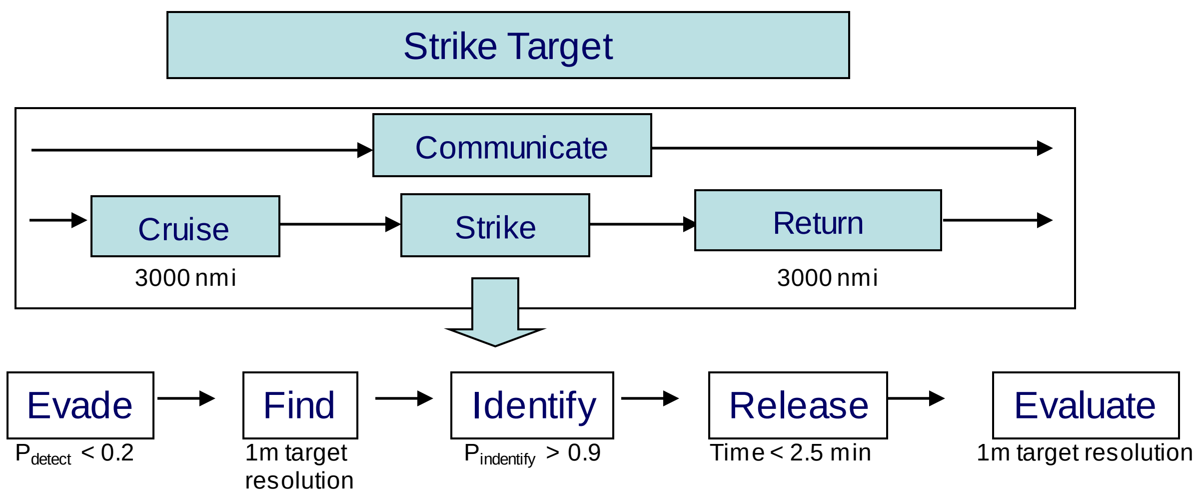
Notional Functional Decomposition

- Example of a possible functional description:
  - Indicate the basic requirements
  - Decompose the basic requirements into second level more specified requirements
  - Third level is used to allocate performance requirements to functions
- A simple rule: make sure that all functions are verbs

First Level:
Basic Functional Requirement

Second Level:
Cruise, Strike, Return

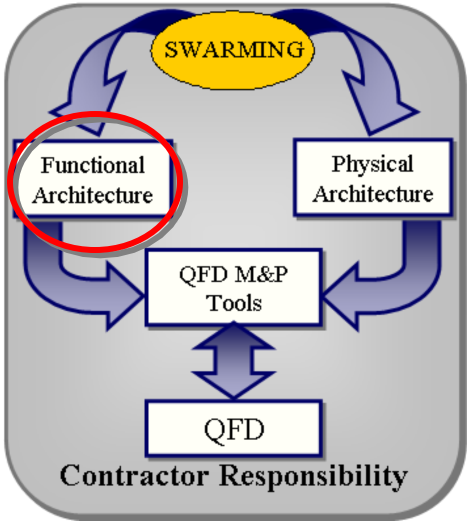
Third Level:
Showing Decomposition of the Strike function



Dr. Dimitri N. Mavris, Director ASDL
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150
dimitri.mavris@ae.gatech.edu

Functional Architecture Definitions

- The ***Functional Architecture*** identifies
  - the **functions** that have to be performed
  - the logical **sequencing** of the functions
  - the **performance requirements**
    - associated with the functions
  - structures the **allocated** requirements
- Engineering uses the functional architecture for specifying
  - subsystems and components
  - Interface requirements
- The functional architecture provides a stable layer of abstraction



Source: Department of Defense Systems Management College, System Engineering Fundamentals

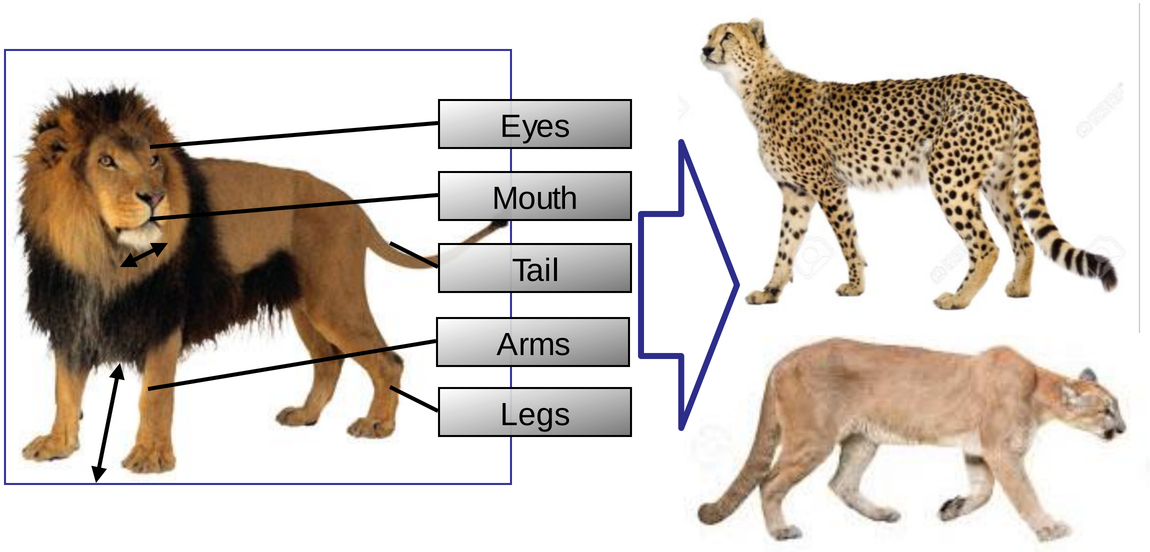
Defining a Functional Architecture

– **Analyze** system requirements to identify, derive, and define functional elements and interfaces
– **Define** functions the system must perform to achieve the desired outputs, and the interfaces between these functions
– **Define** the lower product level functions that the integrated system is to perform
– **Identify** alternative architectures which satisfy all functions
– Analyze and **integrate** the results of the requirements management, risk management, and system design optimization process to determine the preferred architecture
– **Document** the functions, their interfaces and the alternative functional architectures in accordance with established configuration management procedures
– **Define** derived requirements associated with each function and integrating them into the requirements management process

Dr. Dimitri N. Mavris, Director ASDL
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150
dimitri.mavris@ae.gatech.edu

Functional Breakdown

- Systems participating in similar functions
- Advantages
  - Consistent across architectures
  - Intuitive for quantifying requirements
  - Freedom for revolutionary concepts

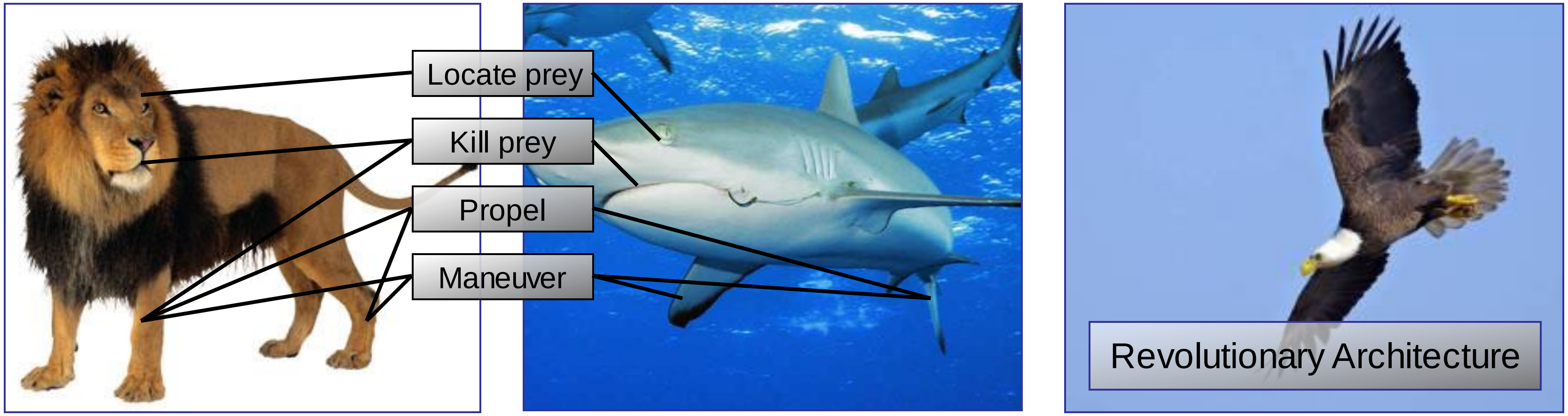


Example of physical breakdown

Dr. Dimitri N. Mavris, Director ASDL
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150
dimitri.mavris@ae.gatech.edu

Functional Breakdown

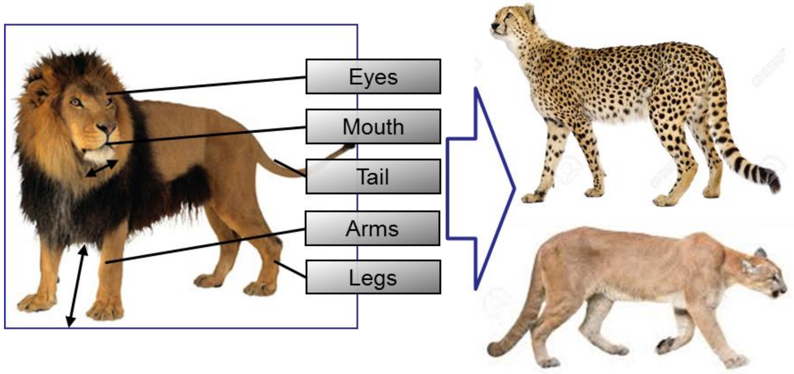
- Systems participating in similar functions
- Advantages
  - Consistent across architectures
  - Intuitive for quantifying requirements
  - Freedom for revolutionary concepts



- Disadvantages
  - Each functional group can have diverse physical implementations
  - Several functions may be served by the same system

Dr. Dimitri N. Mavris, Director ASDL
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150
dimitri.mavris@ae.gatech.edu

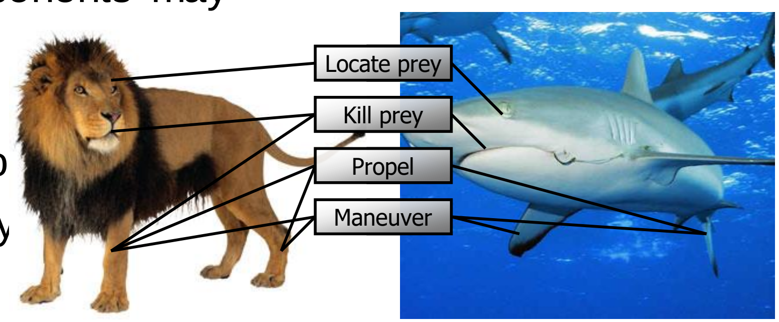
Functional vs. physical approach



A functional approach is advantageous for architecture design and analysis

- **Disadvantages of a physical breakdown**
  - Each decomposition is specific to an architecture
  - Analogous physical components do not always serve the same function
  - Analogous physical components may not exist
  - Comparing the characteristics of analogo physical components may be meaningless
  - Example: ATA Chapters

- **Advantages of a functional breakdown**
  - Universally consistent across architectures
  - Intuitive for quantifying requirements
  - Freedom for revolutionary concepts



Dr. Dimitri N. Mavris, Director ASDL
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150
dimitri.mavris@ae.gatech.edu