Role and figure of the Quantitative Systems Engineer

A fairy tale

about engineering statistics applications within system life-cycle

in order to extract its value.

freely inspired by Peter and the Wolf

Contents

- *If words mean something*
- *Role*
- *Challenges*
- *Development Process walkthrough*
    - *User needs*
    - *Systems Requirements*
    - *System Architecture*
    - *Sub-system/component design*
    - *Component testing*
    - *System integration*
    - *System verification*
    - *System validation*
    - *System Maintenance*
    - *System Disposal*
- *Key Messages*

If words mean something…

.

Quantitative

Drives the “good enough” trade-offs in an un-certain environment by

Engineering statistics alias the integration of statistical and engineering analysis concepts to

Systems

Holystic view with passion and skill for key elements and interfaces

Engineer

Design and manage complexity

Incose SE twelve roles

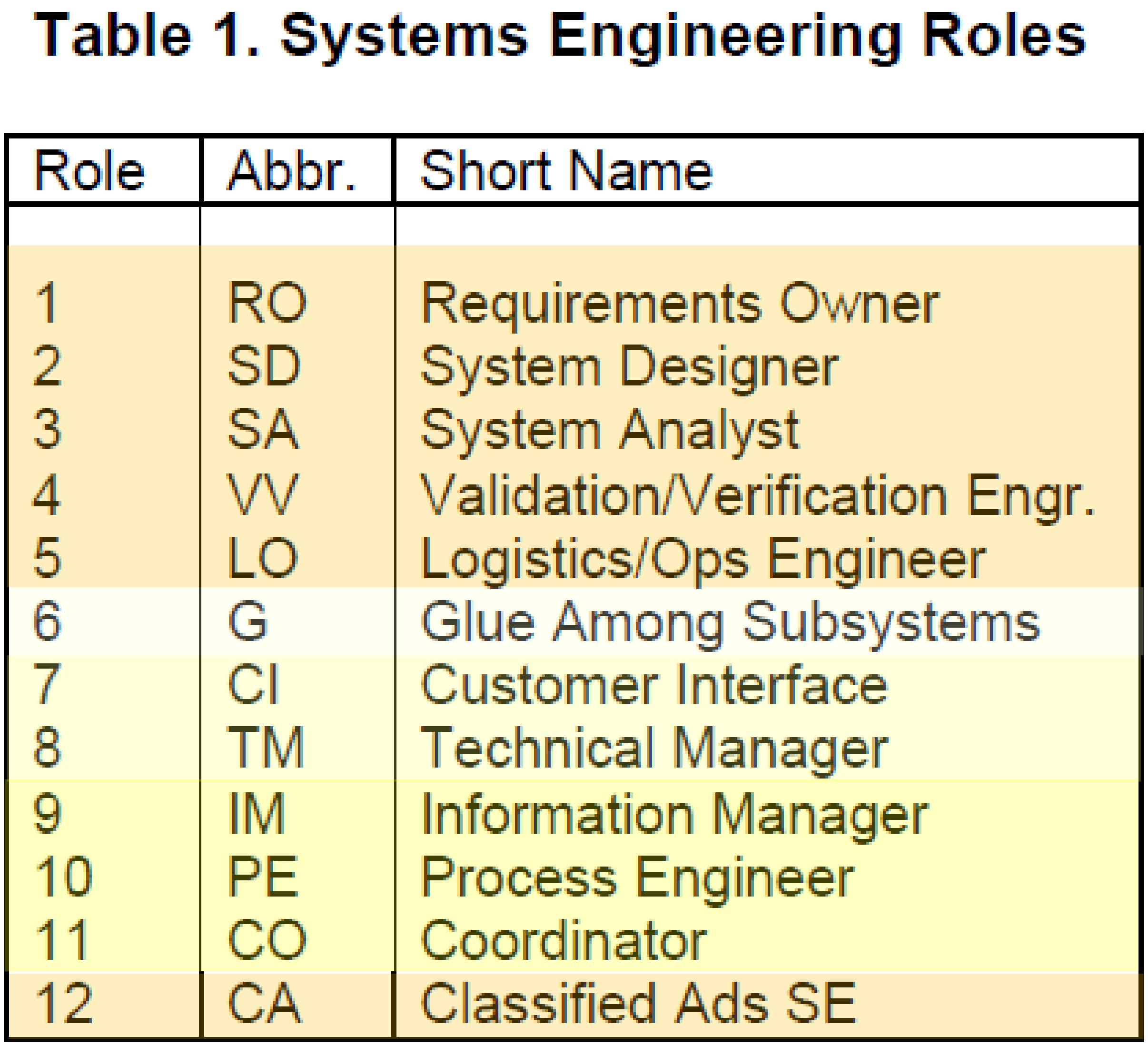


Table 1. Systems Engineering Roles

Role	Abbr.	Short Name
1	RO	Requirements Owner
2	SD	System Designer
3	SA	System Analyst
4	VV	Validation/Verification Engr.
5	LO	Logistics/Ops Engineer
6	G	Glue Among Subsystems
7	CI	Customer Interface
8	TM	Technical Manager
9	IM	Information Manager
10	PE	Process Engineer
11	CO	Coordinator
12	CA	Classified Ads SE

Adapted from “**TWELVE SYSTEMS ENGINEERING ROLES”** Sarah A. Sheard

Role

The Quantitative Systems Engineer is an **engineer**
with **specific skills** and an **holistic view**
that fosters
the system **value**
through the **key decision points**
of the overall system life-cycle
by applying:
the **engineering statistics skills**
and **experience**

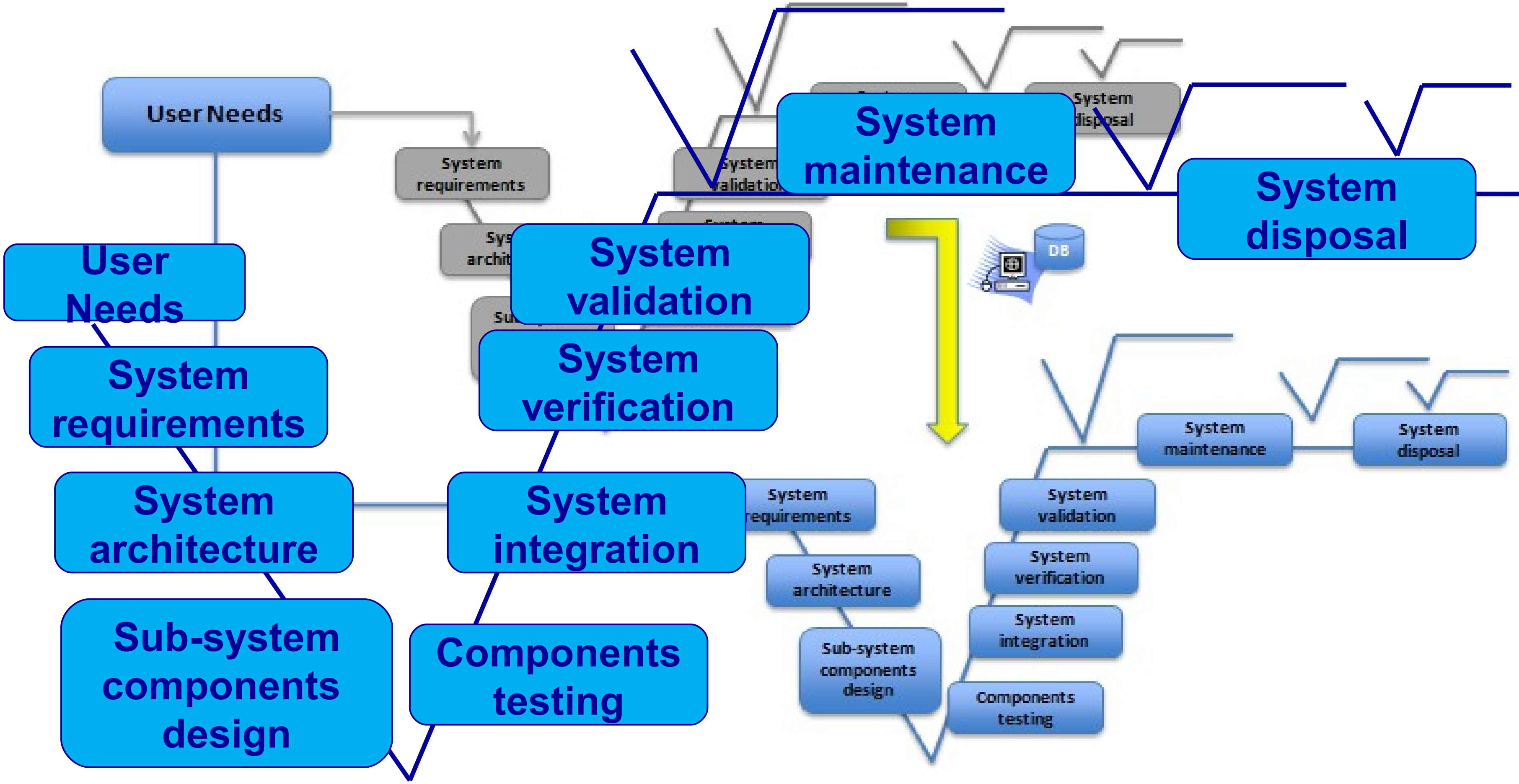
Challenges:

Uncertainty & Variability

Holistic & detailed mindset

Good enough & Value

Tasks overview:
a development process walkthrough



User Needs

System requirements

System architecture

Sub-system components design

Components testing

System integration

System verification

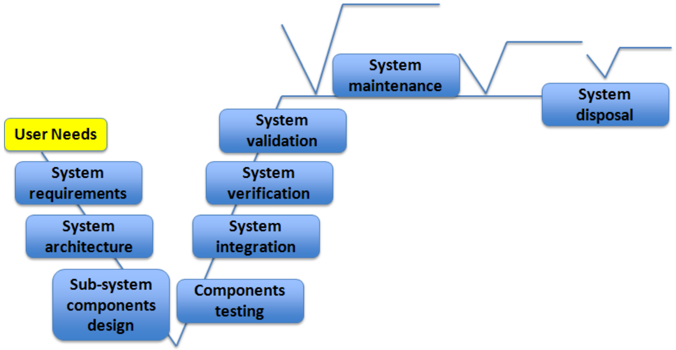
System validation

System maintenance

System disposal

Tasks overview: *User needs*

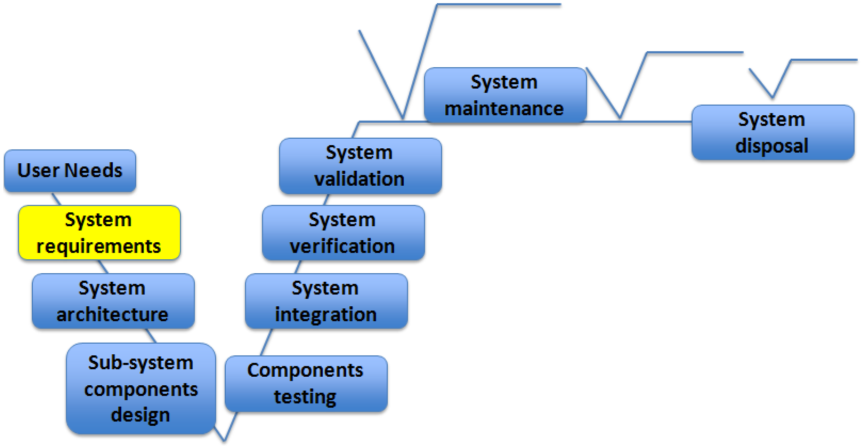
Collect, identify and refine the **user needs**, understand **project challenges**, identify **usage scenario, characterize human factors:**



- Market trends analysis and customers appraisals: *qualitative/quantitative analysis, Kano models, QFD*
- *KPIs target definition: e.g. Cost targets, Ppk, OEE*
- User needs quality assessment: *testable targets*
- Usage scenario: *Multivariate Data Analysis*
- *Set-up Validation strategy and plan*

Tasks overview: System requirements

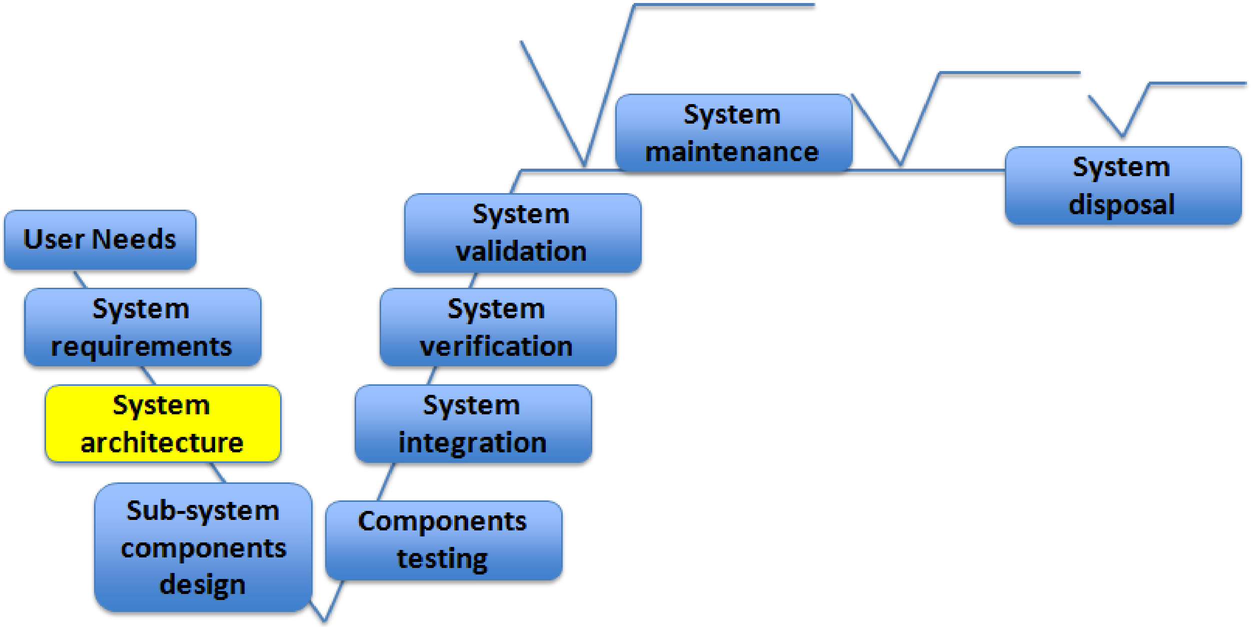
Define and qualitatively assess the System Requirements:



- Targets definition: *Inferential statistics, TPMs: TBF, TTR, binomial, poisson, and normal*
- Evaluate the technical opportunities: *ANoVA, inference*
- Verification strategy: *Combine practical statistical methodologies with technical and project challenges to determine:* ***what, how much and how to test.***

Tasks overview: System architecture

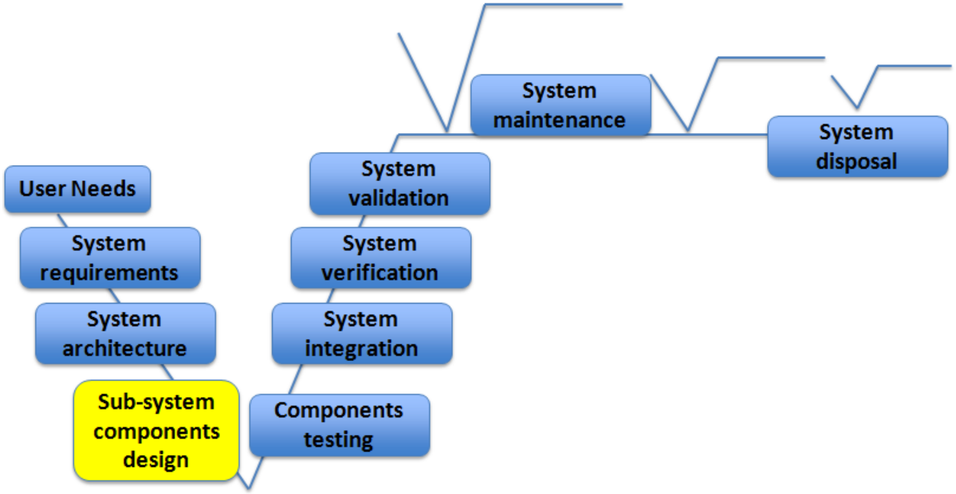
System modelling and understanding:



- System modelling: *Simulation, Sysml*
- Risk assessment: FMECA, *Fault Tree Analysis*
- Evaluate the technical opportunities: *Simulation and inference.*

Tasks overview:
Sub-systems/component design

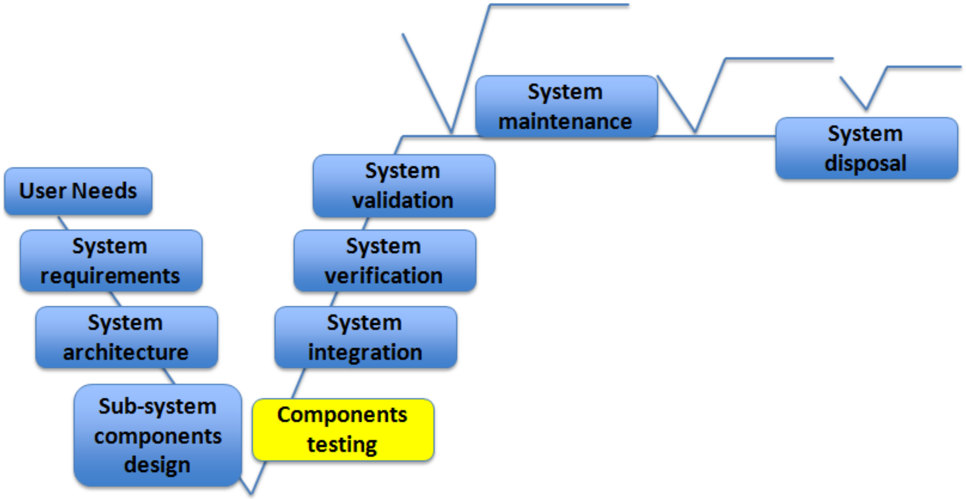
Integration of traditional engineering and statistical analysis:



- Alternatives selection: ANoVA, *Design of Experiments, non-parametric statistics*
- Sub-systems pre-assessment: ***Simulation**, preliminary testing.*
- **Robust Design**: *DoE, noises, loss functions*
- Components reliability: *ALT, stress tests*

Tasks overview: components testing

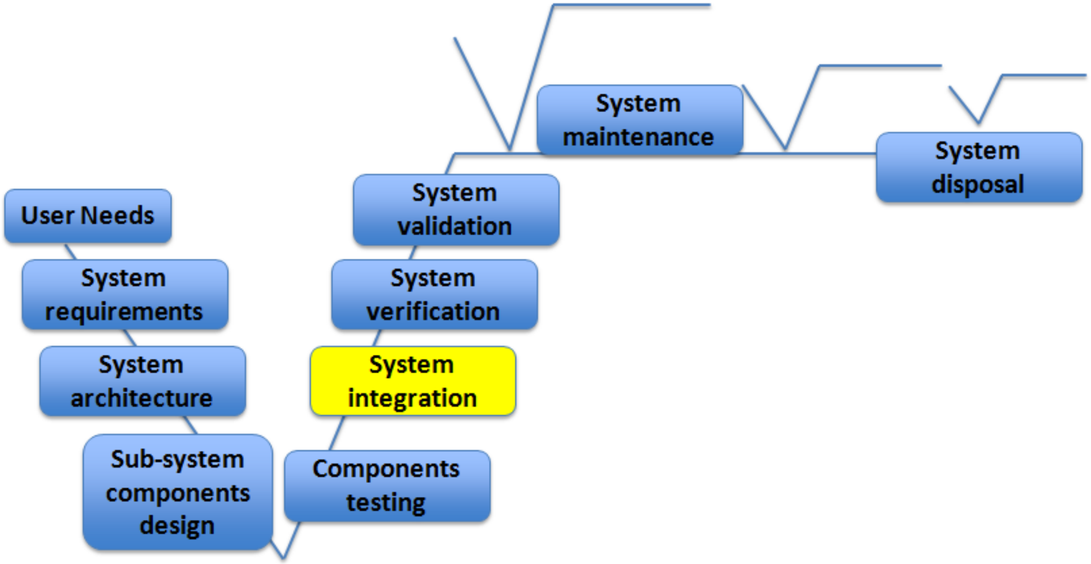
Integration of traditional engineering and statistical analysis:



- Components assessment: *HW/SW black and white box testing*
- Technical Measures assessment: *ANoVA, Design of Experiments, comparison tests.*

Tasks overview: System Integration

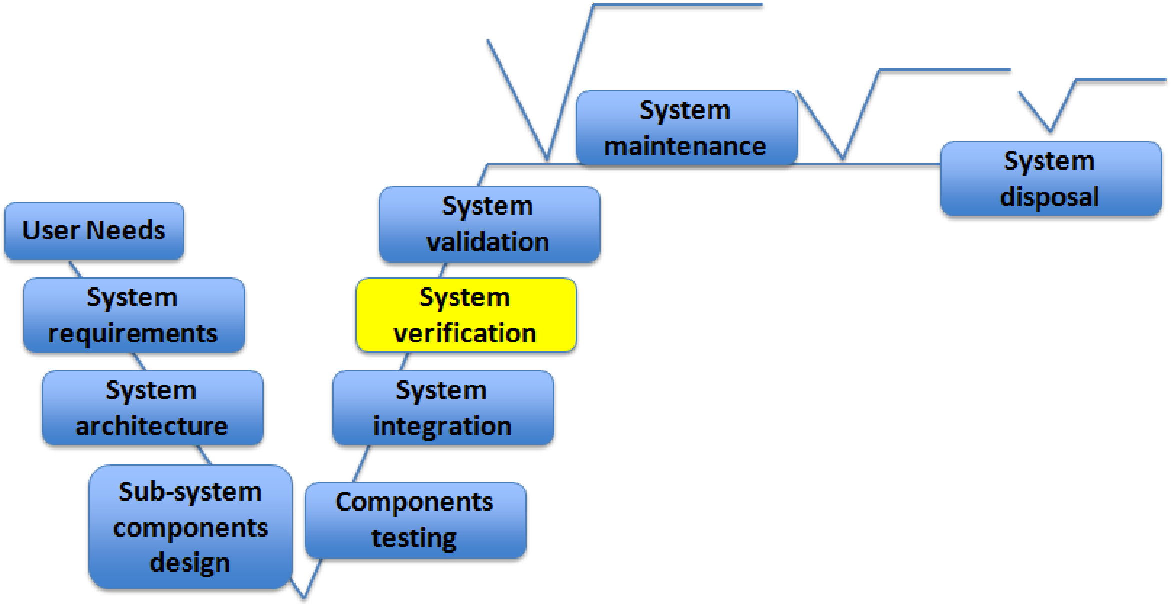
Integrate the system components/sub-systems:



- **Interface** assessment: *Comparative tests, Statistical inference.*
- Technical Measures assessment: preliminary testing, *SPC, inferential and non-parametric statistics.*
- ***Issue** resolution: PDCA cycles, root cause analysis.*

Tasks overview: System verification

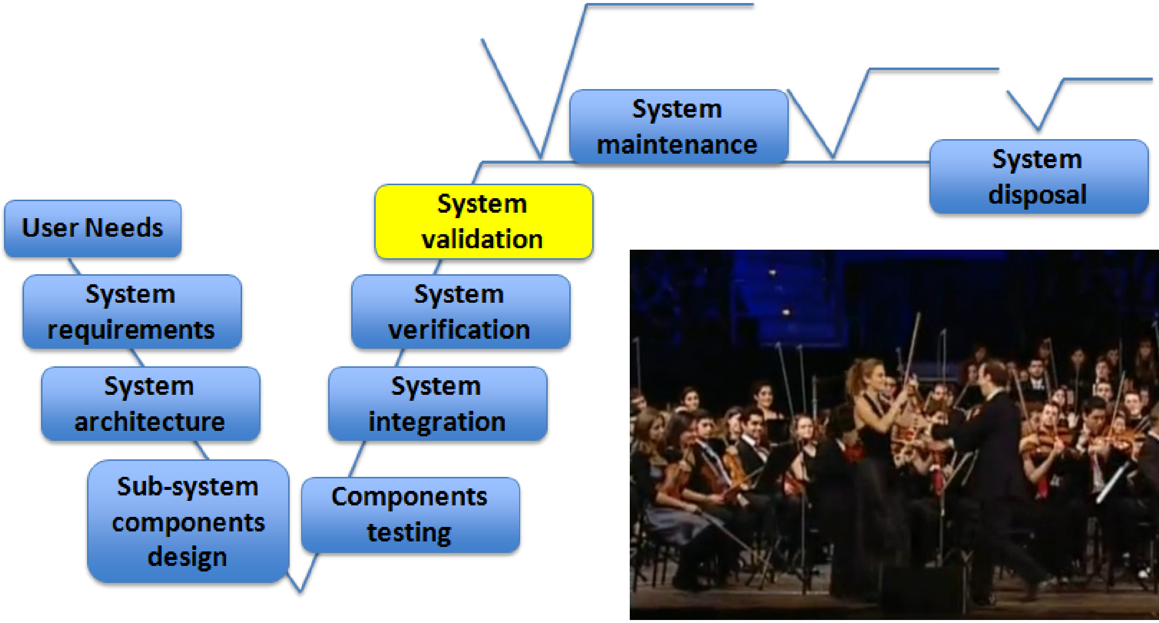
Is the system right?



- Update and deploy verification plans: *Inferential statistic*
- **Un-expected events**: *risk assessment update, risk analysis, bayesian statistic*
- *FMECA re-evaluation*

Tasks overview: System validation

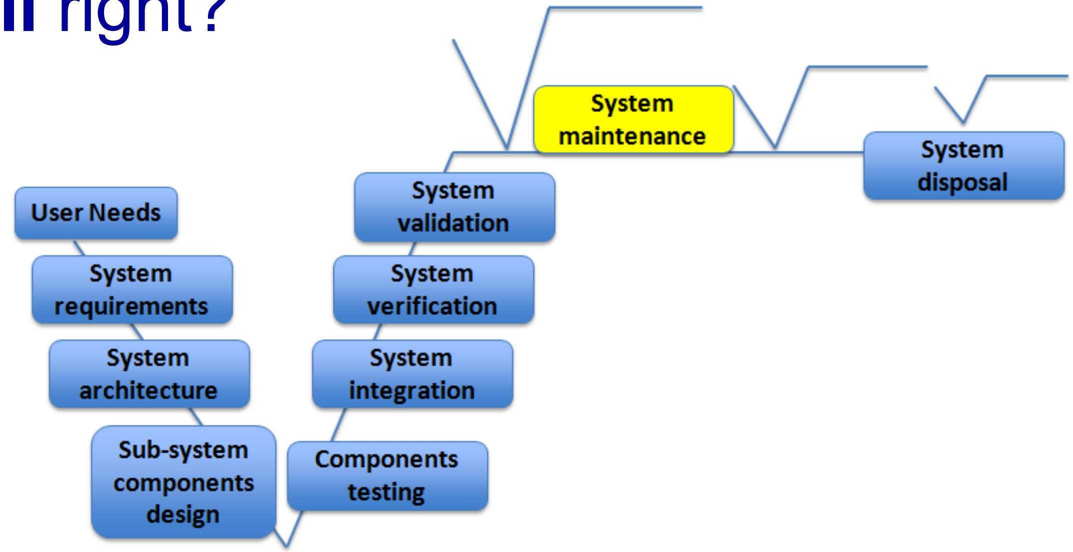
Is the right system?



- Update and deploy validation plans: *Inferential statistic, risk assessment and risk analysis*
- *Un-expected events: root cause analysis, risk assessment and risk analysis, bayesian statistic.*
- ***Value analysis update***

Tasks overview: System maintenance

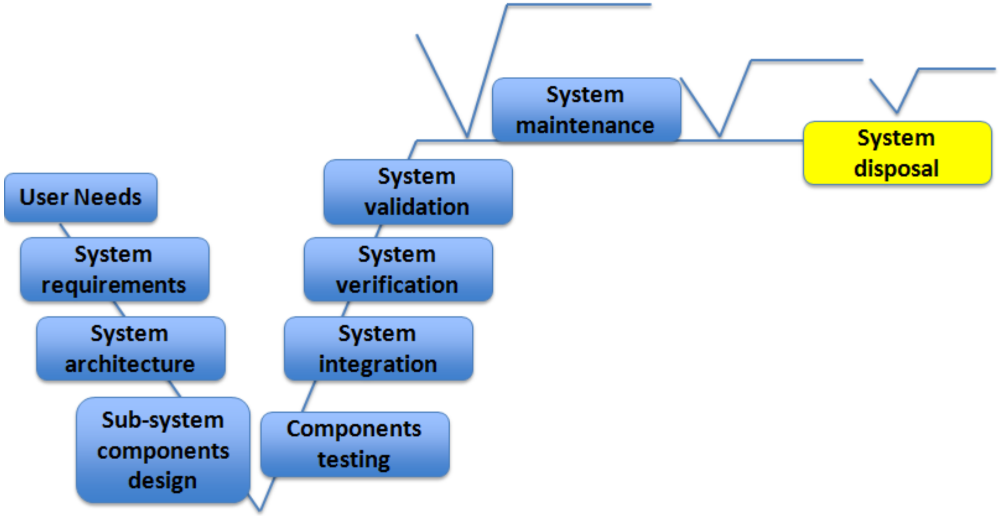
Is the system **still** right?



- Quality checks: *SPC, reliability data analysis*
- Issues/problems/crisis: *Root cause analysis, Design of Experiments, risk re-assessment and analysis*
- Continuous improvement: *Value analysis update*

Tasks overview: System disposal

Was the system valuable? What can we learn?



- Historical data analysis: SPC, Multivariate Data Analysis, data mining.
- Value analysis: Qualitative/quantitative analysis
- ***Lessons learned**: track product success/failures history*

Key messages: knowledge and value

- A relevant part of system knowledge and value is enclosed into data.
- There are methodologies, tools and capabilities to consolidate and to extract it.
- The Quantitative System Engineer can deal this task during and after the overall system life-cycle.