

AI and Cause-Effect Chains – Can QFD Help?



Thomas M. Fehlmann, Zürich
Eberhard Kranich, Duisburg
Euro Project Office AG

E: info@e-p-o.com

H: www.e-p-o.com



QFD Café

**euro project office**
Master your project.

Customer
Orientation

Lean
Six Sigma

Agile
Processes

Project
Estimations

Transfer
Functions

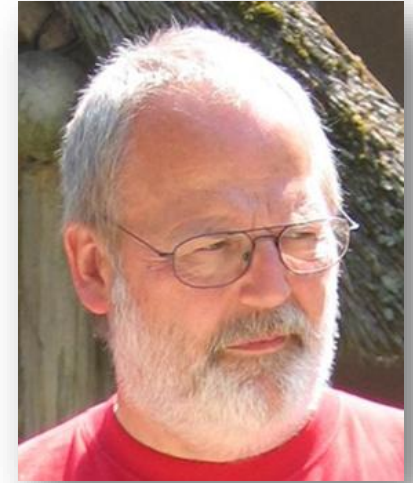
Speaker & Authors



- Dr. Math ETHZ
- Quality Function Deployment (QFD)
- Six Sigma for Software
- Testing Complex Systems
- Representing Knowledge in Artificial Intelligence
- Measuring Software & Knowledge
- Quality Management for Product Development

Thomas M. Fehlmann
Zürich

- Mathematics and Computer Science
- Emphasis on Mathematical Statistics
- Mathematical Optimization
- Six Sigma Black Belt for Software Development
- Software Quality Assurance Manager



Eberhard Kranich
Duisburg

Customer Orientation

Lean Six Sigma

Agile Processes

Project Estimations

Transfer Functions

Goals of this Presentation

- 1) *QFD is not a thing of the past but essential for AI*
- 2) *Motivate you to grow from a QFD expert to become an AI expert*
- 3) *Looking at history of AI and QFD: Same origins, same goals, similar hurdles*



*"In God we Trust,
all others must bring data"*

(Edward Deming, Grandfather of Lean)

Customer
Orientation



QFD as Algebra of Combinators

Lean
Six Sigma



AI as Algebra of Combinators

Agile
Processes



Using QFD for Agentic AI

Project
Estimations



Conclusions

Transfer
Functions



QFD as Algebra of Combinators



AI as Algebra of Combinators



Using QFD for Agentic AI



Conclusions

Level
Six Sigma

Agile
Processes

Project
Estimations

Transfer
Functions

Ishikawa-Diagrams and the House of Quality (HoQ)

$$(b_i \rightarrow a)_j$$

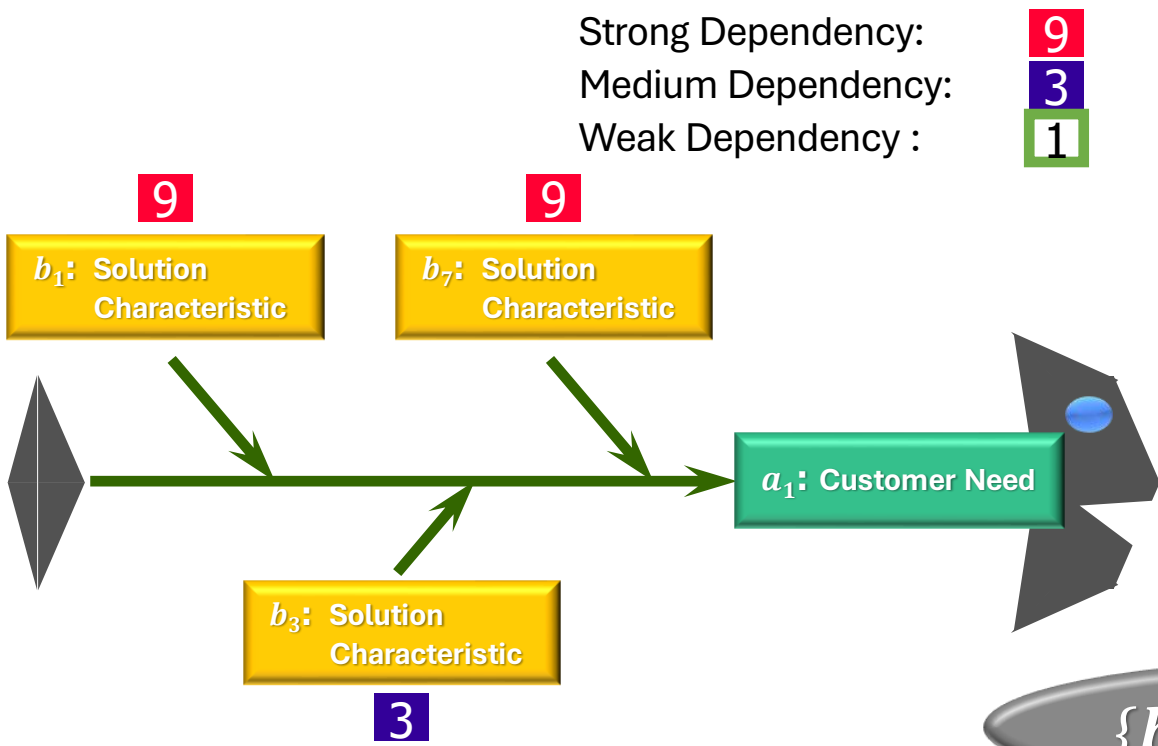
Customer Orientation

Lean Six Sigma

Agile Processes

Project Estimations

Transfer Functions



a_j : Customer Needs

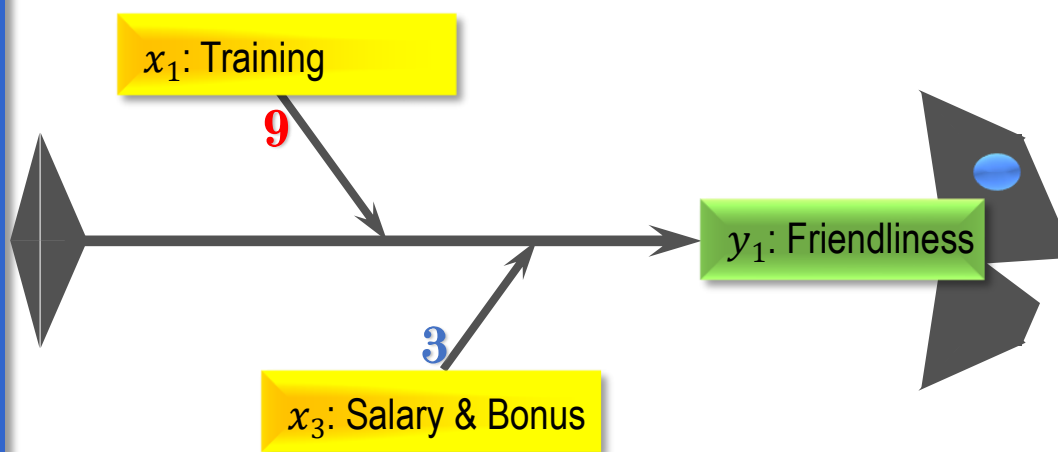
a_1 : Customer Need	9		3				9
a_2 : Customer Need		9			3		
a_3 : Customer Need	9			3		3	3
a_4 : Customer Need		3	9		1		
a_5 : Customer Need				1	3		9
a_6 : Customer Need		1		3		3	
a_7 : Customer Need					3		1
	b_1 : Solution Characteristic	b_2 : Solution Characteristic	b_3 : Solution Characteristic	b_4 : Solution Characteristic	b_5 : Solution Characteristic	b_6 : Solution Characteristic	b_7 : Solution Characteristic

b_i : Solution Characteristics

$$\{b_1, b_3, b_7\} \rightarrow a_1$$

Example: Improving Call Center Services with Classic QFD

- Ishikawa diagrams for identifying cause and effect
 - ➔ Expected response should satisfy customer's needs
- With many causes and effects
 - ➔ Matrix is handier
 - ➔ Quality Function Deployment (QFD)



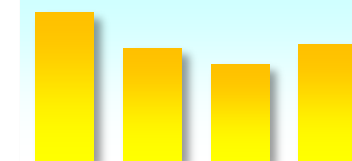
Critical To Quality Deployment Combinator

Critical To Quality

		Goal Profile	Critical To Quality				
			x1 Training	x2 ICT Infrastructure	x3 Salary & Bonus	x4 Work Place	
Expected Response							
y1	Friendliness	0.69	9		3		
y2	Responsiveness	0.56		9		7	
y3	Accuracy	0.45	1		5	3	

Solution Profile for Critical To Quality: 0.62 0.47 0.40 0.49

37 Total Effort Points
0.10 Convergence Range
0.20 Convergence Limit



Customer Orientation

Lean Six Sigma

Agile Processes

Project Estimations

Transfer Functions

Example: Improving Call Center Services with Classic QFD

- Ishikawa diagram for cause and effect
 - ➔ Expected response customer's need
- With many causes
 - ➔ Matrix is handier
 - ➔ Quality Function Deployment

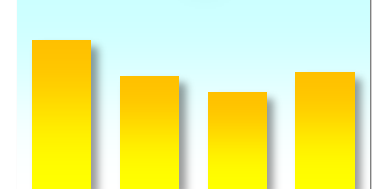


Critical To Quality

	x1 Training	x2 ICT Infrastructure	x3 Salary & Bonus	x4 Work Place
Accuracy	9		3	
Service		9		7
Accuracy	1		5	3

Contribution Profile for Critical To Quality: 0.62 0.47 0.40 0.49

37 Total Effort Points
0.10 Convergence Range
0.20 Convergence Limit



x₁: Training

9

y₁: Friendliness

3

x₃: Salary & Bonus

Customer Orientation

Lean Six Sigma

Agile Processes

Project Estimations

Transfer Functions

The Nineties – Bad Mathematics

Customer Orientation

Lean Six Sigma

Agile Processes

Project Estimations

Transfer Functions

- How to make sure that the identified solution profile delivers the expected response?
 - ➔ Mathematically speaking, $x = A^T y$ is not a solution for $y = Ax$
 - ➔ This is how people used to interpret QFD matrices in the early times
- The Convergence Gap identifies the difference between expected response and achieved response

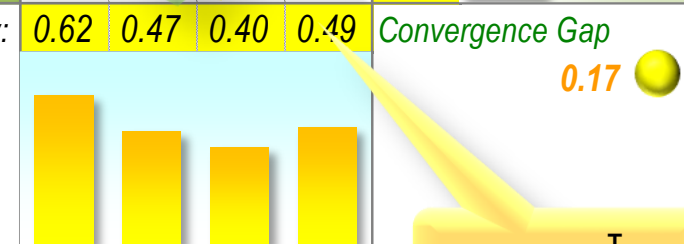
➔
$$\|y - Ax\| = \sqrt{\sum (y_j - \sum a_{ij}x_i)^2}$$

Critical To Quality Deployment Combinator

Critical To Quality

		Goal Profile	x1 Training	x2 ICT Infrastructure	x3 Salary & Bonus	x4 Work Place	Achieved Profile
<i>Expected Response</i>							
y1	Friendliness	0.69	9		3		0.62
y2	Responsiveness	0.56		4		7	0.69
y3	Accuracy	0.45	1		5	3	0.37

Solution Profile for Critical To Quality:



37 Total Effort Points
0.10 Convergence Range
0.20 Convergence Limit

$y \neq Ax$

$x = A^T y$

Solution with Eigenvectors

Customer Orientation

Lean Six Sigma

Agile Processes

Project Estimations

Transfer Functions

- How to make sure that the identified solution profile delivers the expected response?
 - ➔ Mathematically speaking, $x = A^T y$ is not a solution for $y = Ax$
 - ➔ This is how people used to interpret QFD matrices in the early times
- Close the Convergence Gap by shifting focus from Salary & Bonus to more accurate ICT Infrastructure
 - ➔ Eigenvector method yields solutions that are mathematically sound

*Critical To Quality
Deployment Combinator*

y

Goal Profile

Critical To Quality

$y \cong Ax$

Expected Response

		x1 Training	x2 ICT Infrastructure	x3 Salary & Bonus	x4 Work Place	
y1	Friendliness	0.69	9	2		0.69
y2	Responsiveness	0.56		7	5	0.58
y3	Accuracy	0.45	1	2	4	0.43

Achieved Profile

Solution Profile for Critical To Quality:

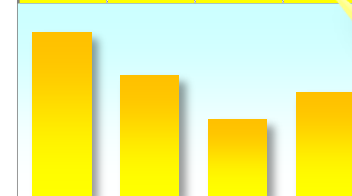
0.68 0.50 0.32 0.43

Convergence Gap 0.03

33 Total Effort Points

0.10 Convergence Range

0.20 Convergence Limit



$x = A^T y$

y_E is the Eigenvector of AA^T

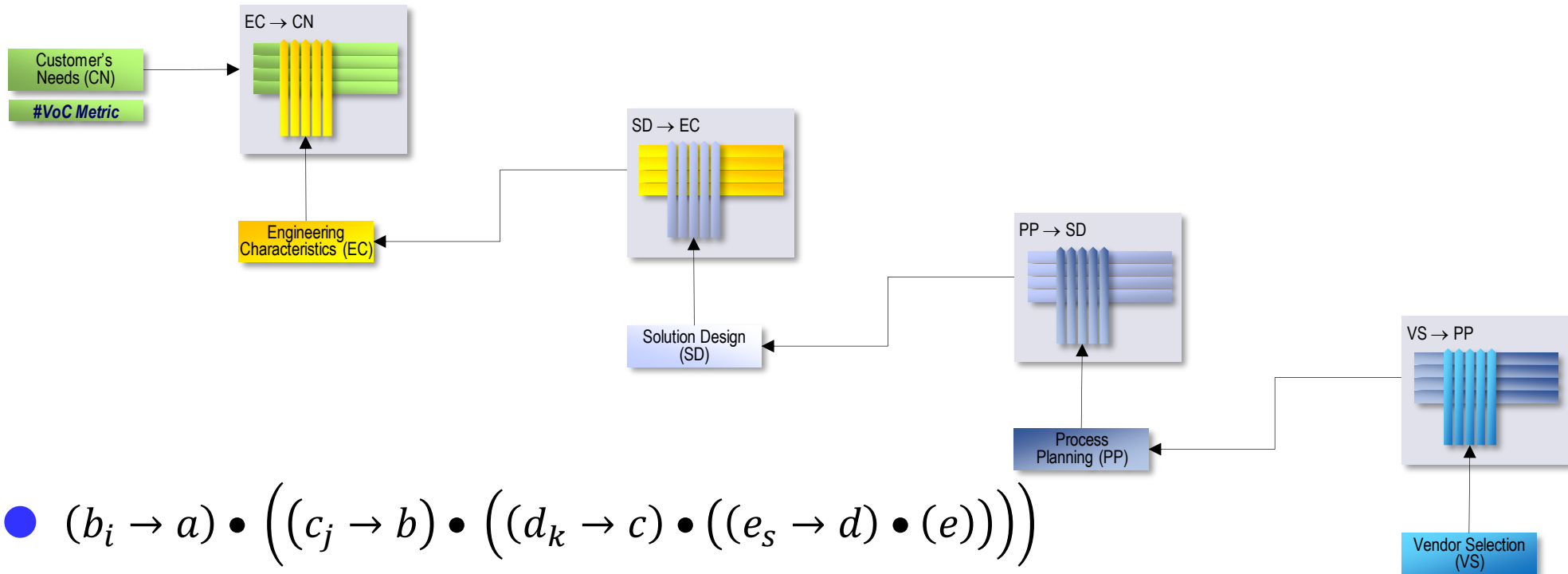
Customer Orientation

Lean Six Sigma

Agile Processes

Project Estimations

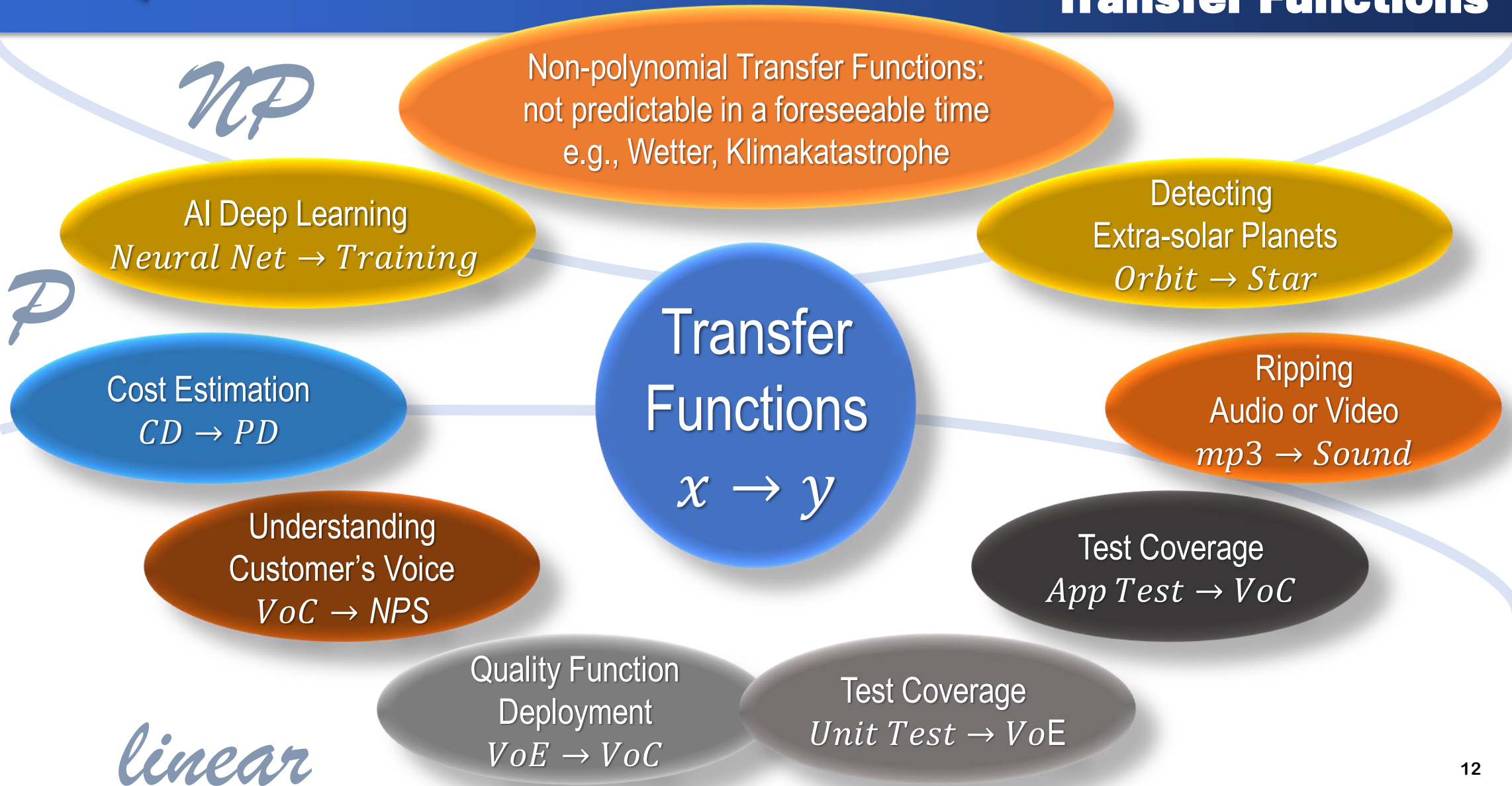
Transfer Functions



- $$(b_i \rightarrow a) \bullet \left((c_j \rightarrow b) \bullet \left((d_k \rightarrow c) \bullet \left((e_s \rightarrow d) \bullet (e) \right) \right) \right)$$

- ➔ $(b_i \rightarrow a)$ is the House Of Quality
- ➔ $(c_j \rightarrow b)$ the Solution Characteristics deployment
- ➔ $(d_k \rightarrow c)$ the Methods & Tools selection
- ➔ $(e_s \rightarrow d)$ the Vendor Selection

Transfer Functions



Customer Orientation

Lean Six Sigma

Agile Processes

Project Estimations

Transfer Functions

The Universal World Formula

The query we want to know

Our system controls

$$y = Ax$$

Our knowledge base, e.g., the Neural Network to be trained

The response

Ax

Customer Orientation

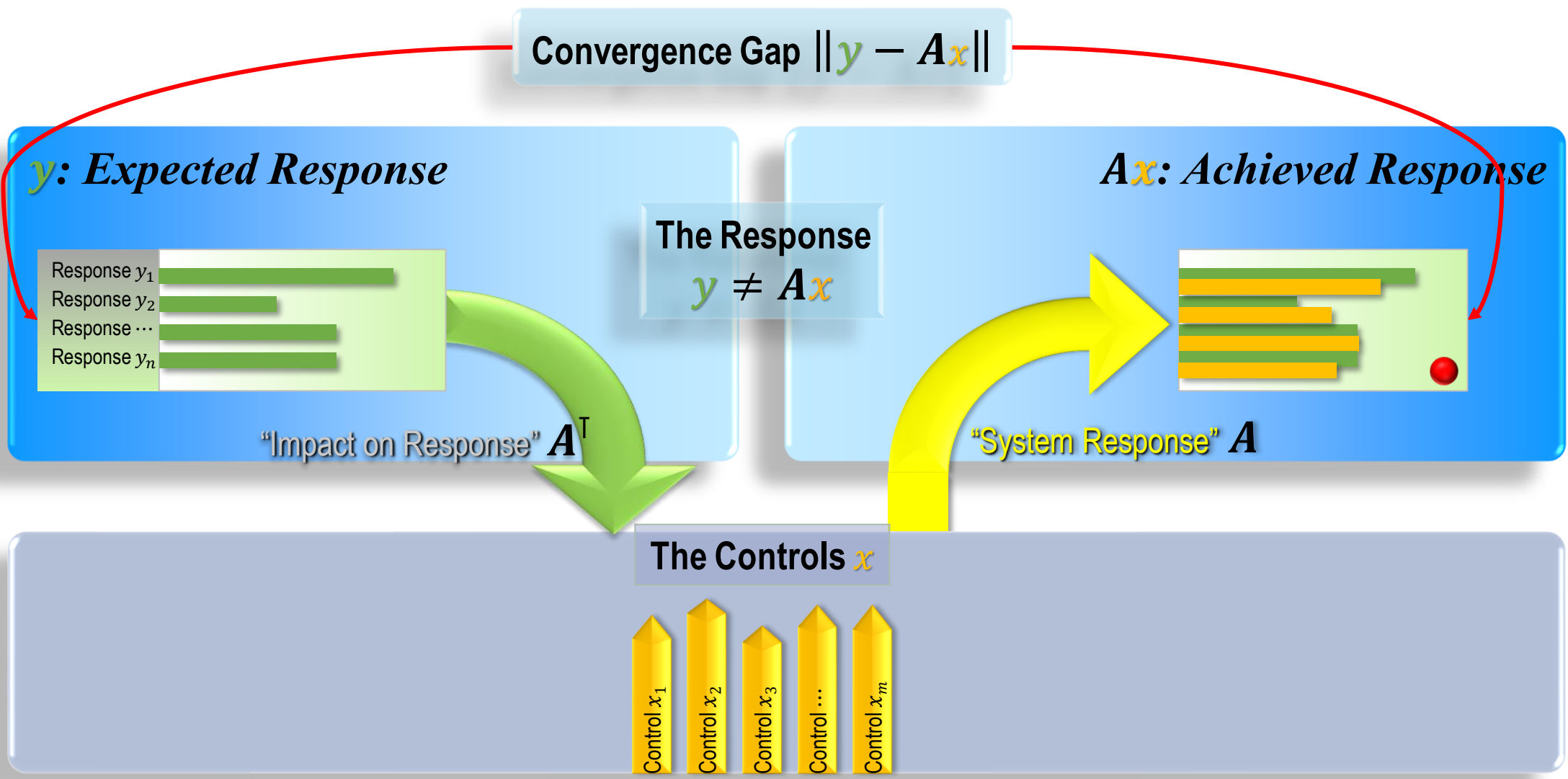
Lean Six Sigma

Agile Processes

Project Estimations

Transfer Functions

Transfer Functions – Measuring A or Keep A Fixed



Customer Orientation

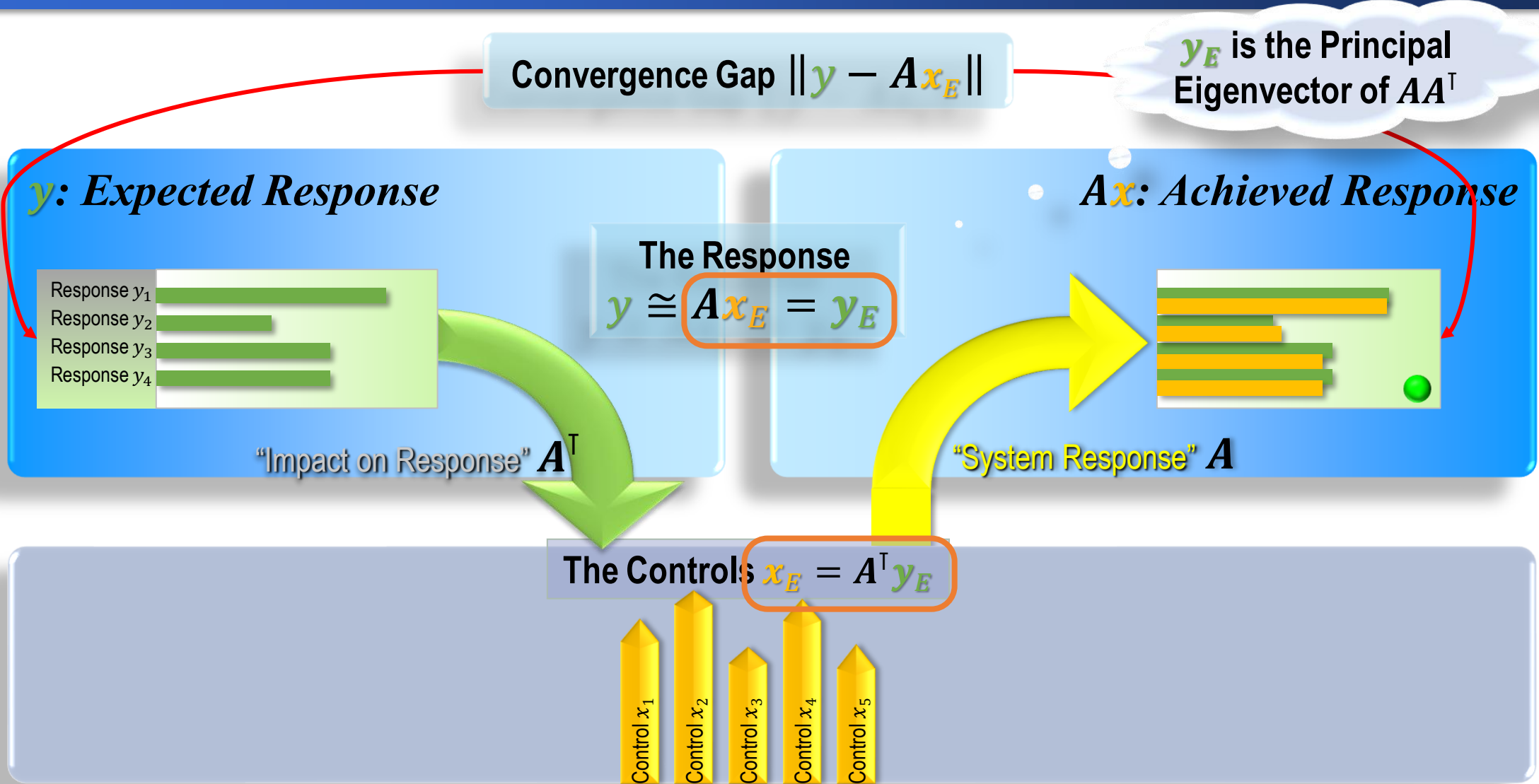
Lean Six Sigma

Agile Processes

Project Estimations

Transfer Functions

Transfer Function – The Convergence Gap



Customer Orientation

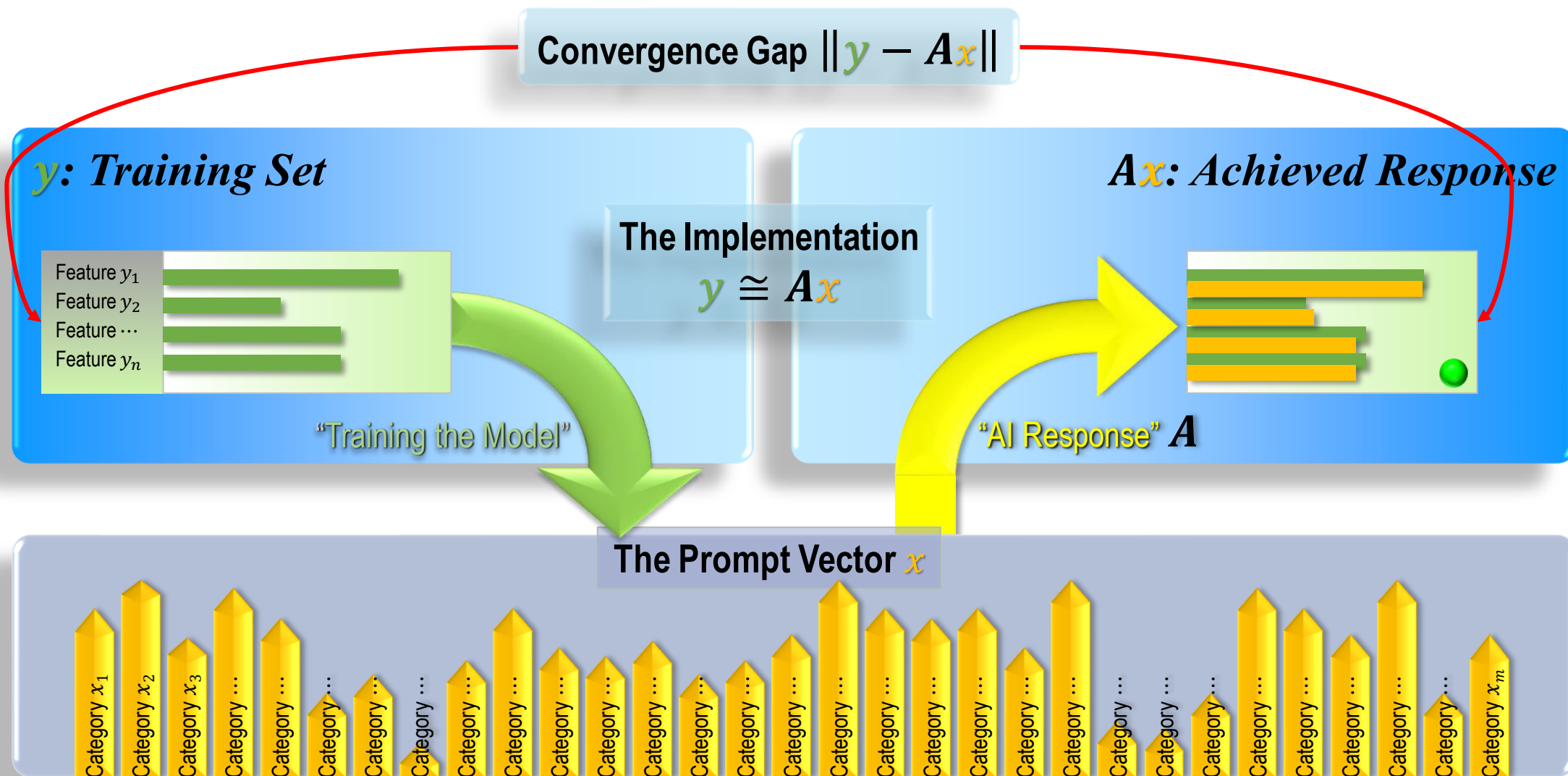
Lean Six Sigma

Agile Processes

Project Estimations

Transfer Functions

Deep Learning in AI – Adapt A to a Training Set of Input Vectors



Customer Orientation

Lean Six Sigma

Agile Processes

Project Estimations

Transfer Functions

Customer
Orientation



QFD as Algebra of Combinators



AI as Algebra of Combinators

Pro

Project
Estimations



Using QFD for Agentic AI

Transfer
Functions



Conclusions

Counting Frequencies

- **Large Language Models (LLM)**
 - ➔ Generative Pretrained Translator
 - ➔ Pretrained by counting words and their sequence
 - It understand nothing
 - ➔ 80° Temperature
 - Means that it selects randomly among the 20% best matches
 - Otherwise, LLM become boring

- LLMs can be used to provide a flawless, elegant English
 - ➔ And a few other languages as well, including Kölsch and even Schwyzertüüsch
 - ➔ "Daach zosamme!" or «Guete Morge zäme»



Customer
Orientation

Lean
Six Sigma

Agile
Processes

Project
Estimations

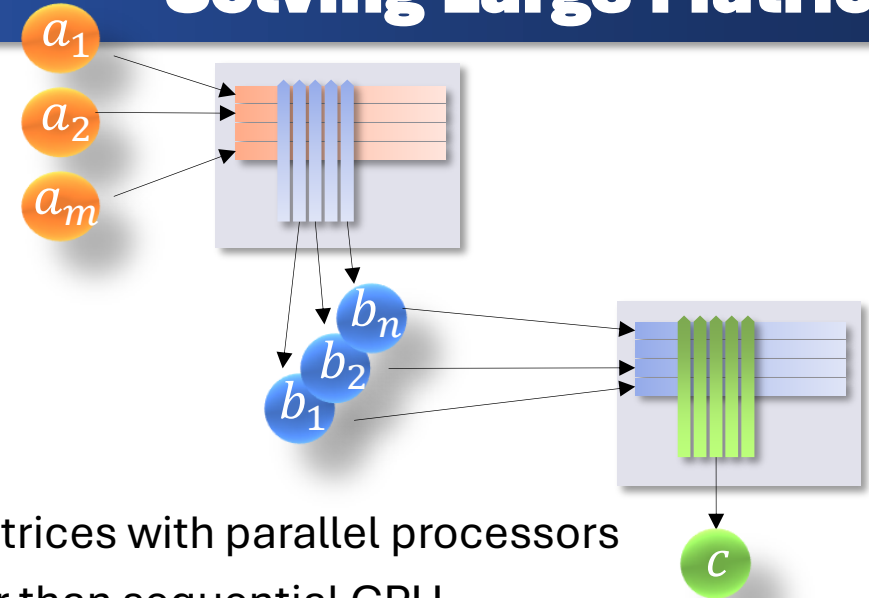
Transfer
Functions

- An LLM does not use English words but tokens
 - ➔ Tokens modify a linguistic structure just like endings like “-ing” do
 - ➔ It is the basis for the Translator
- Tokens are more easily to count and to cluster into nearby embeddings
 - ➔ Embeddings are similar objects
 - E.g., dogs and cats are common pets, whereas crocodiles are not
 - All embeddings are enumerated
 - Tokens have a distance between them
- As results we must solve very large matrices
 - ➔ Like good old Quality Function Deployment (QFD)
 - ➔ Although much, much larger



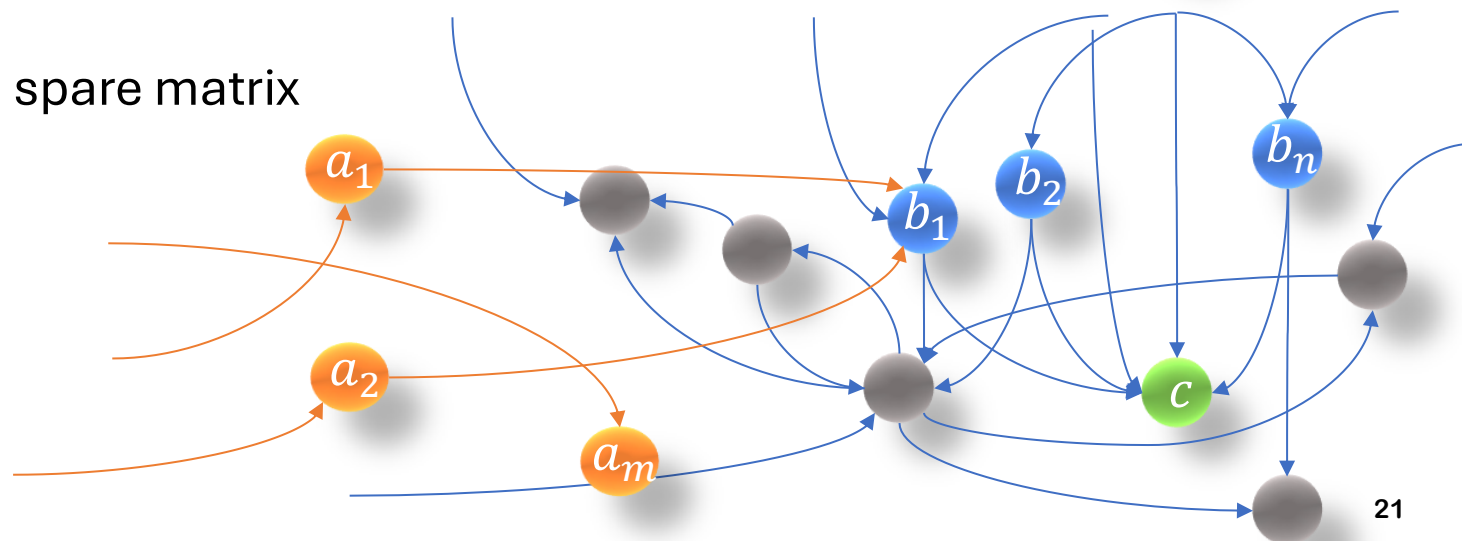
Solving Large Matrices

- There is no hope to calculate the Eigenvector
 - ➔ There doesn't exist an algebraic solutions
 - ➔ All Eigenvalues are numerical approximations
- The big breakthrough was in 2012
 - ➔ It was understood how to calculate very large matrices with parallel processors
 - ➔ Graphical Processor Units (GPU) are much better than sequential CPU



- A typical large AI matrix is a sparse matrix
 - ➔ Most cells remain empty
 - ➔ This is why they are called

Neural Nets



$$M \bullet N = \{b | \exists a_k \rightarrow b \in M, a_k \subset N\}$$

The Graph Model of Combinatory Logic

Selection of Observations

$$x_j \rightarrow y$$

Observed Effect

Grounding Observations



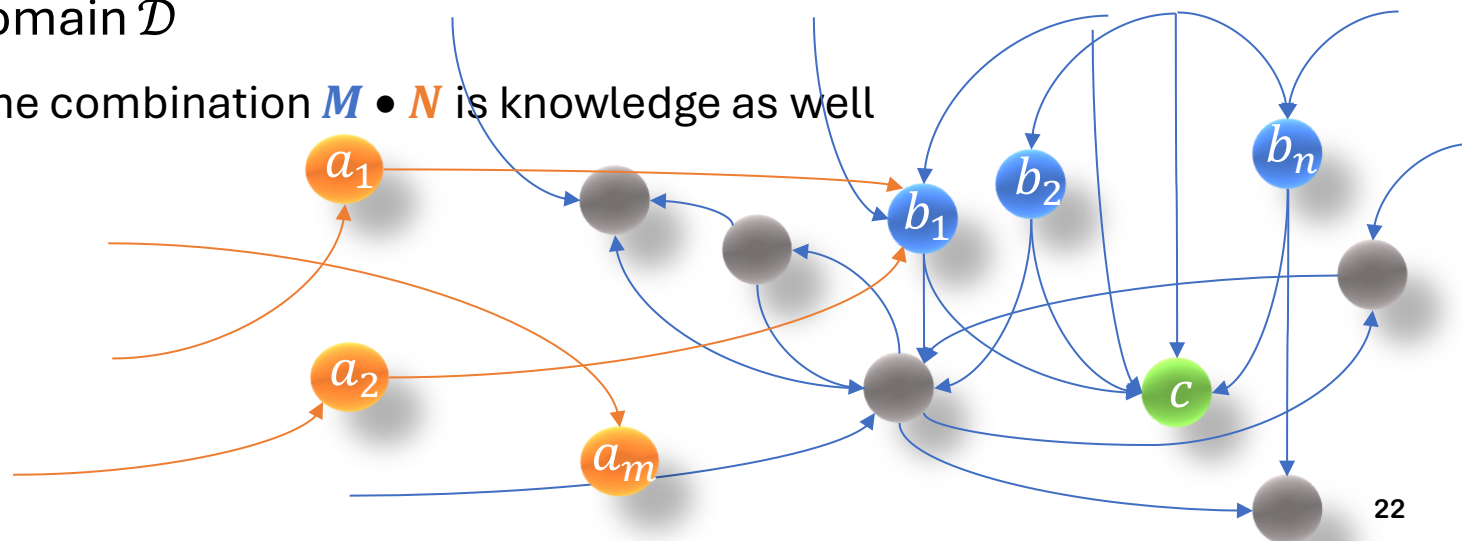
- Algebraic combination of **Observations** and **Concepts** is called **Knowledge** about some Domain \mathcal{D}

➔ If M and N is knowledge, the combination $M \bullet N$ is knowledge as well

➔ $M = ((a_m \rightarrow b)_n \rightarrow c)_j$

➔ $N = (a_m \rightarrow b)_n$

- $M \bullet N = c_j$



$$M \bullet N = \{b | \exists a_k \rightarrow b \in M, a_k \subset N\}$$

What is Intelligent AI?

Selection of Observations

$$x_j \rightarrow y$$

Observed Effect

Grounding Observations

- Intelligence means two different skills
 - ➔ Collecting data (CIA, FBI, ...)
 - ➔ Understanding data (**vous**)
 - ➔ **Intelligent Systems** are smart

- Knowledge means
 - ➔ Know the data
 - ➔ Know the rules
 - ➔ Work smart

- An LLM is intelligent in the first sense, but not smart
 - ➔ It finds answers by statistical fit from the data it collected
 - ➔ It has no idea why this fits

- The **Graph Model of Combinatory Logic**
 - ➔ Selecting relevant instances from the world that it represents (grounding)
 - ➔ It is **Turing complete**:
 - That means you can write rules and full algorithmic programs in the same model

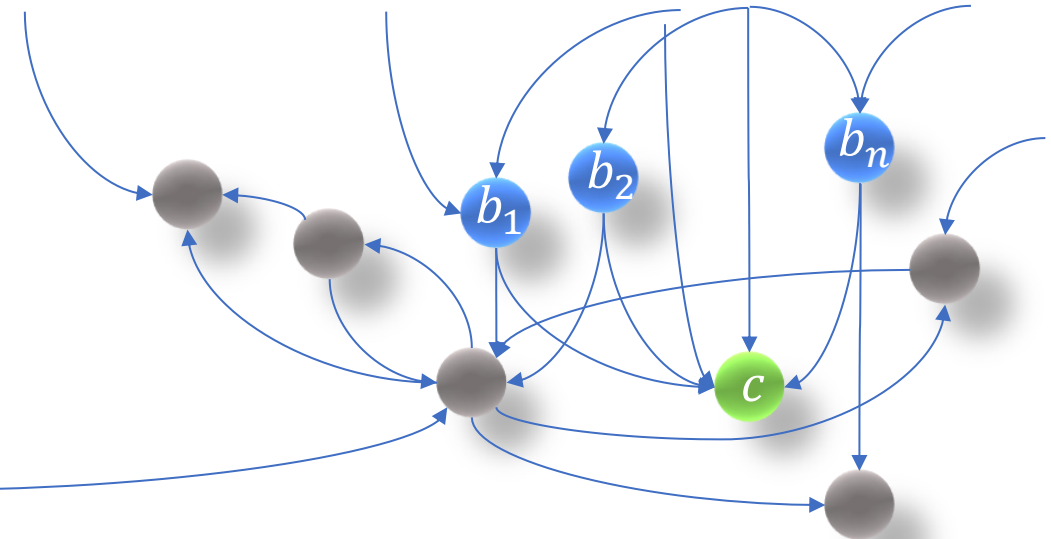
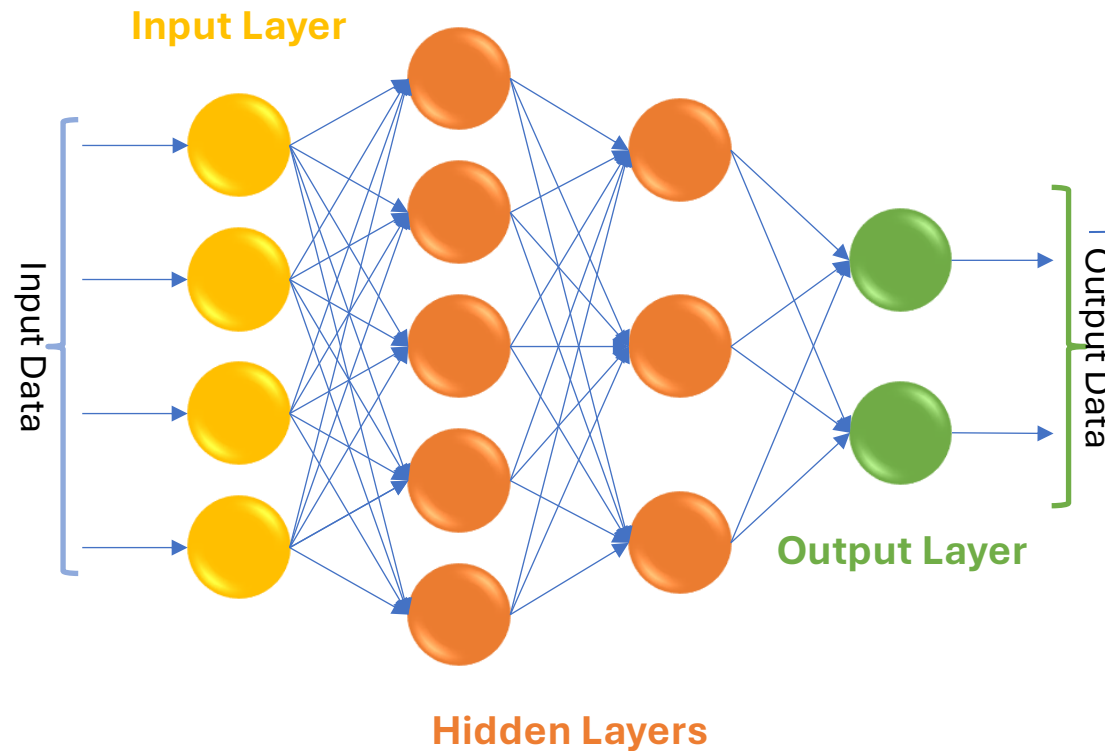


Erwin Engeler
Professor emeritus of Logic
and Computer Science,
Mathematics Department,
ETH, Zurich

Three Types of Knowledge Graphs

1

- Recognizing Objects
- ➔ For instance, visual recognition



- $c \in \mathcal{D}$
- ➔ Graph evaluates to domain element
- ➔ Visual object recognition
- ➔ LLM language recognition

Customer Orientation

Lean Six Sigma

Agile Processes

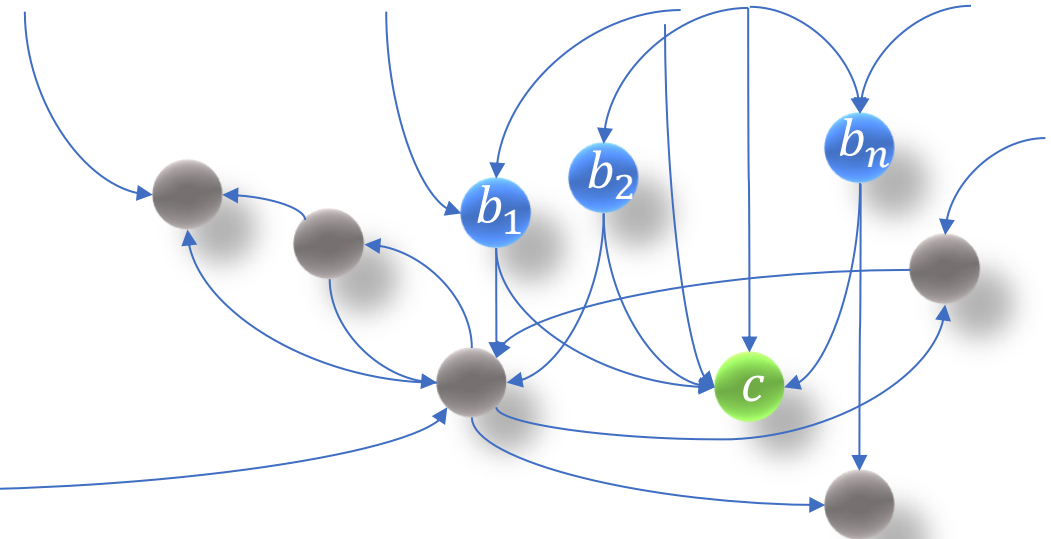
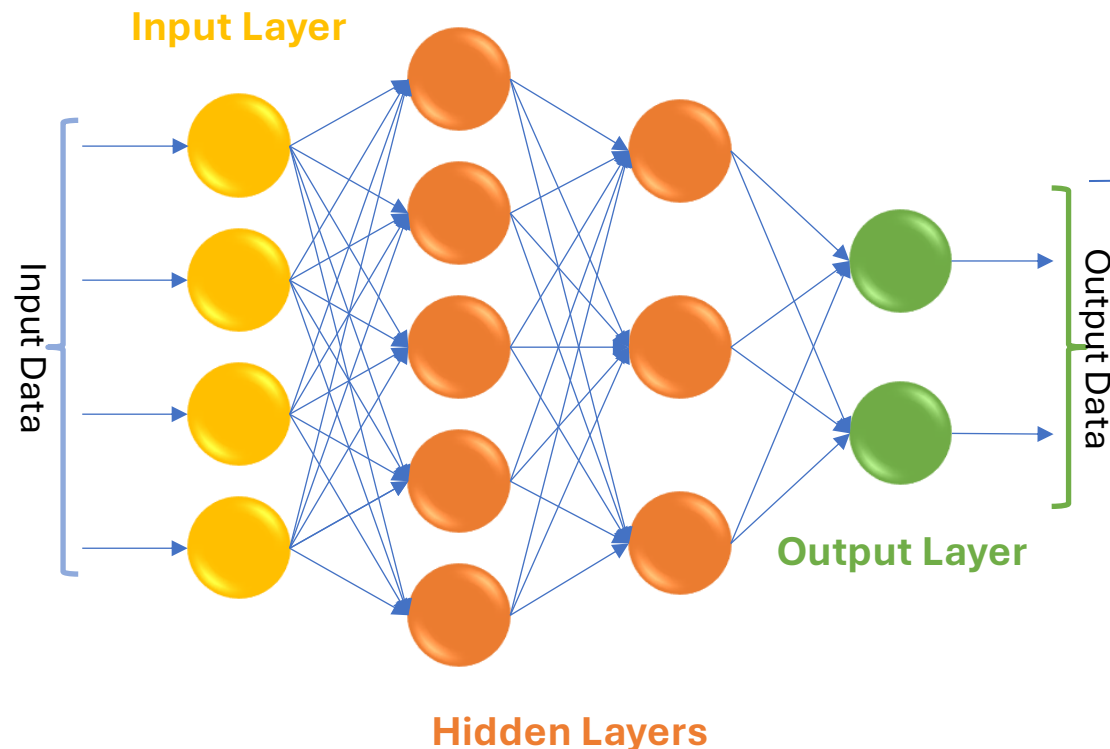
Project Estimations

Transfer Functions

Three Types of Knowledge Graphs

2

- Recognizing Rules
- ➔ Looking at behavior
- ➔ Reinforcement Learning



- $c \in (a \rightarrow b_j)_k$
- ➔ Rules recognition
- ➔ Behavior learning

Customer Orientation

Lean Six Sigma

Agile Processes

Project Estimations

Transfer Functions

Three Types of Knowledge Graphs

3

- Closed Loop Graphs

- ➔ Called Lambda Concepts
- ➔ Algorithmic behavior
- ➔ Traditional Programs
- ➔ **Logical Derivation**



- Substitution of arrow terms



- ➔ $I = a_1 \rightarrow a$
- ➔ $K = a_1 \rightarrow \emptyset \rightarrow a$
- ➔ $S = \left(a_i \rightarrow (b_j \rightarrow c) \right)_1 \rightarrow (d_k \rightarrow b)_i$
 $\rightarrow (a_i + b_{j,i} \rightarrow c)$
- ➔ $\lambda f. Kf = S\lambda f. K\lambda f. f = S(KK)I$

Three Types of Knowledge

- Three Types of Knowledge

1

➔ What objects should it recognize?

- The underlying domain objects
- Perceptrons that recognize them
- Many solutions for the same object

2

➔ What concepts should it learn?

- Some concepts are easier to learn than being programmed
- Continuously adapting to user's behavior and preferences

3

➔ Which concepts are compulsory?

- Deterministic behavior
- Structural knowledge
- Algorithmic programs

- Technical Solutions

➔ **Training Models**

- Typical samples from a typical world
- Deep Learning, continuous updates
- Neural networks

➔ **Reinforcement Learning Concepts**

- Learning to perform actions that are typical for the domain they are built for
- Continuously adapting by collecting and evaluating experiences

➔ **Lambda Concepts**

- Reacting on specific objects & scenarios
- Predefining behavior
- Compulsory decisions

Three Types of Tests

Customer
Orientation

Lean
Six Sigma

Agile
Processes

Project
Estimations

Transfer
Functions

● Three Types of Knowledge

1

➔ What objects should it recognize?

- The underlying domain objects
- Perceptrons that recognize them
- Many solutions for the same object

2

➔ What concepts should it learn?

- Some concepts are easier to learn than being programmed
- Continuously adapting to user's behavior and preferences

3

➔ Which concepts are compulsory?

- Deterministic behavior
- Structural knowledge
- Algorithmic programs

● A Test is passed, when

➔ The AI Agent recognizes the object with expected reliability

- The individual test case is not predictable
- The knowledge is of limited reliability

➔ The AI Agent recognizes the concept

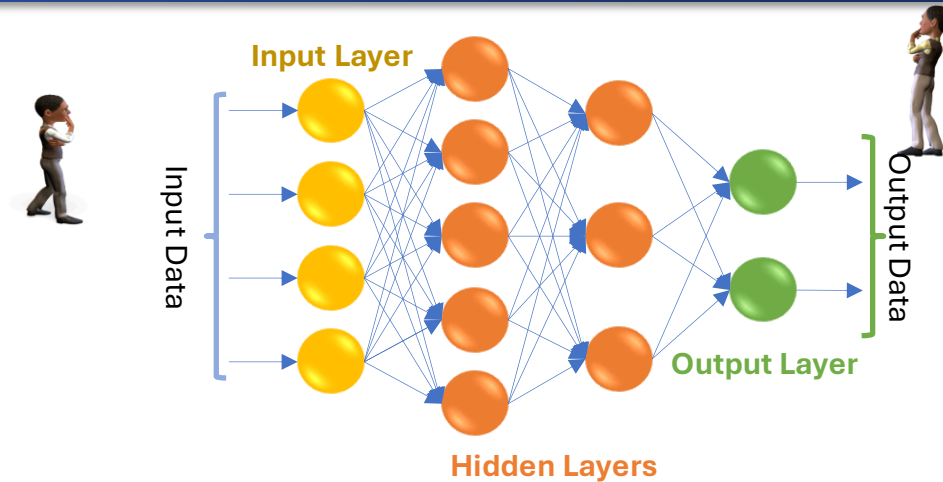
- The individual test case is not predictable
- The knowledge is of limited reliability
- Design of AI Agent must include corrective measure in case the response is wrong

➔ Responses are correct

- The AI Agent uses knowledge that is 100% reliable – usually written in Python
- Testable like any traditional program

Explainable AI (XAI)

- A big problem for AI is that nobody knows what happens in hidden layers
- It is unlikely that hidden layers implement the most straightforward arrow schemes
 - ➔ In contrary, combinatory logic has no normal form
 - ➔ There exist many ways to find a valuable result, even for the same result
- Regulators try to impose rules for training sets
 - ➔ It is very difficult to assess quality of a training set, because knowledge is always limited
- Current explainability techniques
 - ➔ SHAP (SHapley Additive exPlanations) is a game-theoretic approach
 - ➔ LIME (Local interpretable model-agnostic explanations) is a method that fits a surrogate glass-box model around the decision space of any black-box model's prediction
 - ➔ A lot more, all based on statistical correlations, not on reasoning

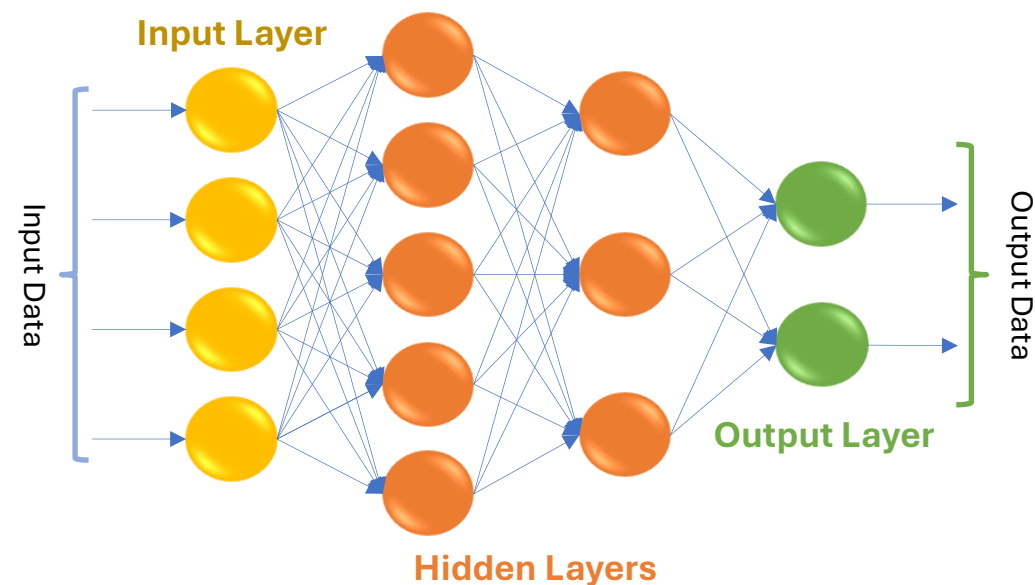


Intelligent Systems are Based on Controlling Combinators

- **Controlling Combinators** \mathcal{C} do loop and augment knowledge X under control
 - ➔ Controlling Combinators are of type ③
 - ➔ Focusing $X_{i+1} = \mathcal{C} \cdot X_i, i \in \mathbb{N}$
 - ➔ Control Sequence $X_0 \subseteq X_1 \subseteq X_2 \subseteq \dots$
 - ➔ The amount of available knowledge is measured as **Reliability**
 - ➔ Stop looping when approximation to correct response is good enough

- Neural Networks in the brain do loop!
 - ➔ Controlling Combinators keep looping programs under control

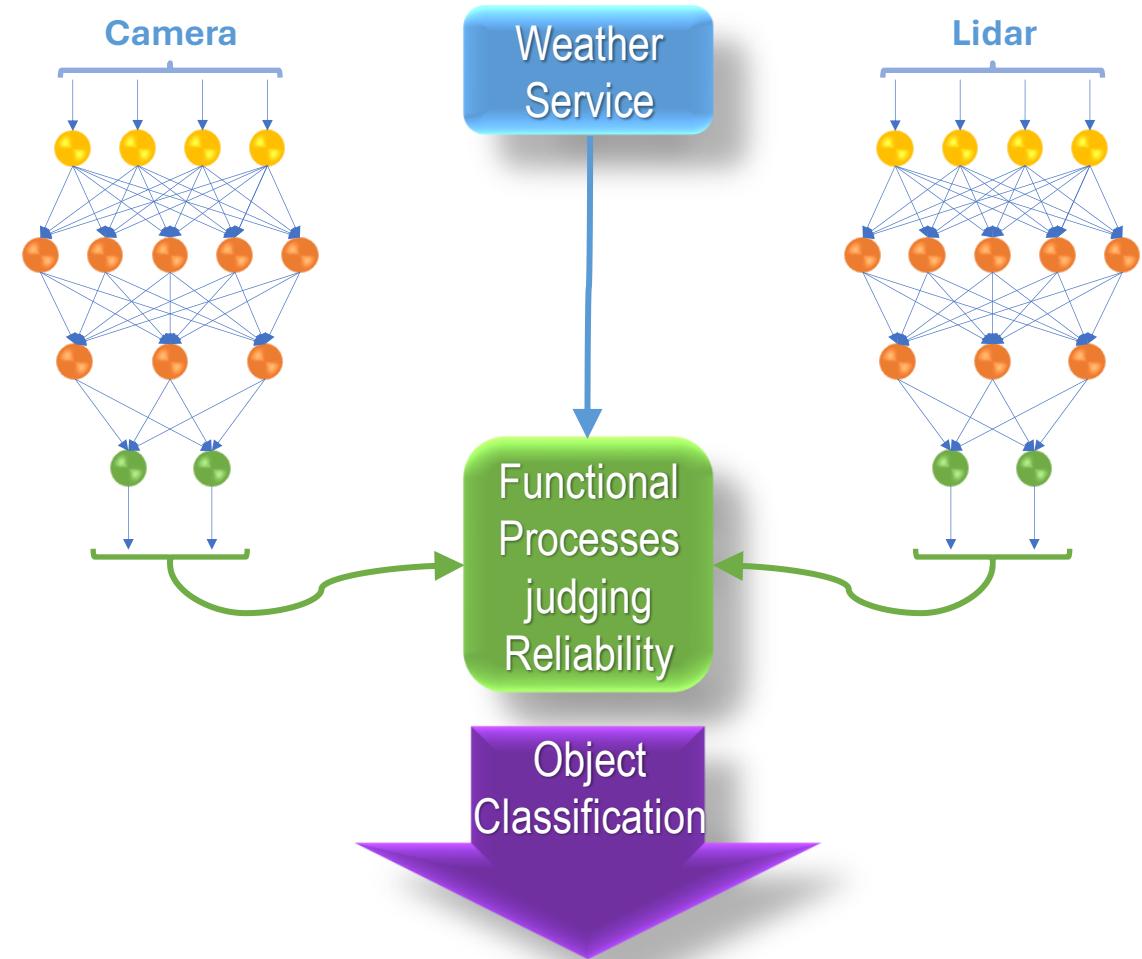
- Deep Learning by multi-layered perceptrons (MLP) is ordered
 - ➔ No loops from input layer to output layer



Intelligent Systems Consisting of Many AI Engines

- Assume reliability can be measured
 - ➔ You can compare reliability of two AI engines, e.g., camera and Lidar
 - ➔ The judge about which reliability is higher
 - That might depend on external factors such as weather condition or lighting
 - ➔ Then decide according to the best prediction

- Intelligent systems arise from cooperation between AI and functional programming



Customer Orientation

Lean Six Sigma

Agile Processes

Project Estimations

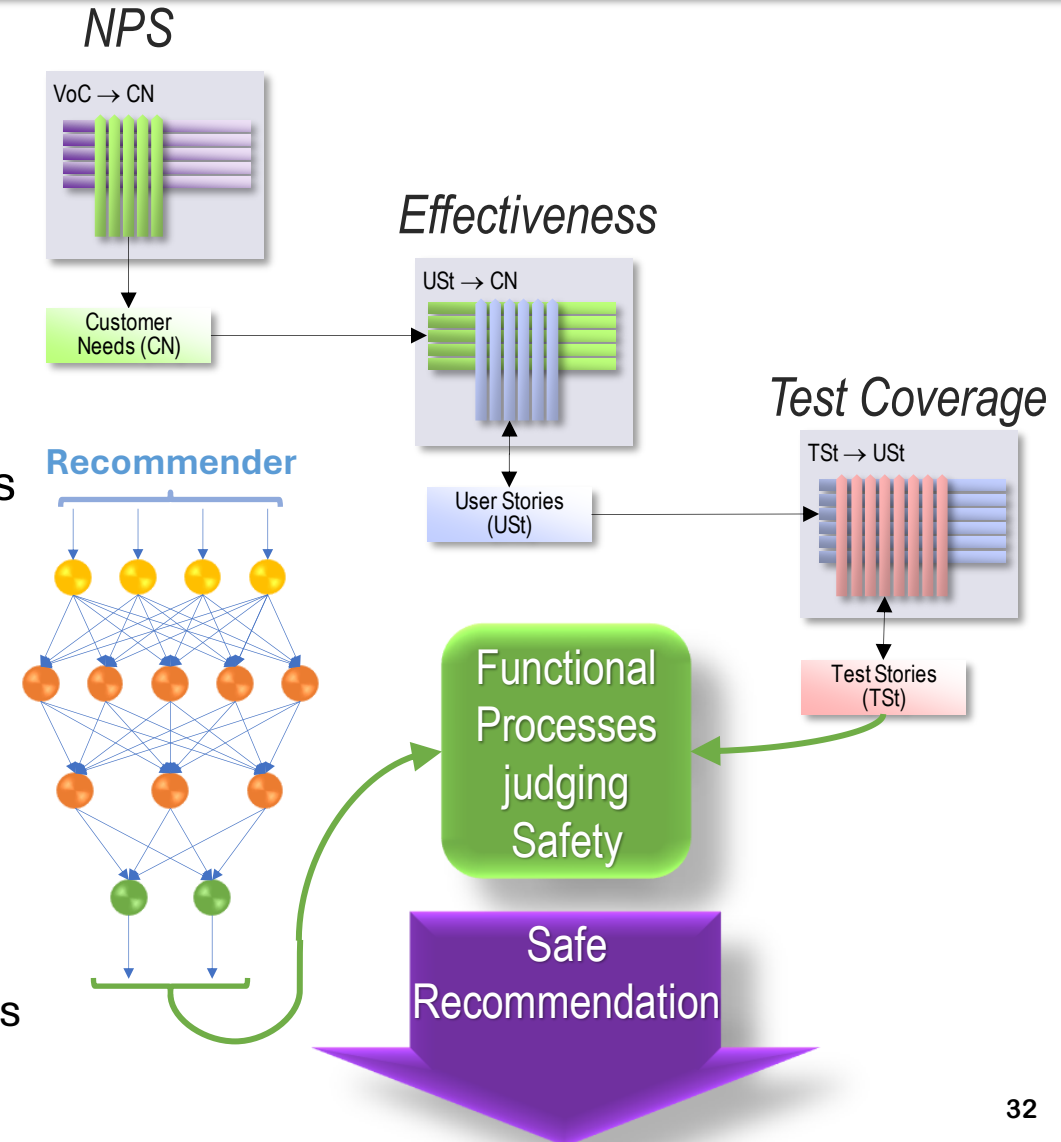
Transfer Functions

Combine QFD Test Deployment with AI Recommender

- QFD Test Deployment
 - ➔ Net Promoter Score yields Customer Needs
 - ➔ Effectiveness yields User Stories (FUR)
 - ➔ Test Coverage yields Test Stories

- Test Stories consist of set of arrow schemes (*Test Data → Response*)

- AI Recommender produces a set of arrow schemes that can be compared with Test Stories
 - ➔ This yields real-time validation for AI recommendations
 - ➔ Their arrow schemes must fit into Test Stories



Customer Orientation

Lean Six Sigma

Agile Processes

Project Estimations

Transfer Functions

Controlling Combinators Develop Skills in Intelligent Systems

Customer Orientation

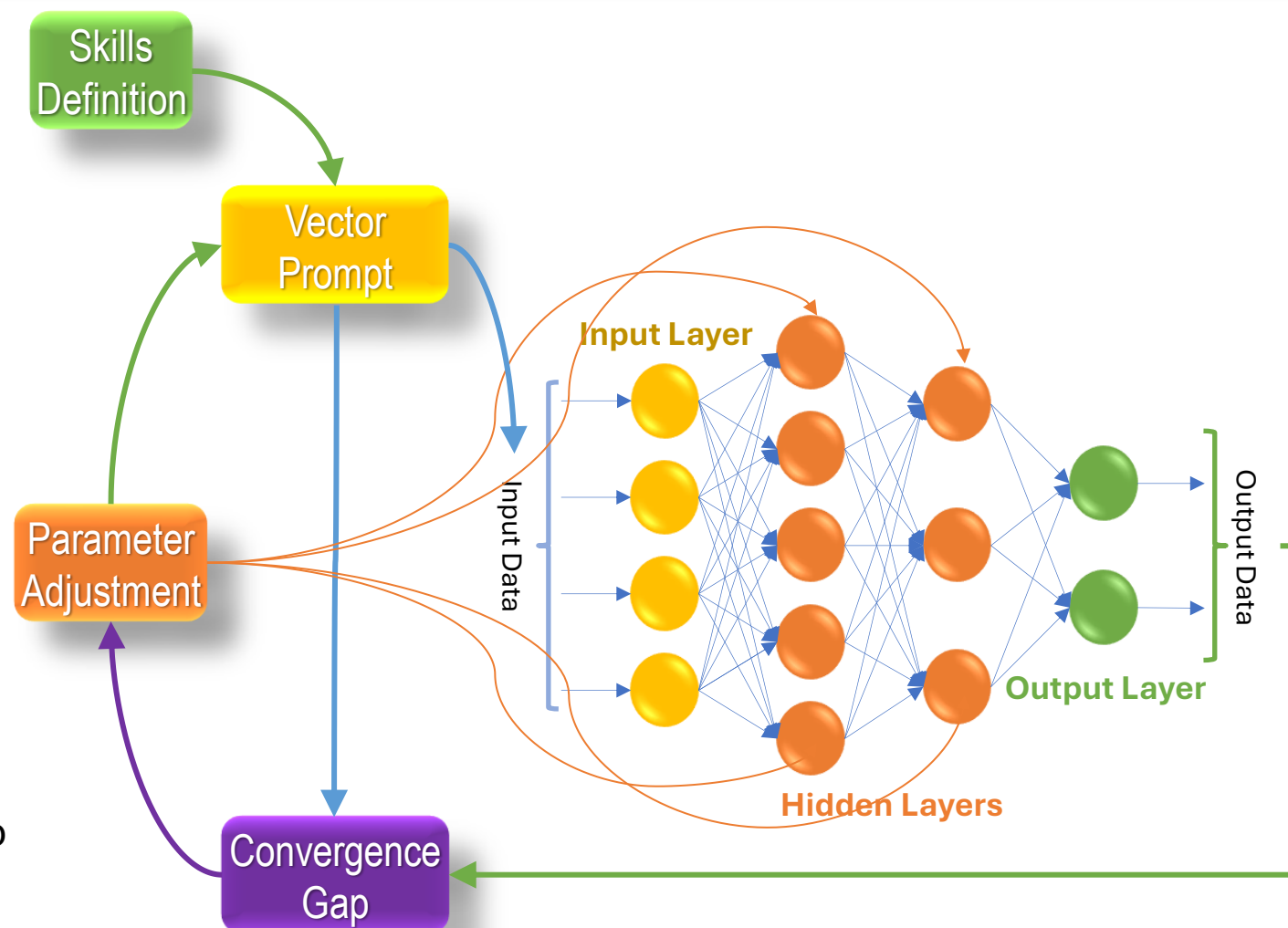
Lean Six Sigma

Agile Processes

Project Estimations

Transfer Functions

- The **Skills Definition**
 - ➔ How do we identify the skills that we want to improve
- The **Vector Prompt**
 - ➔ The vector furnished to the neural net to stimulate a response
- The **Convergence Gap**
 - ➔ The difference between skill goals and skills achieved
- The **Parameter Adjustment**
 - ➔ The deep learning strategy to improve the AI response



The Cause-Effect Concept Generator (CECG)

Customer
Orientation

Lean
Six Sigma

Agile
Processes

Project
Estimations

Transfer
Functions

- An **Intelligent System** is a hybrid system between Symbolic AI and LLMs that checks responses with a **Cause-Effect Concept Generator (CECG)**
 - ➔ A CECG generates rules derived from some rule set, using the same tokens as the LLM
 - ➔ Because tokens have a distance, building a convergence gap calculator is relatively easy
 - ➔ The CECG uses mathematical logic to derive valid combination of rules
- 15 years ago, the world was full of IoT Smart Systems, also called Intelligent Systems
 - ➔ However, they used no AI Engines
 - ➔ All was algorithmic
 - ➔ Type ③ requirements
- An LLM is not smart
 - ➔ A CECG can be smart
 - ➔ It's classical, symbolic AI
 - ➔ With smart search



Customer
Orientation



QFD as Algebra of Combinators

Lean
Six Sigma



AI as Algebra of Combinators



Using QFD for Agentic AI

Transfer
Functions

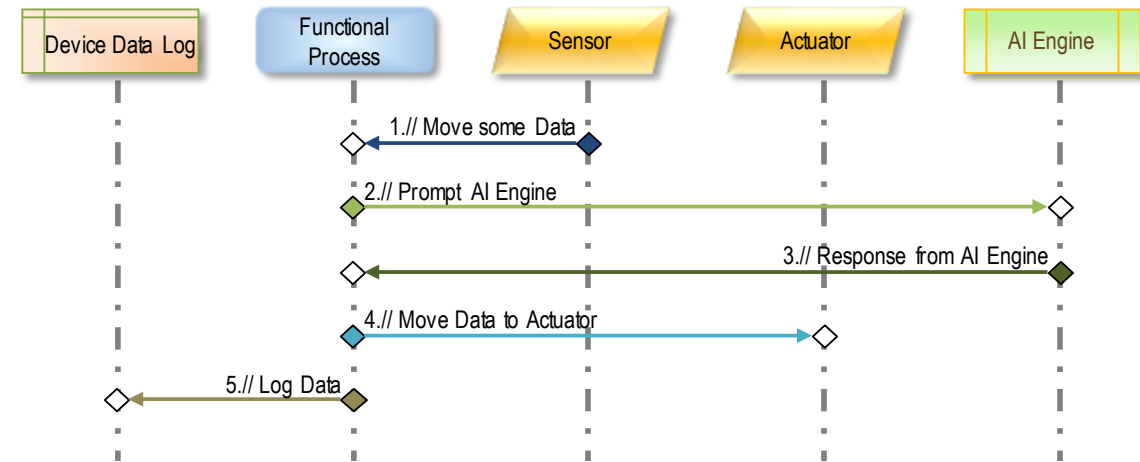


Conclusions

Modeling Functionalities According to ISO 19761 (COSMIC)

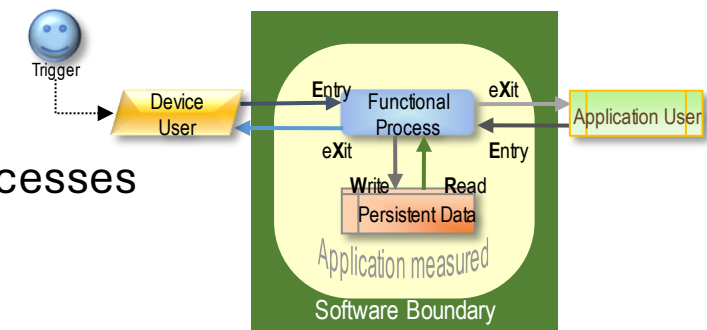
- **Data Movements** move **Data Groups**

- ➔ **Data Groups** categorize what kinds of data is moved between objects
- ➔ **Data Groups** are relevant for security aspects, such as **Privacy** and **Safety**
- ➔ **Data Groups** transport **Knowledge**



- Data Movements always involve a **Functional Process**

- ➔ Either **Entry & eXit**, or **Read & Write**
- ➔ Functional Processes communicate with other Functional Processes either via Devices, or Persistent Data



- Data Groups containing knowledge have a certain **Reliability**

Customer Orientation

Lean Six Sigma

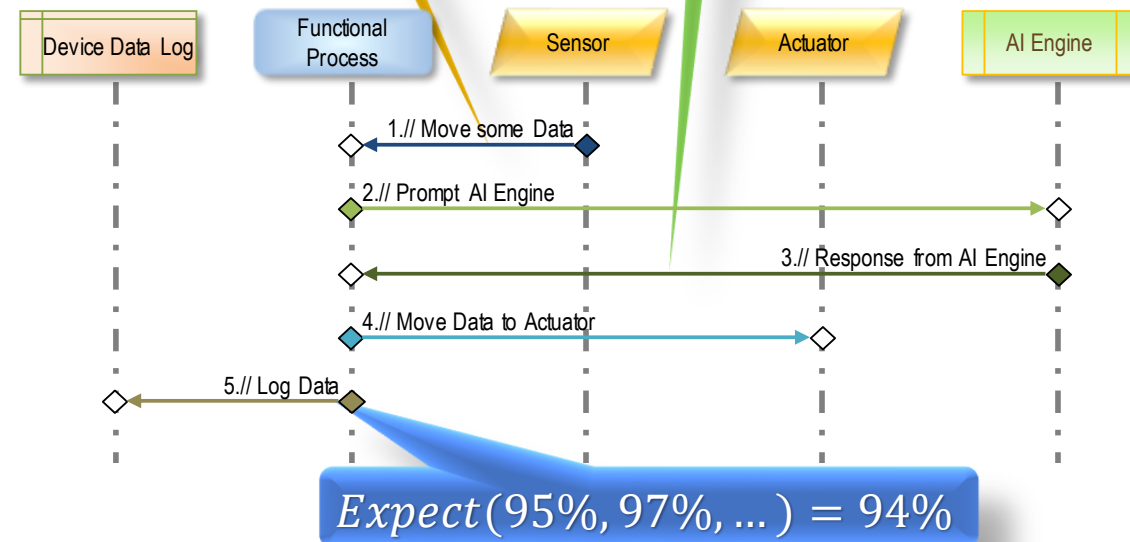
Agile Processes

Project Estimations

Transfer Functions

In AI, Data Groups become Knowledge with Varying Reliability

- **Reliability** of Data Groups depends on the Objects of Interest
 - ➔ Devices
 - Data comes with a reliability ratio
 - ➔ Other Applications
 - An AI app comes with a reliability ratio
 - ➔ Persistent Data
 - Might contain data with a reliability ratio



- **Reliability** of Data Groups combines on Functional Processes
 - ➔ Combination calculates statistical expectation of reliability originating from all input data groups
 - ➔ Functional Processes yield data groups with combined reliability
- Programmers of AI should take care of reliability of the data groups moved

A Simple AI Agent for Implementing Business Processes

- Our Organization needs an AI Agent that quotes customers individually based on

- ➔ Business Rules
 - For Terms & Condition
 - Discounts
- ➔ Price Book



- Customer Needs
 - ➔ By AHP
 - ➔ Integrating stakeholders
 - End Customers
 - Management
 - Compliance

AHP Priorities		y1	y2	y3	y4	y5	Weight	Ranking	Profile
Customer Needs		Responsive	Explanation	Reliable	Creative	Trusted			
y1	Responsive	1	3	1/3	1/6	6	21%	2	0.45
y2	Explanation	1/3	1	1/3	1/3	3	11%	5	0.23
y3	Reliable	3	3	1	6	1	29%	1	0.63
y4	Creative	6	3	1/6	1	1/6	20%	3	0.43
y5	Trusted	1/6	1/3	1	6	1	19%	4	0.40

Customer Orientation

Lean Six Sigma

Agile Processes

Project Estimations

Transfer Functions

Functional User Requirements as User Stories

Customer Orientation



Lean Six Sigma

Agile Processes

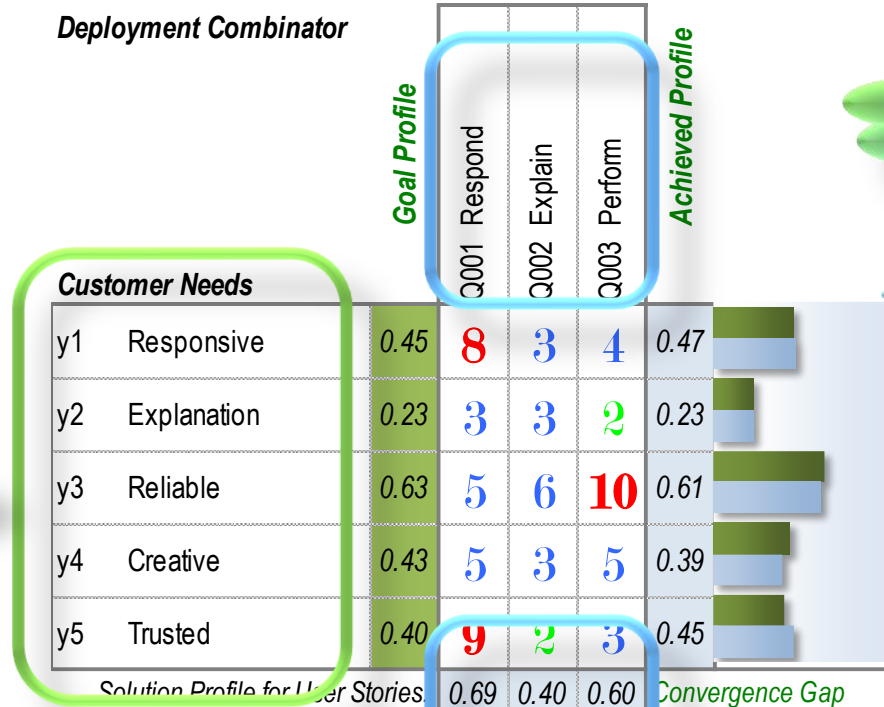
Project Estimations

Transfer Functions

What are Customer Needs for some User of an AI Agent?

Customer Needs
Deployment Combinator

User Stories



Which User Stories fulfill Customer Needs?

After analyzing Customer Needs, we can define a Profile for User Stories needed to implement an AI Agent

71 Total Effort Points

0.10 Convergence Range

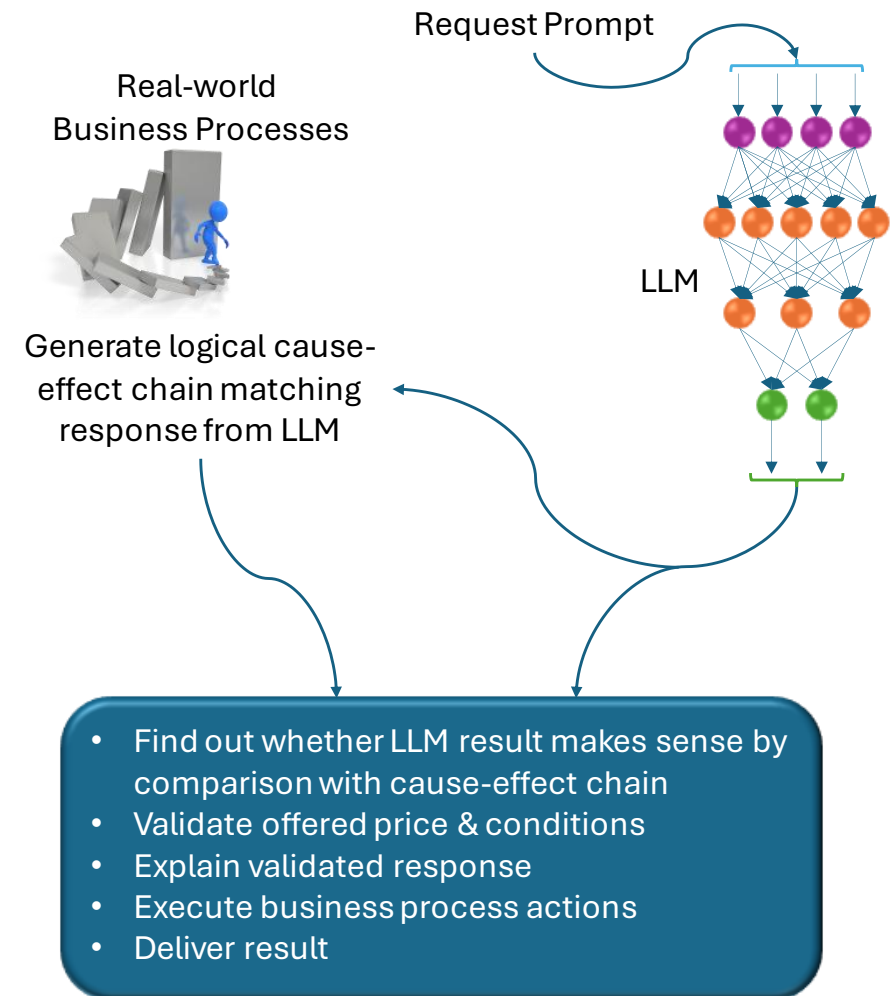
0.20 Convergence Limit

User Stories

User Stories Topics	As a ... [functional user]	I want to ... [get something done]	such that ... [quality characteristic]	so that ... [value or benefit]	Priority
1) Q001 Respond	Business Manager	AI responding to customer queries	the customer understands	customer needs are fulfilled	Weight: 41%, Profile: 0.69
2) Q002 Explain	Process Owner	explain decisions taken by the AI Agent	the customer understands	customer feels understood	Weight: 24%, Profile: 0.40
3) Q003 Perform	Process Owner	the Agent executing the process	the process is automated	I can save cost	Weight: 35%, Profile: 0.60

Design of an AI Agent

- An **Intelligent System** is a hybrid system between Symbolic AI and LLMs that checks responses with a **Cause-Effect Concept Generator (CECG)**
 - ➔ A CECG generates rules derived from some rule set, using the same tokens as the LLM
- An **AI Agent** is an Intelligent System whose CECG models Business Processes for some organization
- Type ③ requirements for an AI Agent:
 - ➔ All responses must be consistent with business processes
 - ➔ The AI Agent never must hallucinate!



Customer Orientation

Lean Six Sigma

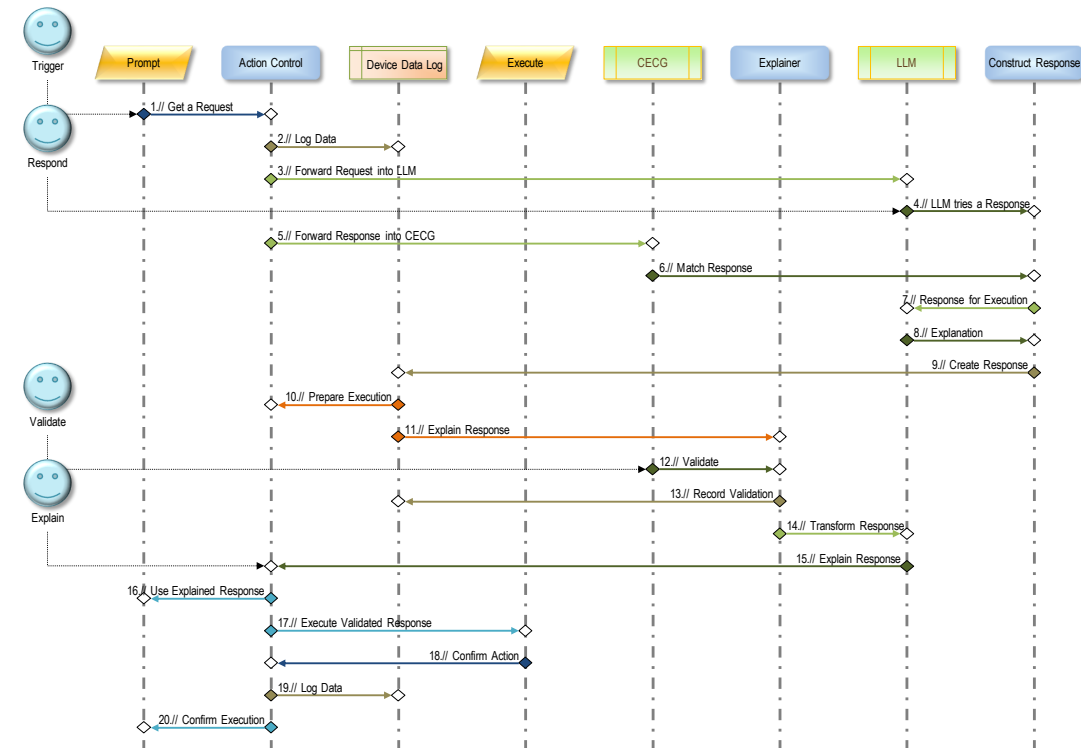
Agile Processes

Project Estimations

Transfer Functions

Design of an AI Agent

- An **Intelligent System** is a hybrid system between Symbolic AI and LLMs that checks responses with a **Cause-Effect Concept Generator (CECG)**
 - ➔ A CECG generates rules derived from some rule set, using the same tokens as the LLM
- An **AI Agent** is an Intelligent System whose CECG models Business Processes for some organization
- Type **3** requirements for an AI Agent:
 - ➔ All responses must be consistent with business processes
 - ➔ The AI Agent never must hallucinate!



Customer Orientation

Lean Six Sigma

Agile Processes

Project Estimations

Transfer Functions

Design of an AI Agent

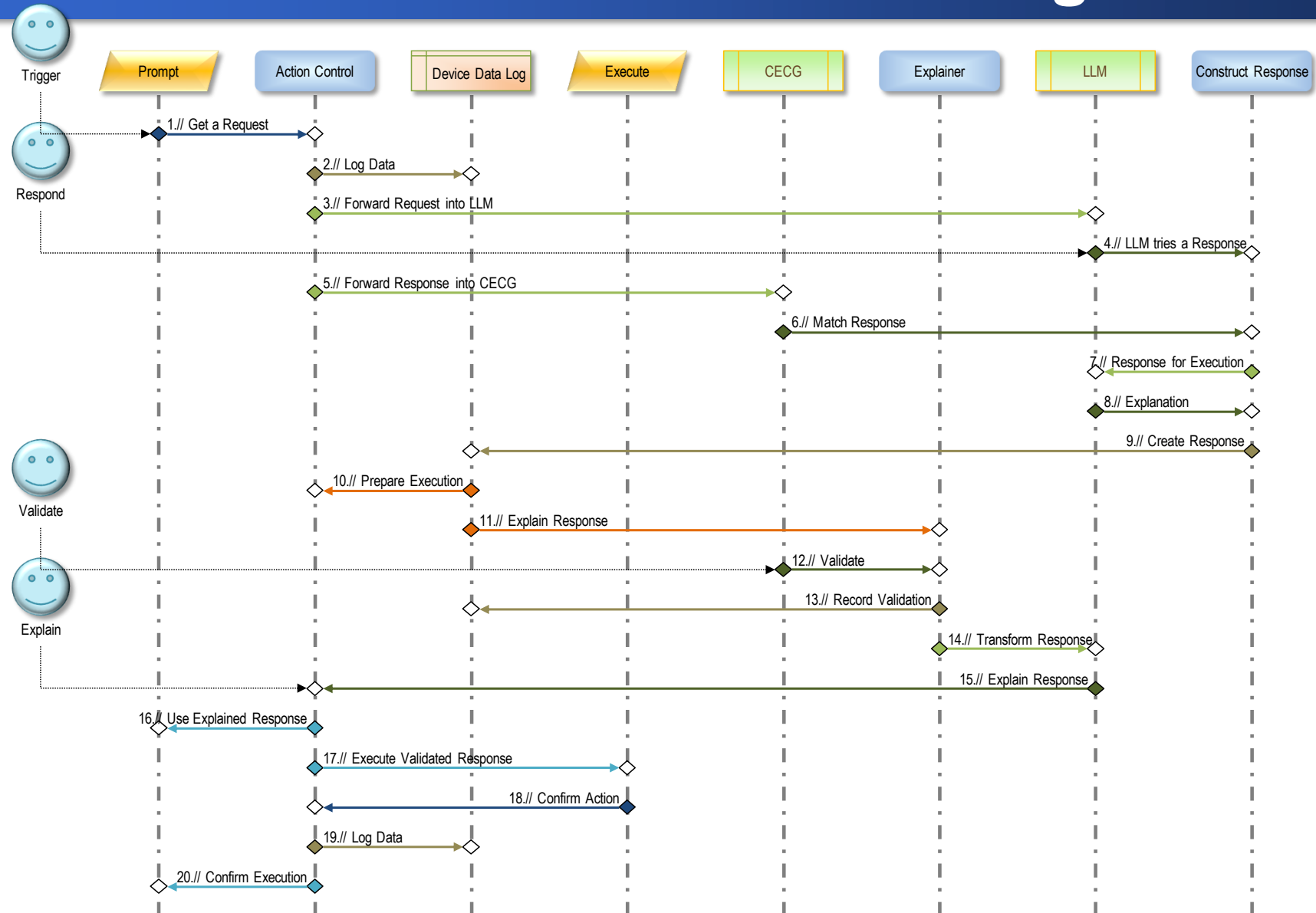
Customer Orientation

Lean Six Sigma

Agile Processes

Project Estimations

Transfer Functions



Splitting a Data Movement Map into Independent Test Cases

Customer Orientation

Lean Six Sigma

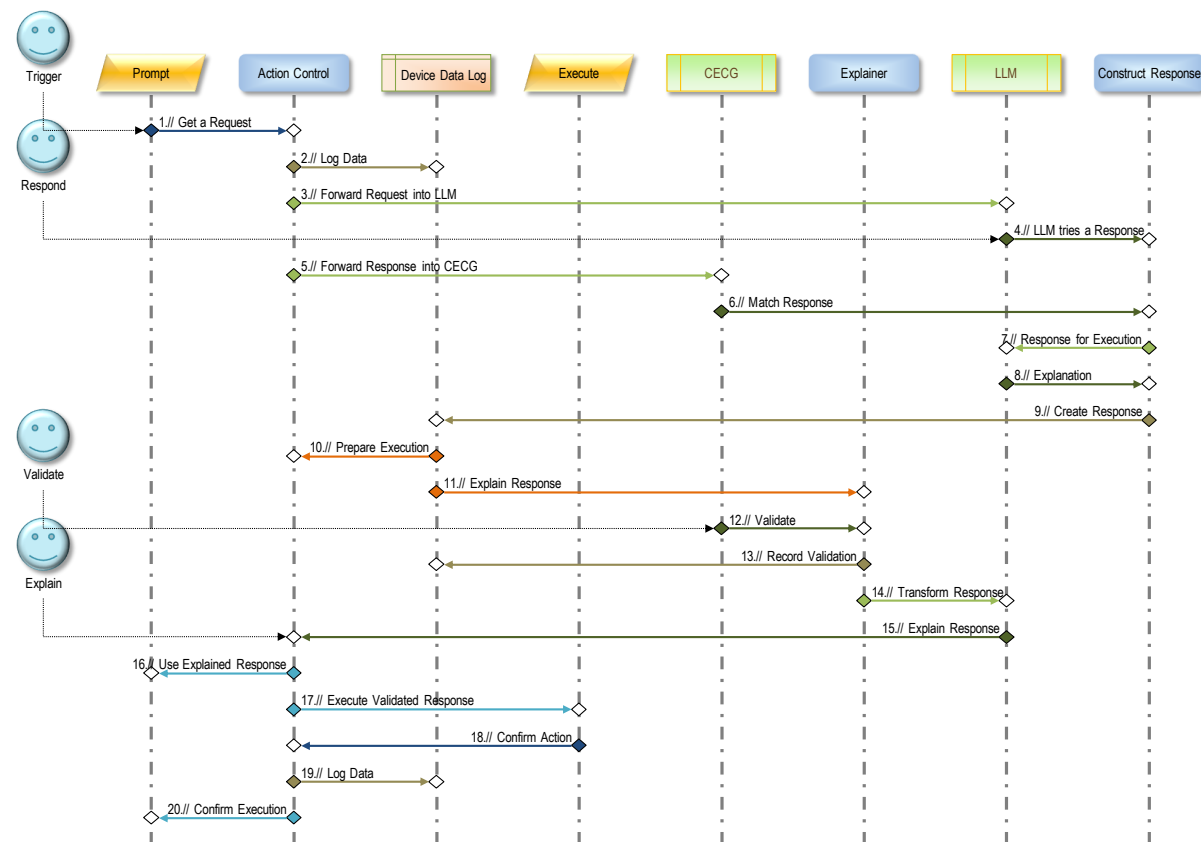
Agile Processes

Project Estimations

Transfer Functions

- Data groups play a major role
 - ➔ Collect the referenced data
 - ➔ Data groups hold test data
 - You might freely choose test data within data group boundaries
 - Thus, tests become executable as soon as the data groups are known for each test case

- For a test case, it is sufficient to consider the data group as test data for each triggered entry data movement
 - ➔ These are the initial graph entries!



Splitting a Data Movement Map into Independent Test Cases

Customer Orientation

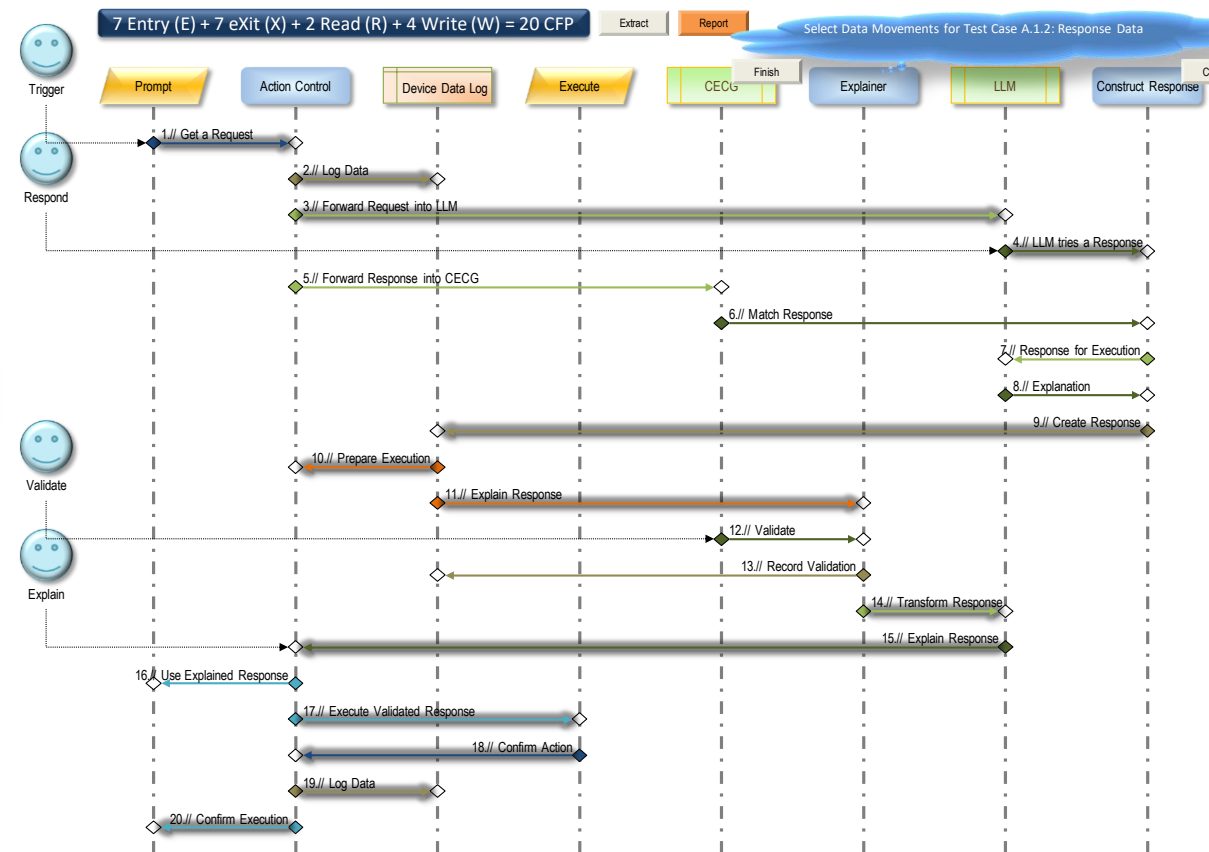
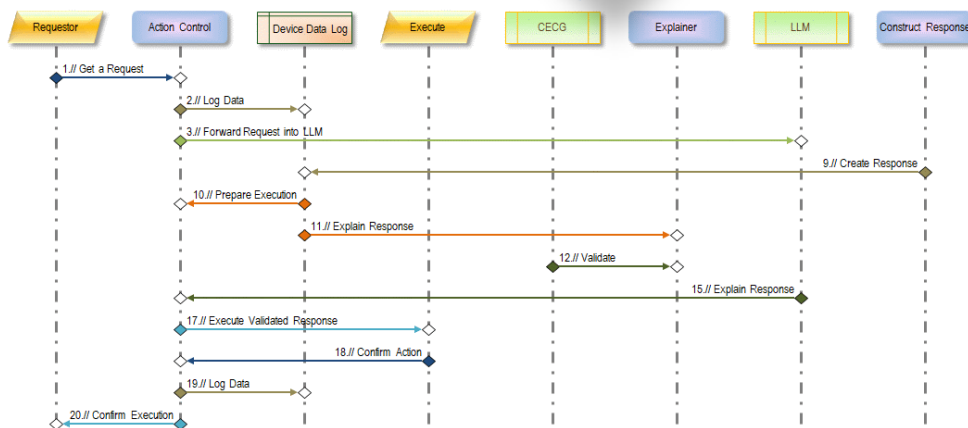
Lean Six Sigma

Agile Processes

Project Estimations

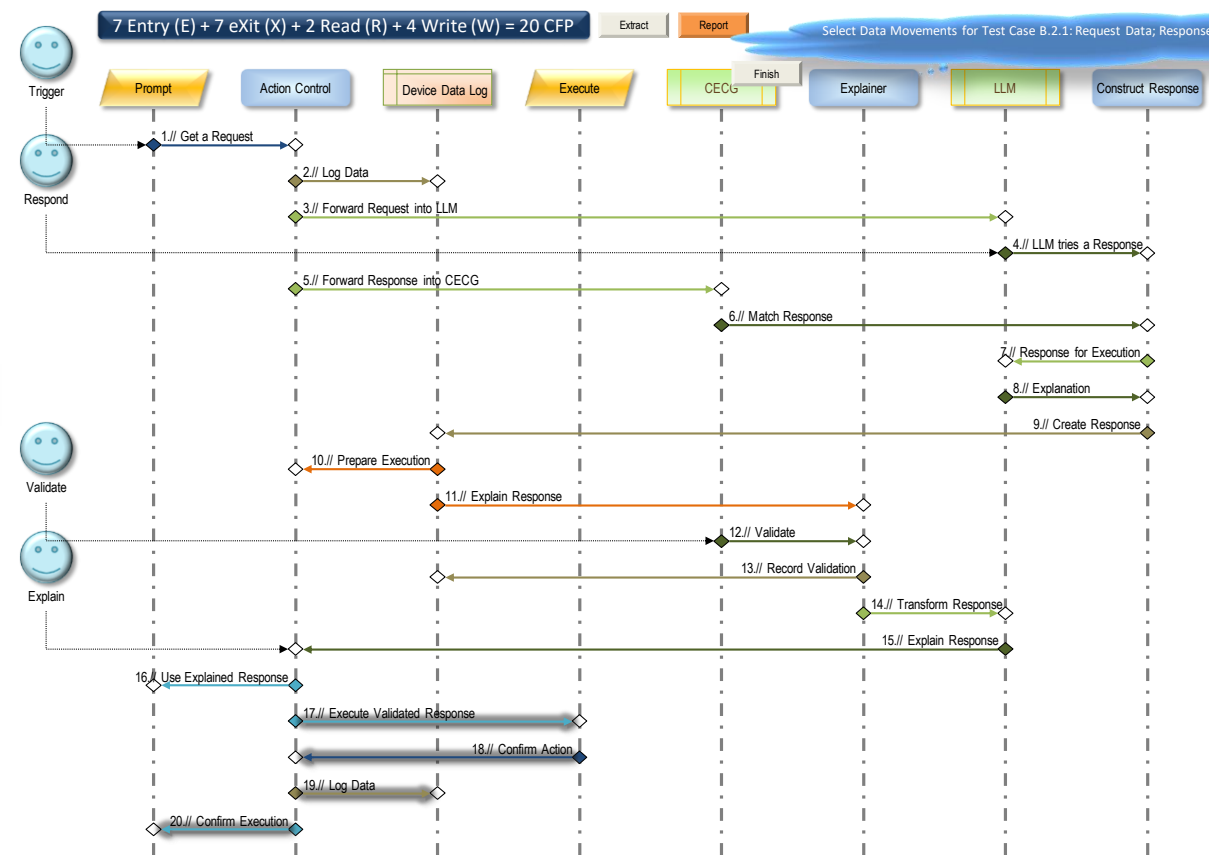
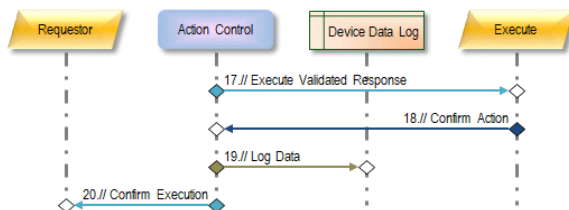
Transfer Functions

- Now click on the data movements that belong to the test case
 - ➔ Test Case must be coherent
 - All data movements must be reachable
 - Data groups might carry a **Reliability**



Splitting a Data Movement Map into Independent Test Cases

- Smaller test cases are also possible
 - ➔ Unit tests often consist of one data movement only



Customer Orientation

Lean Six Sigma

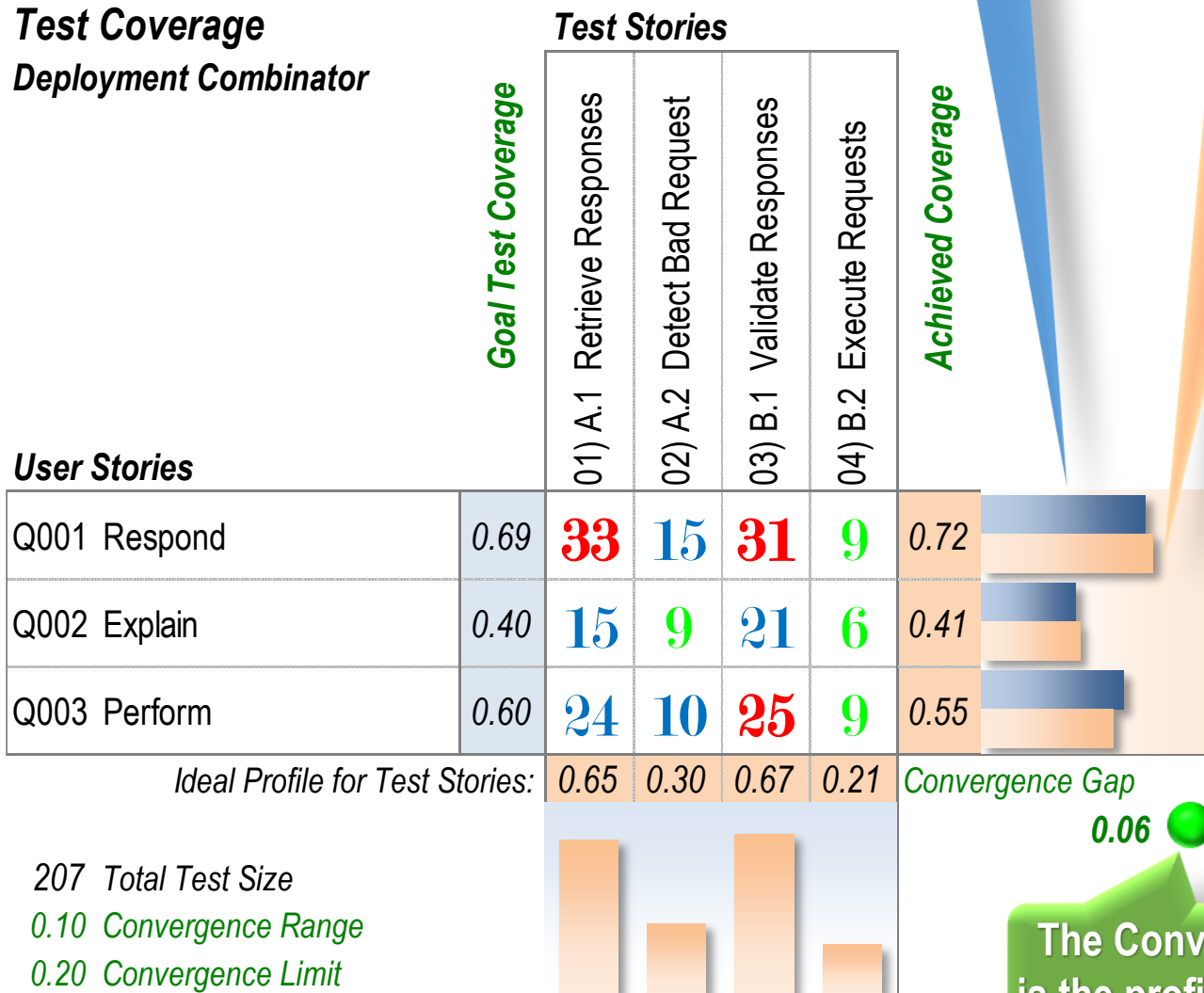
Agile Processes

Project Estimations

Transfer Functions

Test Coverage Matrix

Test Coverage Deployment Combinator



With the Test Coverage Matrix, you can see the effect of test cases

- ➔ You can always argue that too much tests do no harm
- ➔ Except to the budget

Thus, balancing the Test Coverage Matrix is a key requirements for automated and autonomous real-time testing

- ➔ Remember, with AI Agents, testing is an ongoing task that never ends
- ➔ The data movement count can vary depending on test cases.

The Convergence Gap is the profiles' difference

Customer Orientation

Lean Six Sigma

Agile Processes

Project Estimations

Transfer Functions

Customer
Orientation

Lean
Six Sigma

Agile
Processes

Project
Estimations

Transfer
Functions

		Test Cases								Priority		Measured Defect Profile
	Test Story	Case 1 Test Data	Expected Response	Case 2 Test Data	Expected Response	Case 3 Test Data	Expected Response	Case 4 Test Data	Expected Response	Weight	Profile	
A Prepare ① ②	A.1 Retrieve Responses	A.1.1 {Request Data}	Response Data	A.1.2 {Response Data}	Execution Data	A.1.3 {Execution Data; Business Process}	Explanation	A.1.4 {Explanation}	Transformed Response	35%	0.65	
	A.2 Detect Bad Request	A.2.1 {Request Data}	No Response	A.2.2 {}	No Execution	A.2.3 {}	Explanation			16%	0.30	
B Response ③	B.1 Validate Responses	B.1.1 {Bad Request Data; Business Processes}	No Execution	B.1.2 {Bad Response Data; Business Processes}	No Execution	B.1.3 {No Execution}	Explanation	B.1.4 {Explanation}	Transformed Response	37%	0.67	
	B.2 Execute Requests	B.2.1 {Request Data; Response Data}	Execution Data	B.2.2 {Request Data; Response Data}	Explanation	B.2.3 {Explanation}	Transformed Response			11%	0.21	

- With 14 Test Cases only, an AI Agent can be tested effectively for compliance to business processes
 - ➔ Or for a price list, or for other real-world constraints
 - ➔ Test Stories A are mainly type ① and ②; responses are type ③
 - ➔ **There is no excuse not to test your AI Agent!**
- But you must do it the right way for AI
 - ➔ This is not an ordinary test
 - ➔ Testing AI is an ongoing process – you're never done with it

Automated Test Case Execution

- Since each data movement moves exactly one data group, once a test case $x_i \rightarrow y$ is known, sample test data from the data groups x_i can be selected
 - ➔ Thus, the test case becomes executable
 - ➔ If the same data group is displayed multiple times, we take them all
 - ➔ The expected result is also displayed as a data movement that moves a specific data group
 - ➔ All what is needed is to specify valid data ranges for these data groups
 - Selecting specific data from a data group, e.g., from a data range, may lead to multiple executions
 - ➔ Any path in the data movement map that leads from the test data to the result can be executed as a test case
 - Thus, a test case does not specify a unique test case execution

- Not only can test cases be created automatically, thus completing the test coverage matrix, but the execution of test cases no longer requires extra coding
 - ➔ If you have APIs, the approach works also for services where you don't have access to code

Customer
Orientation

Lean
Six Sigma

Agile
Processes

Project
Estimations

Transfer
Functions

QFD Deming Chain and Multi-Layered Perceptron

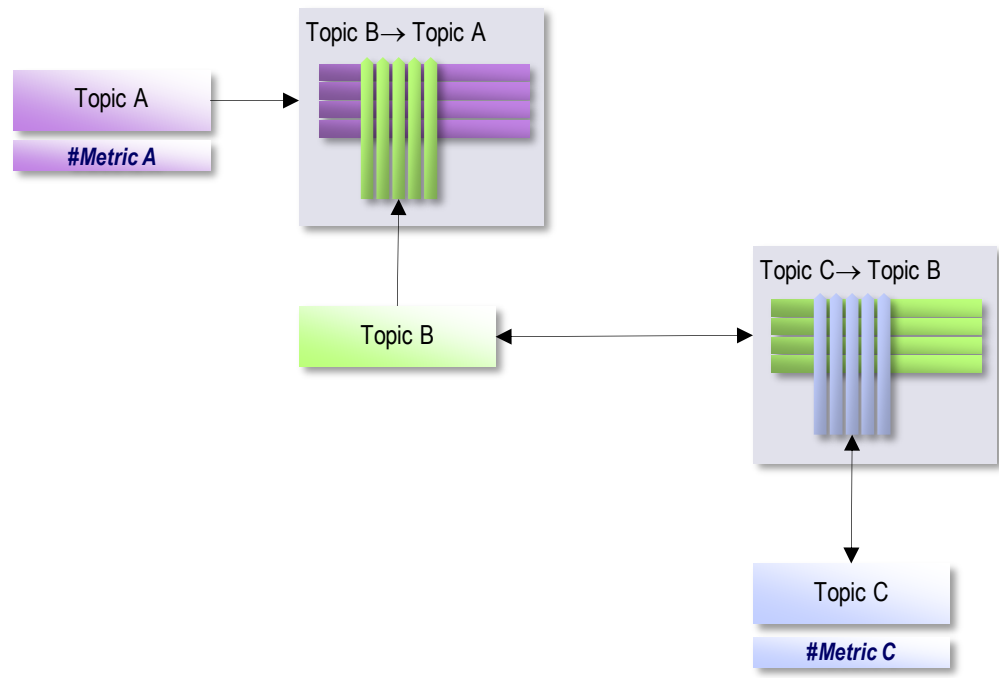
Customer Orientation

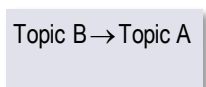
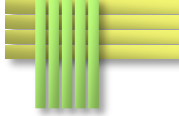



Lean Six Sigma

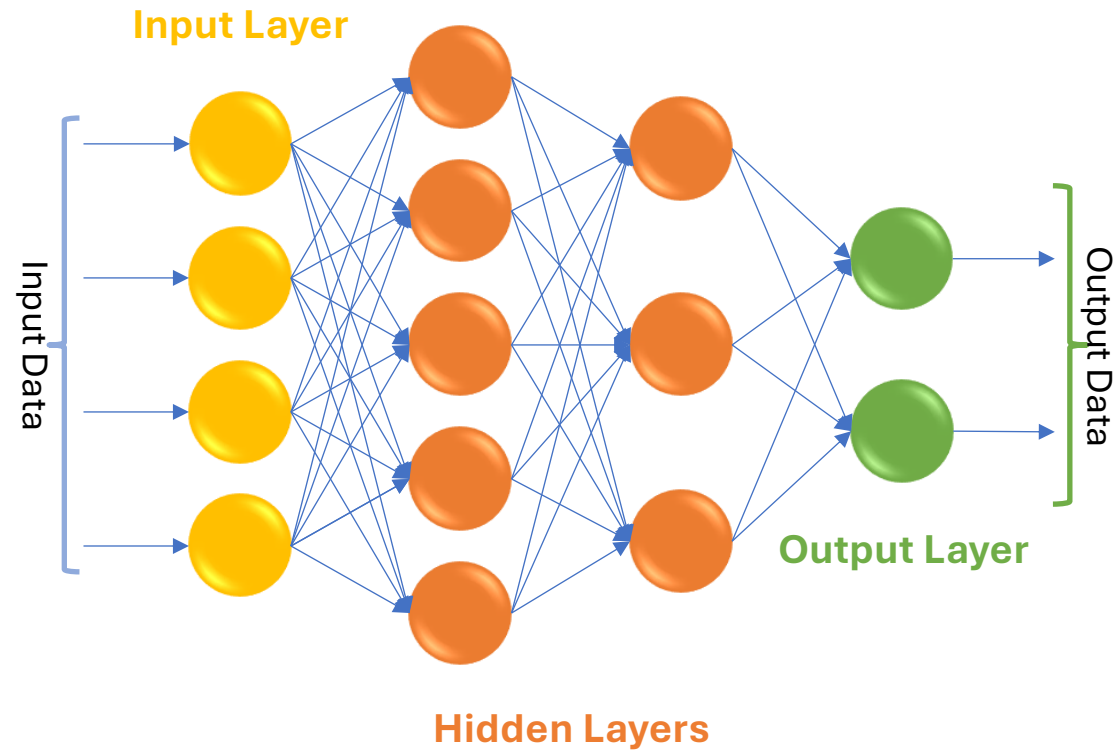
Agile Processes

Project Estimations

Transfer Functions



	= Deming Process		= Transfer Function
	= Deployment Topics		= Deployment Direction
	= Metric		= Control possible



QFD Deming Chain and Multi-Layered Perceptron

Customer
Orientation

Lean
Six Sigma

Agile
Processes

Project
Estimations

Transfer
Functions

- In QFD, layers are not hidden
 - ➔ Each layer is a matrix in itself
 - ➔ It has a sensible meaning
 - ➔ Matrices are connected by discernible topics
- You can easily explain QFD
 - ➔ Although it may look complicated
 - ➔ It's cause-effect relationships
- People won't believe it
 - ➔ They rather trust own gut feelings
 - ➔ Otherwise, much more organizations would be prosperous and address customer needs properly
- In Artificial Neural Networks (ANN)
 - ➔ Layers are hidden
 - ➔ They do not refer to some specific topic
 - Exception: Visual Recognition systems
 - ➔ Advantage: Deep Learning is more flexible
- You cannot explain how an ANN works
 - ➔ Although it may look simply
 - ➔ It's statistics; it's counting
- People believe (currently) in AI
 - ➔ Do they adore authorities?
 - ➔ This might be an evolutionary heritage
 - ➔ Survival of those that followed their leaders

Why isn't QFD Widespread?

- Just a few personal experiences
 - ➔ Made Swissair best airline again
 - ➔ Proposal Centers with 97% success
 - ➔ Drove a Swiss startup to world leader in personalized communication
 - ➔ QFD in Testing of Complex Systems
- Why are they not here, all of them?
- A possible answer
 - ➔ Being a leader is difficult
 - ➔ You need to claim success for yourself
 - ➔ You cannot afford to be perceived as a follower to some method
 - ➔ Up-front investment aren't agile
- Usually, QFD programs were scrapped after management change
- **All approaches will run into same problem with leadership & management**
 - ➔ **This happened already four times in history of AI**
 - ➔ **It will happen again**

Customer
Orientation

Lean
Six Sigma

Agile
Processes

Project
Estimations

Transfer
Functions

Customer
Orientation



QFD as Algebra of Combinators

Lean
Six Sigma



AI as Algebra of Combinators

Agile
Processes



Using QFD for Agentic AI

Project
Estimation

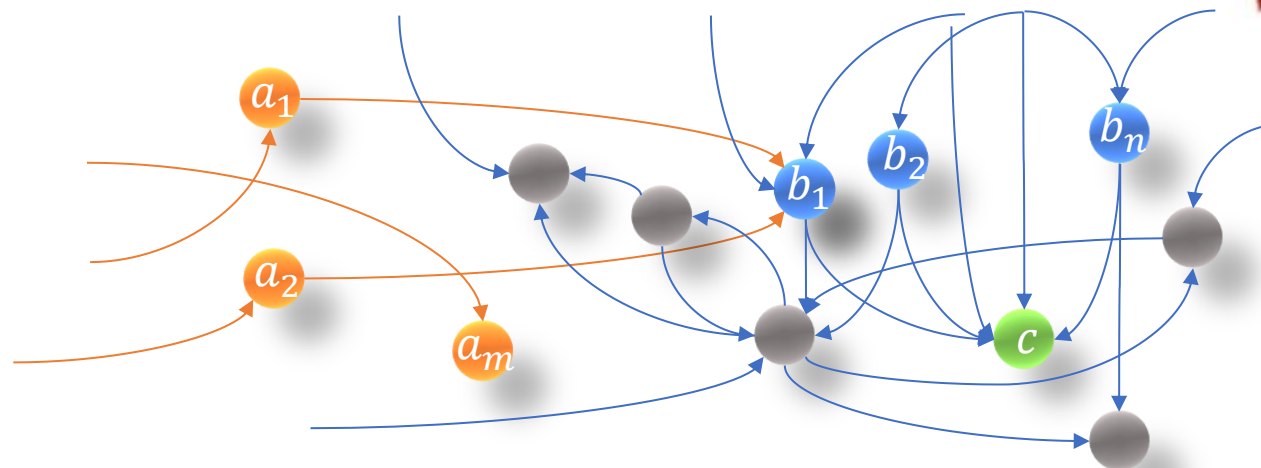


Conclusions

- There is enough evidence that studying the graph model of combinatory logic yields phantastic insights into

- ➔ Brain
- ➔ QFD
- ➔ Testing
- ➔ AI

- This list is not exhaustive



QFD in Decision Making

- QFD is much slower and less attractive for decision making
 - ➔ You cannot detect which important topic you missed
 - Except sometimes thanks to the Convergence Gap
 - If you cannot find the valid Eigenvectors for the matrix
- However, it is much easier to explain decisions taken by some QFD deployment than those with AI
- QFD has no understanding of natural language and does not produce fake images
 - ➔ It is less attractive for writing fake papers, essays, and exams
 - ➔ It is not generative
- Maybe it is possible to combine QFD with some version of an LLM
 - ➔ Provided it does Reference-augmented Generation (RAG)
 - ➔ Instead of hallucinations

Customer Orientation

Lean Six Sigma

Agile Processes

Project Estimations

Transfer Functions

QFD in Product Development

- QFD in Product Development

- ➔ Needs experts
- ➔ Experts are rare species
- ➔ But it's fun & sustainable

- AI in Product Development

- ➔ AI needs many Nuclear Power Plants
- ➔ However, AI only produces similar designs to existing designs
 - Because of statistics
 - Because of 80° Temperature

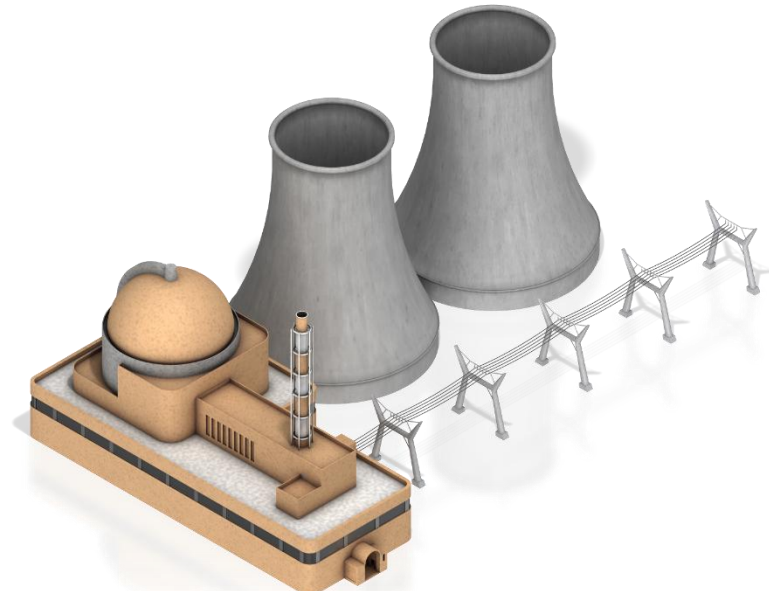
Customer Orientation

Lean Six Sigma

Agile Processes

Project Estimations

Transfer Functions



Questions?

Customer Orientation

Lean Six Sigma

Agile Processes

Project Estimations

Transfer Functions

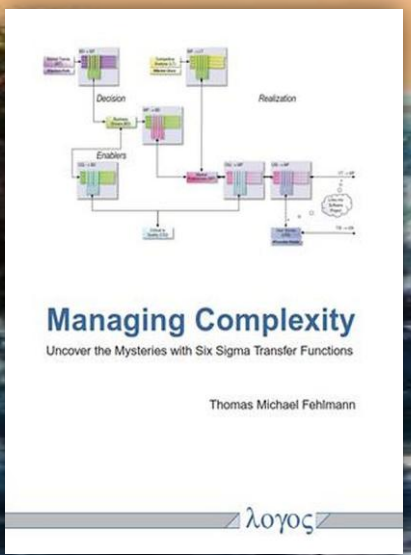


thomas.fehlmann@e-p-o.com

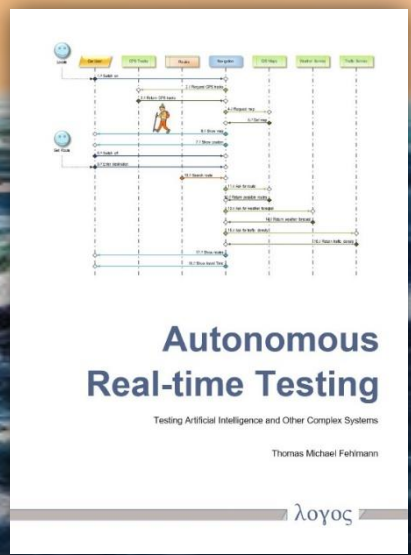


www.linkedin.com/in/thfehlmann

www.logos-verlag.com
Search for "Managing" and "Testing"



Logos Press
Berlin 2016



Logos Press
Berlin 2020