



# The Not-So-Fuzzy Front End:

*Flowing the Voice of the Customer into Functional Requirements*



**Thomas Scripps**  
Scripps & Associates  
Principal

Sponsored by:



# MBB Webcast Series Sponsor: MoreSteam.com

- Founded in 2000
- Trained 435,000+ Lean Six Sigma professionals
- Served over 2,000 corporate customers (including 50+% of the F500)
- First firm to offer the complete Black Belt curriculum online
- Courses reviewed and approved by ASQ and PMI
- Academic Partnerships with Ohio State University, Cal Poly and George Washington University

## Select Customers:



# Today's Program



- Welcome
- Introduction of MBB Webcast Series
  - Ellen Milnes, MoreSteam.com
- Presentation:
  - Thomas Scripps, Scripps & Assoc.
- Open Discussion and Questions



# Today's Presenter



## Thomas Scripps

*Scripps & Associates, Principal*

- 30+ years experience supporting world-class product/process development and improvement initiatives
- Clients have included 3M, Kraft Foods, Pfizer, Caterpillar, and Nestlé Purina
- B.S. in Systems Engineering – Southern Methodist University, M.S. in Statistics – Colorado State University, Master Black Belt

# *The Not-So-Fuzzy Front End – Outline*

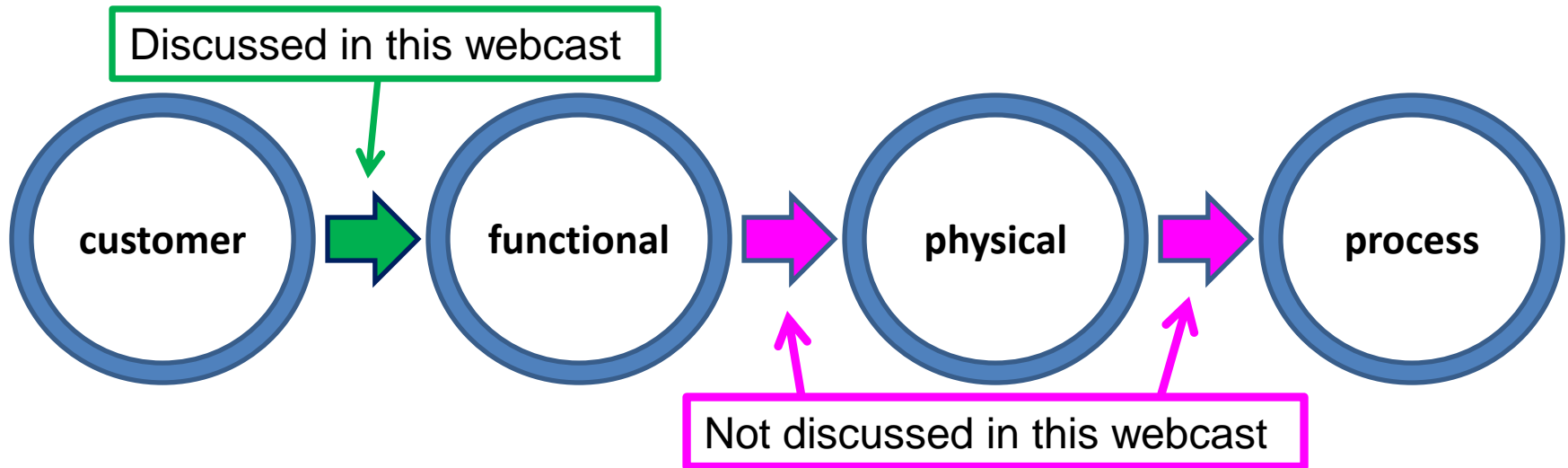
- Scope of this webcast
- Voice of Customer (VoC) Flow
  - From Fuzz to Functions
  - Prioritize Functions
  - Deploy Functions
  - From Functions to Fulfillment ...

# *The Not-So-Fuzzy Front End – Scope*

- We will not discuss the various methods for collecting the VoC
- We will not discuss the various methods for innovative teams to add their voices

# The Not-So-Fuzzy Front End – Scope

- We will discuss how to efficiently and effectively flow the set of voices from their “raw” state to functional requirements.
- The domain structure of Axiomatic Design is helpful here:



# The Fuzzy Front End

- Here, “fuzzy” refers to the difficulty of **extracting underlying customer values** from their voices, and then **translating those values** into requirements.
- These tasks are inherently fuzzy
  - Customers are not consistently rational
  - Customers make decisions based on feelings and emotions – the limbic system



# The Fuzzy Front End

Product/process development people may be inclined to say as a result of this fuzziness:

*“Customers don’t know what they want”,*

but in fact

Customers do know what they like.  
They just cannot articulate a solution!

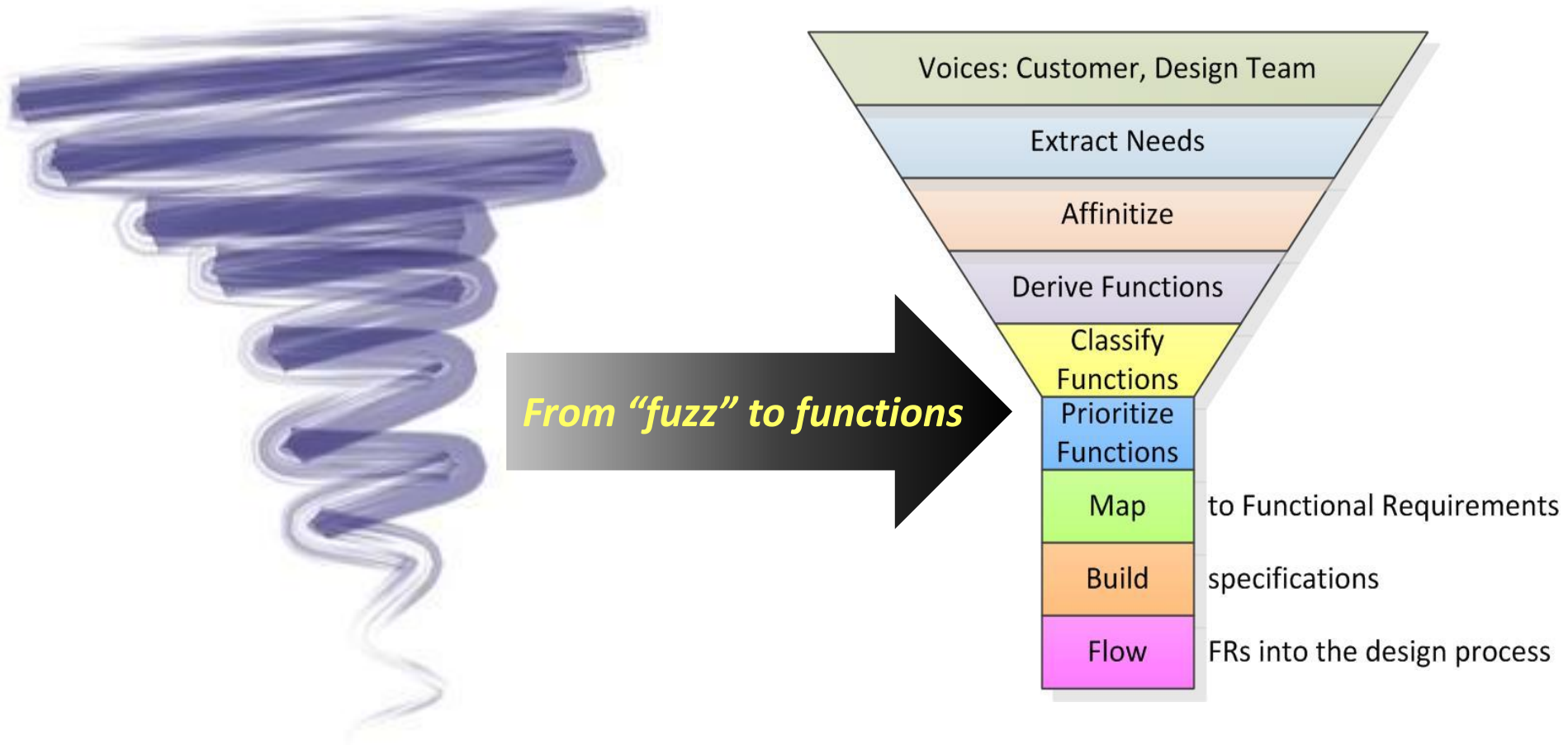
# The Fuzzy Front End

- Design Team's goal is to understand what makes customers happy and what makes them unhappy.



- Through increasing understanding of customers: **likes** and **dislikes**, **priorities**, and **sensitivities**, a design team smoothes the fuzz, thereby focusing their limited resources on the high-value activities.

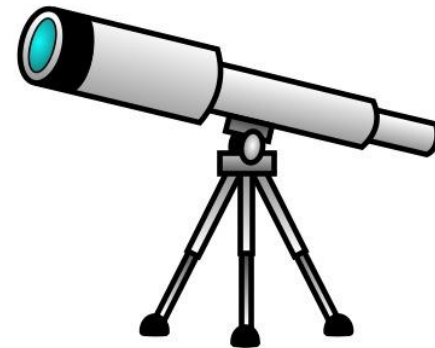
# The Flow of the VoC into Functional Requirements



# The Journey

- To illustrate this logical flow, we use a simple, hypothetical example to which everyone can relate:

The development of an entry-level, backyard telescope, the **VisoCaelus** – “View the heavens”



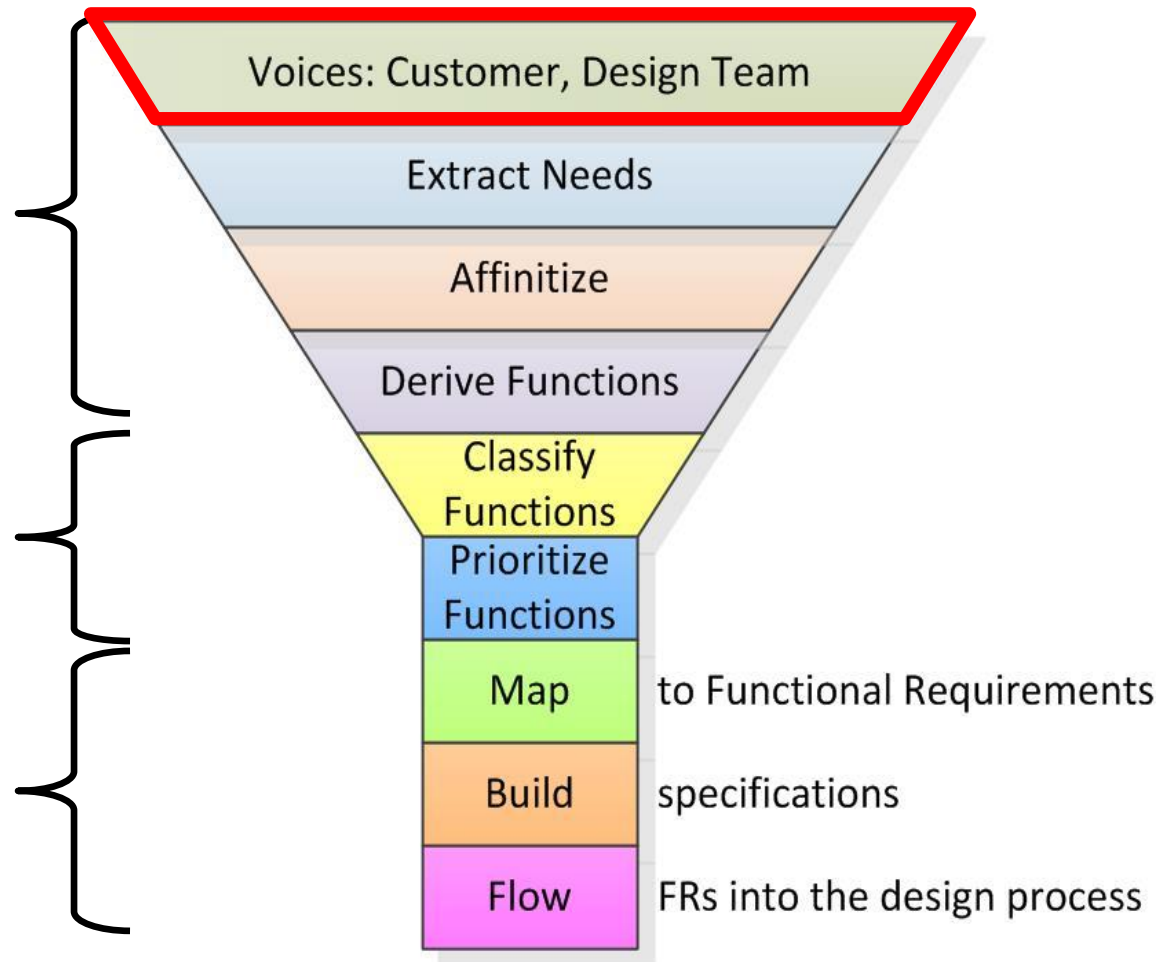
# The Flow of the VoC into Functional Requirements

## VoC Flow

-Fuzz to  
Functions

-Prioritize  
Functions

-Deploy Functions



This icon represents a point in the flow at which it would be logical to schedule a customer touch point.



- The **active Voice of Customer** data from solicited and unsolicited techniques used in market research, service, and quality
- The **latent Voice of the Customer** as derived from the **Design Team**'s knowledge of the market, psychology, anthropology, ethnology, technology

# Gather a plethora of voices

At this point, quantity is valued more than quality

## Solicited

Surveys

Lead users

Focus groups

Interviews

## Unsolicited

Complaints

Field reports

Gemba walks

Benchmarking

Add the DESIGN TEAM's innovations to fulfill latent customer needs!

# VisoCaelus: Verbatim Voices

- *"I want to be able to clearly see what it is that I am looking at"*
- *"Can I wear my glasses to look through this?"*
- *"How can my kids look through this? It's too high off the ground."*
- *"Why do the stars shake so much?"*
- *"I want to see lots of detail on Mars."*
- *"I want one of those computer-controlled 'scopes that finds objects for me."*
- *"Why does the object drift out of the field of view?"*
- *"It's always so difficult for my daughter to really see the object that is in the field of view. She only sees 'space'."*
- *"I want to see Pluto"*
- *"I want to use higher power on that star cluster."*
- *"I want to be able to move this telescope so that the maple tree in our backyard does not block our view."*
- *"I want lots of power!"*
- *"A buddy of mine nearly kicked over a 'scope in the dark."*
- *"How much will it cost?"*
- *"Can I store this 'scope in my garage?"*
- *"I had to crane my neck to look through a telescope last week"*
- *"Can I take pictures through this telescope?"*
- *"The stars are shaking too much in this wind."*
- ...



# *VisoCaelus: Design Team voices*

Latent voices – customers have not explicitly asked for these

- Add something that will prevent condensation on the optics.
- Include a toll-free customer support number 24/7<sup>1</sup>
- Add something that will allow users to take photographs through the telescope with their cell phones

<sup>1</sup>Any product must be developed with consideration for the whole customer experience.

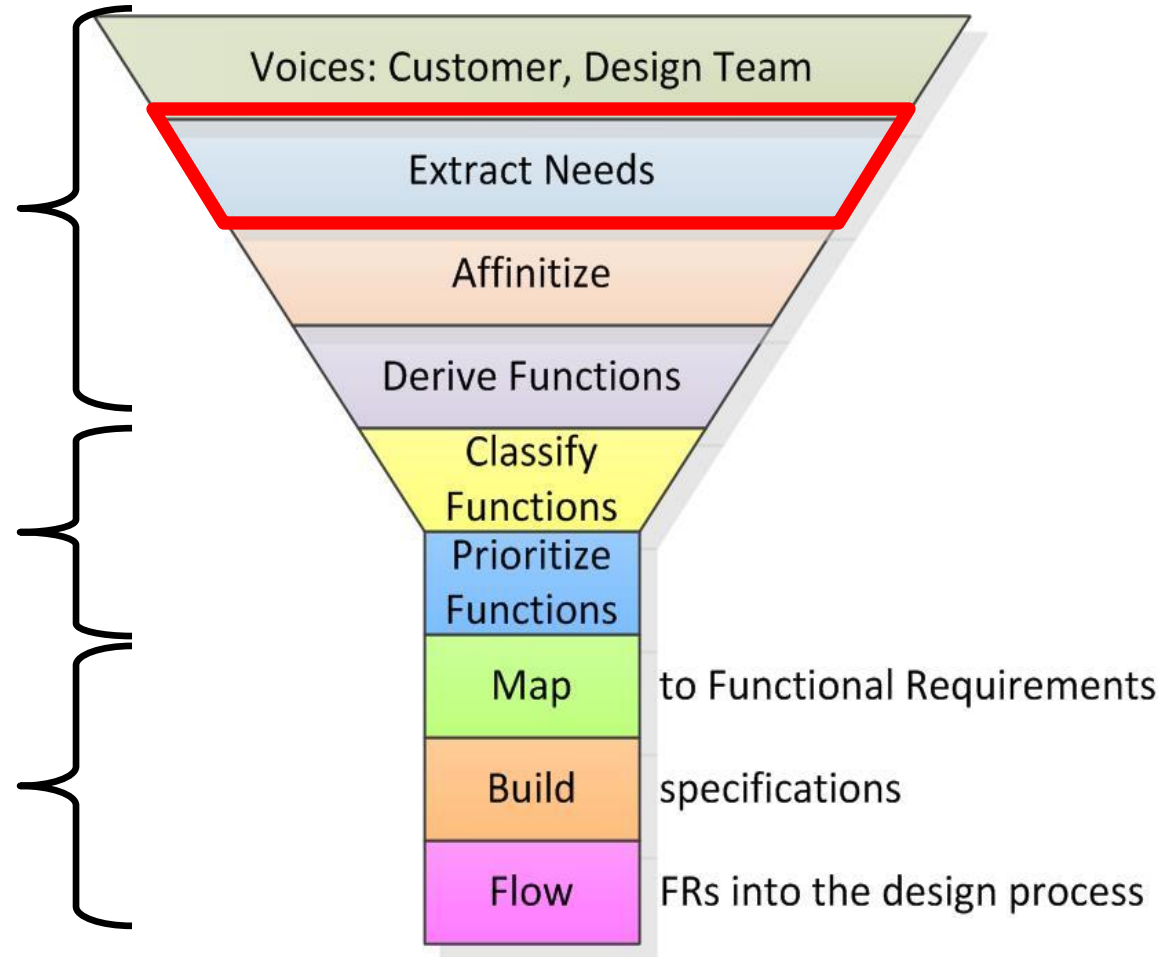
# The Flow of the VoC into Functional Requirements

## VoC Flow

-Fuzz to  
Functions

-Prioritize  
Functions

-Deploy Functions



# Working with verbatim voices

- Verbatim voices
  - Points of **pain** (a symptom of a need)
  - Points of **joy** (a symptom of a need)
  - Comments
  - **Solutions** to pain points
  - **Causes** of pain or joy
  - Irrelevant comments

*NOTE: Most useful are those **bolded***

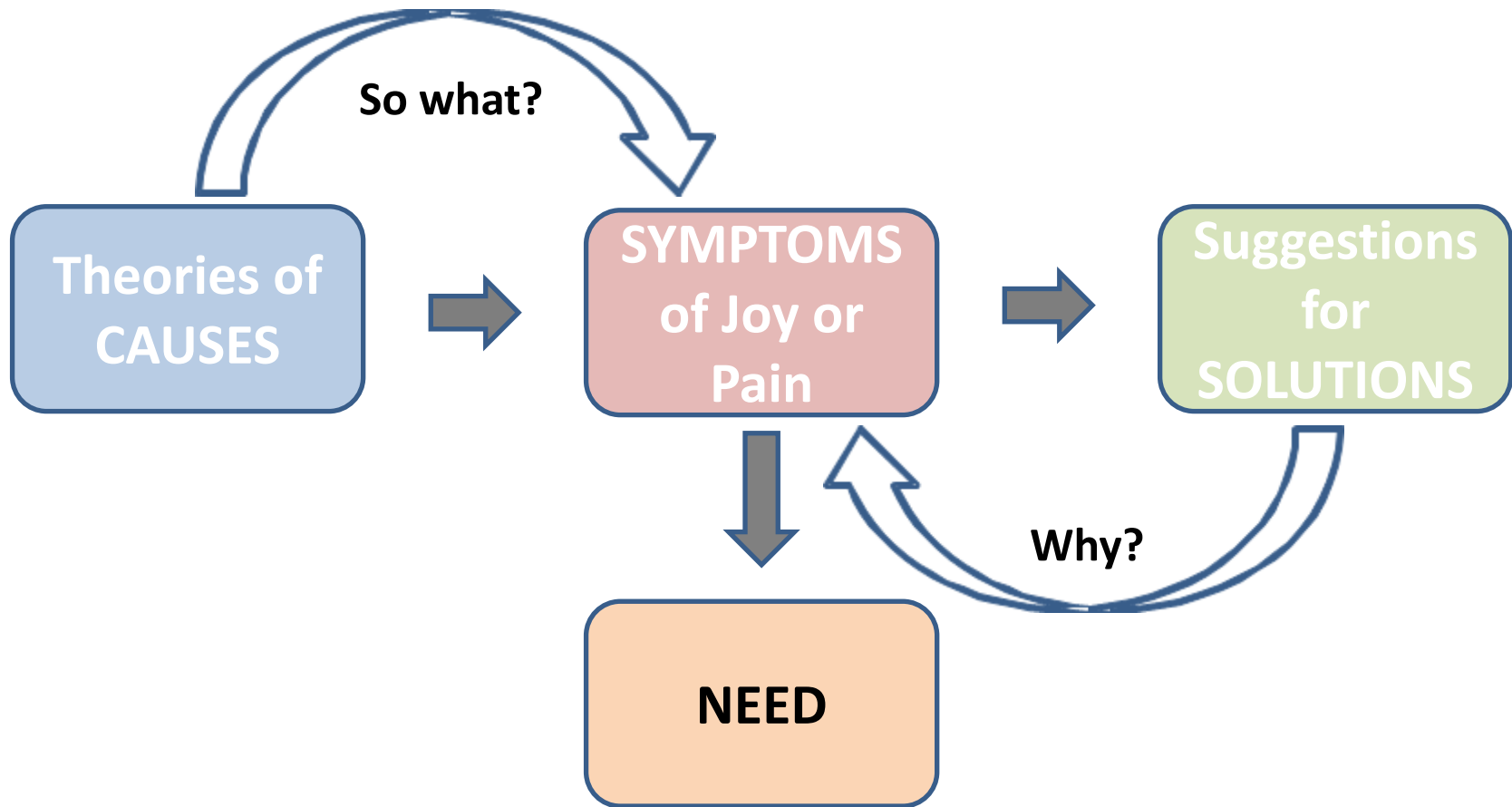
# Extract Symptoms of Needs



- The **Symptom** of pain or joy can be a direct verbatim
  - “As we arrived at the restaurant I really liked that the hostess greeted us by name.” → point of joy
- Derived from a **solution** by asking “What would that do for you?” (i.e. “Why?”)
  - “I need my caffeine” This is a **solution**. Ask “Why?” “I like that it picks me up in the morning.” → point of joy
- Derived from a **cause** by asking “Why do you care?” (i.e. “So what?”)
  - “They don’t have enough parking spots.” This is a **cause**. Ask “So what?” “I often have to park a fifteen-minute walk away.” → point of pain

# Extract Symptoms → Needs

Seek that which **pleases** or **displeases** customers



# Symptoms to Needs



*“As we arrived at the restaurant I really liked that the hostess greeted us by name.”*

Customers need to be recognized as people.

*“I like that caffeine picks me up in the morning.”*

Customers need a boost of energy in the morning.

*“I often have to park a fifteen-minute walk away.”*

Customers need better access to the venue.

# VisoCaelus: Vetting Verbatim Voices

Voice	Classification	Question	Answers (as a point of pain or joy)	Need ("Customers need...")
"Why do the stars shake so much?"	symptom			a steady image
"I need a ladder to look through this!"	solution	Why?	My eyes cannot reach the eyepiece while I am standing on the ground.	to be able to look through the eyepiece comfortably while standing on the ground
"I live in a large city with light pollution."	cause	So what?	I cannot find objects in the nighttime sky	to be able to see through light pollution

- 
- 
- 

etc.

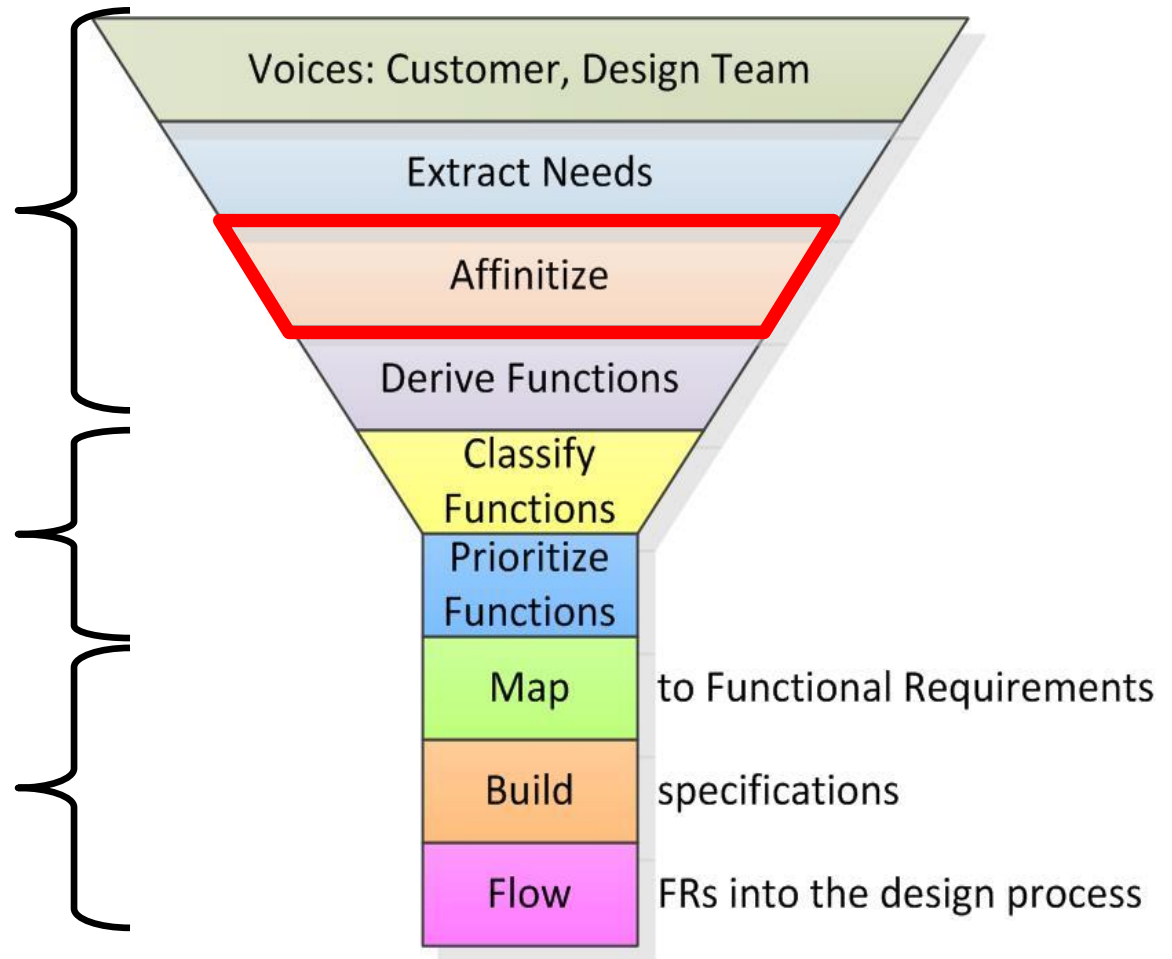
# The Flow of the VoC into Functional Requirements

## VoC Flow

-Fuzz to  
Functions

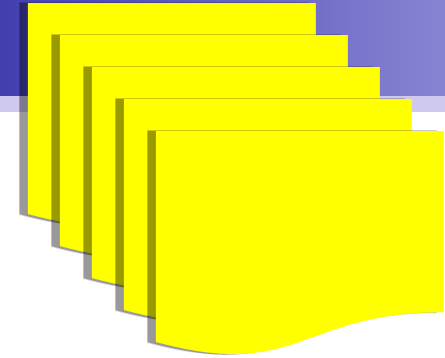
-Prioritize  
Functions

-Deploy Functions





# Affinitize Needs



- One need per Post-Its® or similar
- Group together like needs and provide encompassing word or statement of the summary need
- Goal: as few independent groups (dimensions) as possible without losing any needs.
- It's tricky to group needs without diluting needs.

# VisoCaelus: Needs

A partial sampling of needs:

Need simple  
set up

Need to  
keep objects  
in field of  
view.

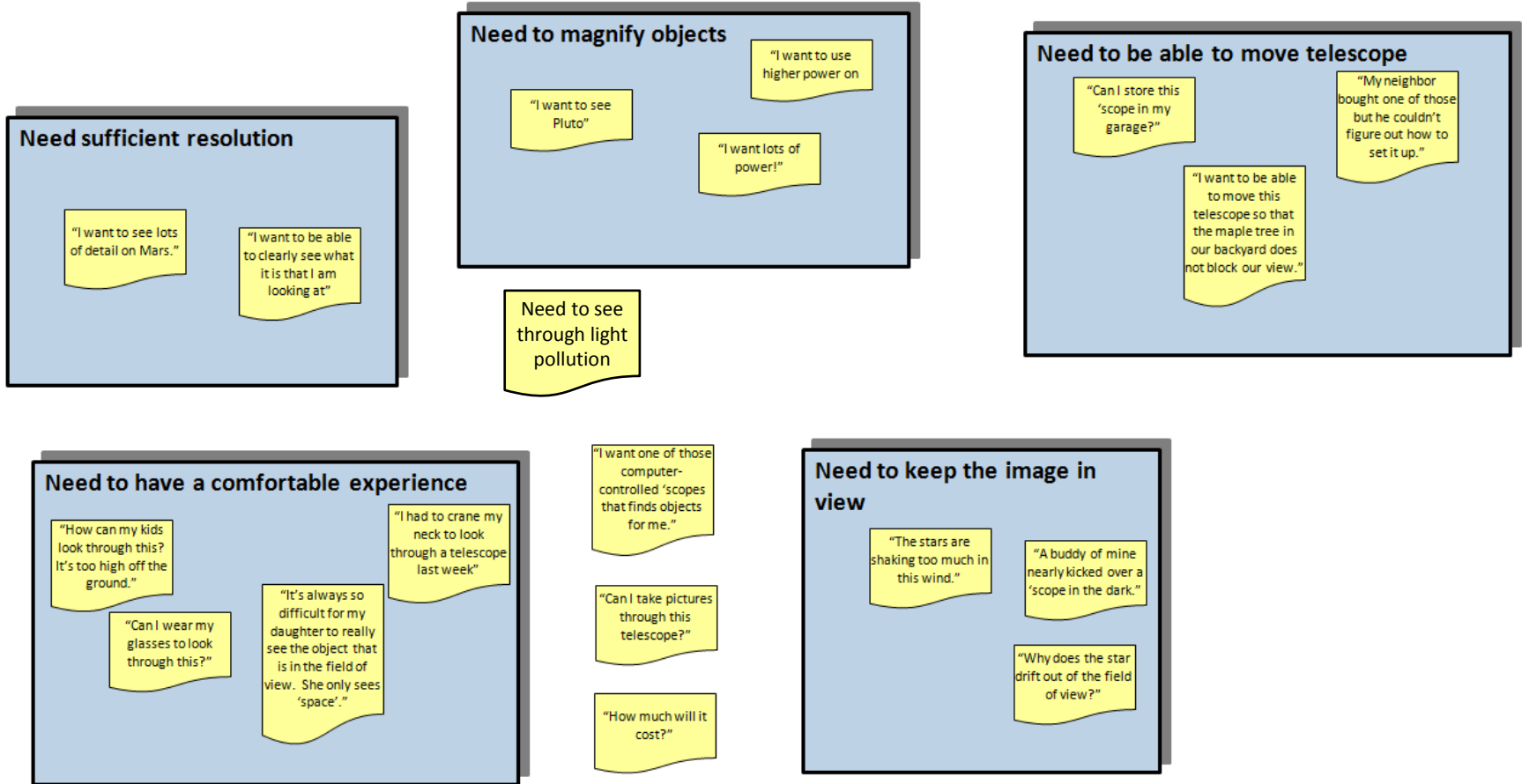
Need to  
stand on the  
ground

Need  
sufficient  
resolution

Need to  
keep  
eyeglasses  
on

# VisoCaelus: Verbatim Needs

- Affinitized needs:



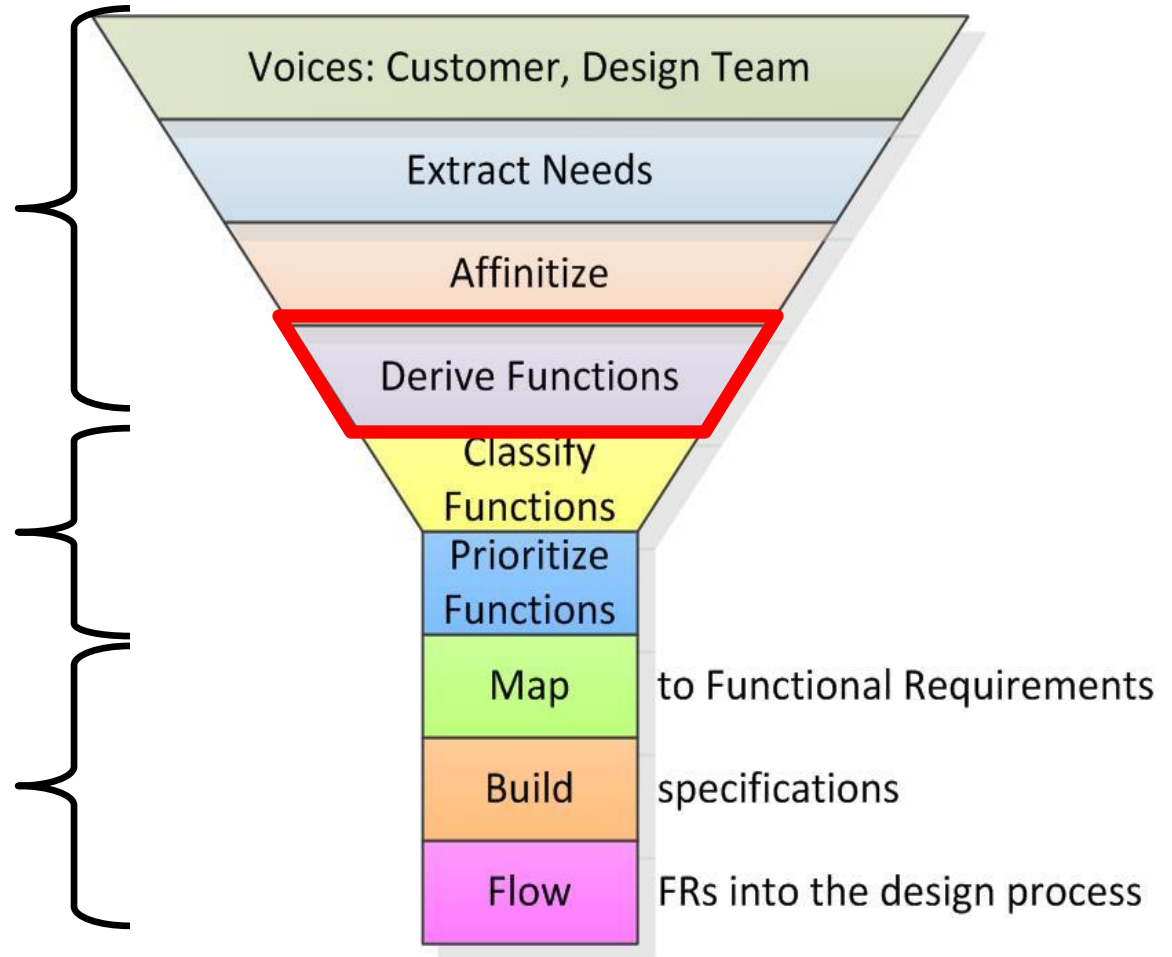
# The Flow of the VoC into Functional Requirements

## VoC Flow

-Fuzz to  
Functions

-Prioritize  
Functions

-Deploy Functions



# Derive Functions

- **Function Analysis** (FA) is the linchpin for flowing the customer voices into the design process
- Lawrence Miles (1948): “The customer wants a function...that is *all* he cares about.”
- Symptoms and corresponding needs express desired or undesired functions
- FA distills needs to a standard format useful for the efficient and effective flow of voices into the design

# Derive Functions

## Cost is not a function

Cost is not brought into the discussions when we are in the functional realm. Cost is brought into the discussions only as we propose design solutions (that include objects and substances or processes for delivery) in the physical realm.

$$Value = \frac{Function}{Cost}$$

# Derive Functions

- Express needs in the canonical form:  
**(verb) (noun)**
- Verb can be
  - **use/active** (preferred) e.g. apply, convert, protect
    - A screwdriver (transmits)(force)
  - **aesthetic/passive** e.g. allow, facilitate, provide
    - An ambassador (exudes)(charm)
- Noun cannot refer to design solutions
  - The value to the customer is in the function, not the solution
  - The noun can refer to a system-level design solution under technological constraints (e.g. a telescope will have optics)

# VisoCaelus: functions

## Voice of Customer to Functions

Raw VoC direct quotes, or as affinitized	Drivers <i>(optional)</i>	Verb	Noun
image needs to have sufficient resolution		resolve	object
need to be able to keep their glasses on	want to wear my eyeglasses	retain	eyeglasses
need to see with the feet on the ground	don't want to use a ladder	create	height
need the telescope to find objects for them		find	objects
need simple set-up		facilitate	set-up
need to keep objects in the field of view		track	objects
need to see from many angles and distances	adds to "comfortable" experience	create	field-of-view
need to be able to see dim objects		create	contrast
need to provide a choice of magnifications		magnify	image
need to easily transport the telescope	light weight	facilitate	transportation
need to be sturdy and resist vibrations	from wind and bumping	support	optics
need to be able to photograph the image		offer	photography



# *Why function analysis*

- Through FA, the design team can create a consistent, cohesive set of independent needs phrased in unambiguous language. This invites greater creativity in solutions.
- This is not easy to do, but the resulting structure simplifies the remainder of the process

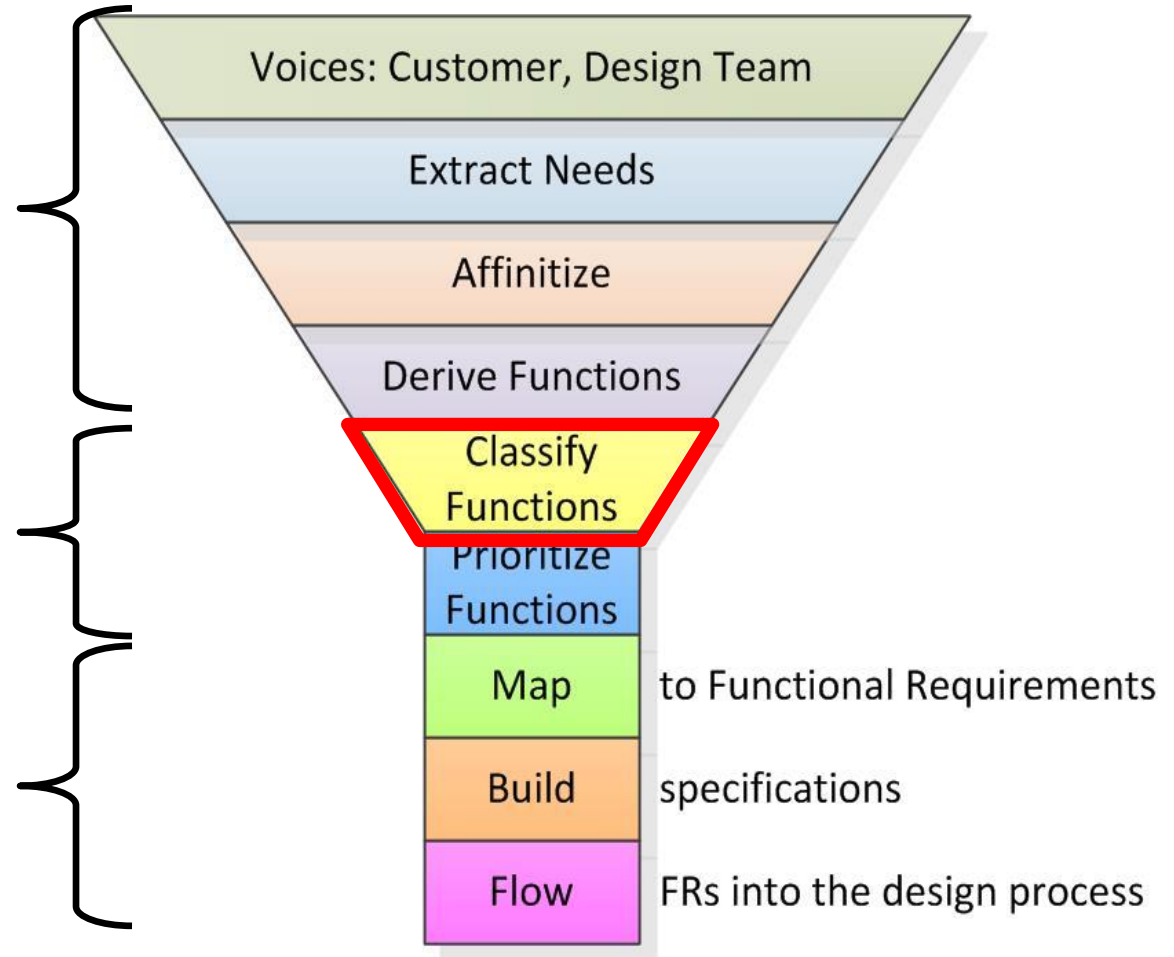
# The Flow of the VoC into Functional Requirements

## VoC Flow

-Fuzz to  
Functions

-Prioritize  
Functions

-Deploy Functions



# Classify Functions

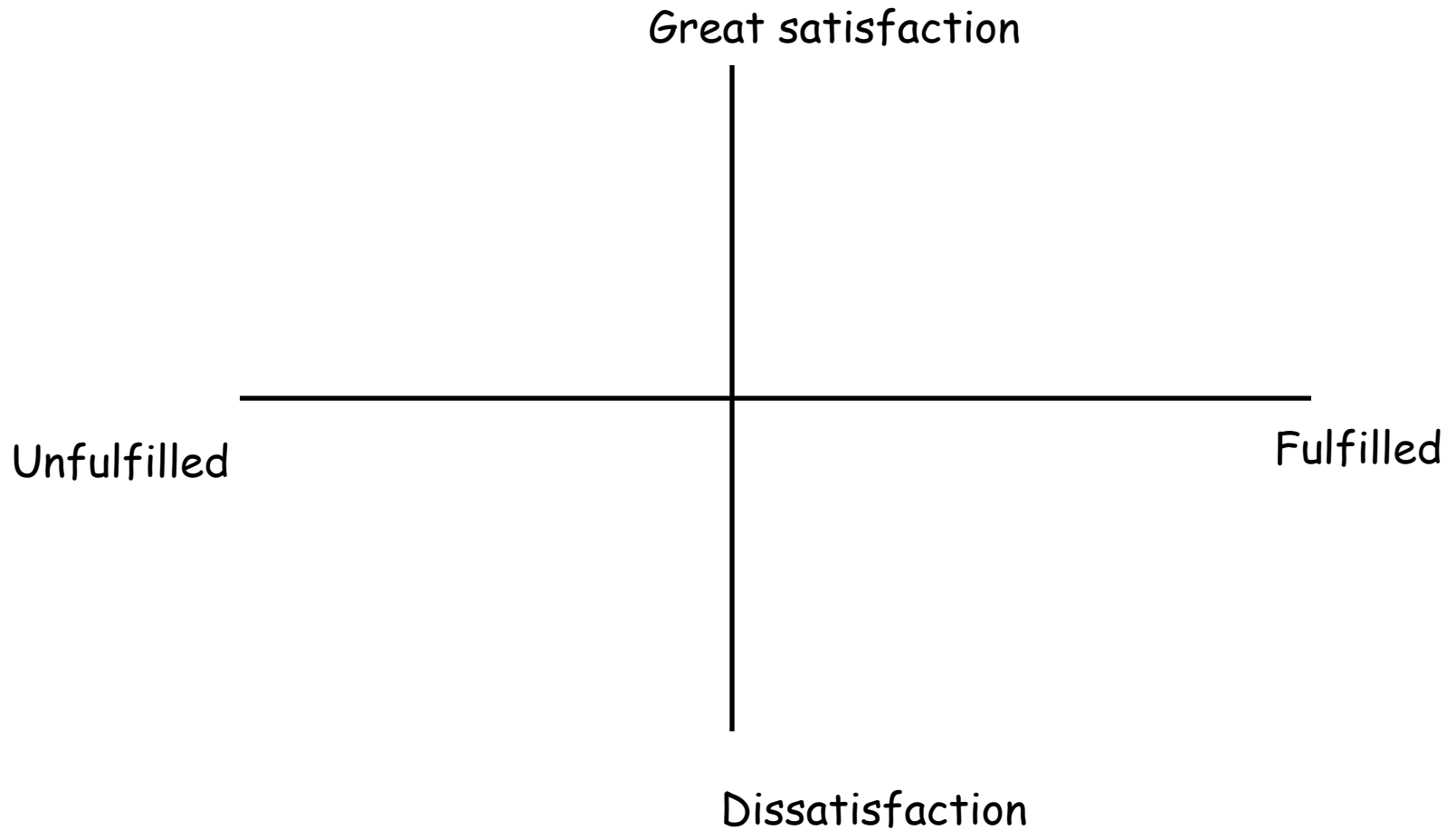
- Classify a given function according to how customers **feel** about getting more or less of the function.
- In this way we understand
  - The role a given function plays in their satisfaction
  - How we can better allocate our limited resources
- The most efficient way to do this is through **Kano** modeling, originally developed by Professor Noriaki Kano (1984)

# Classify Functions - Kano

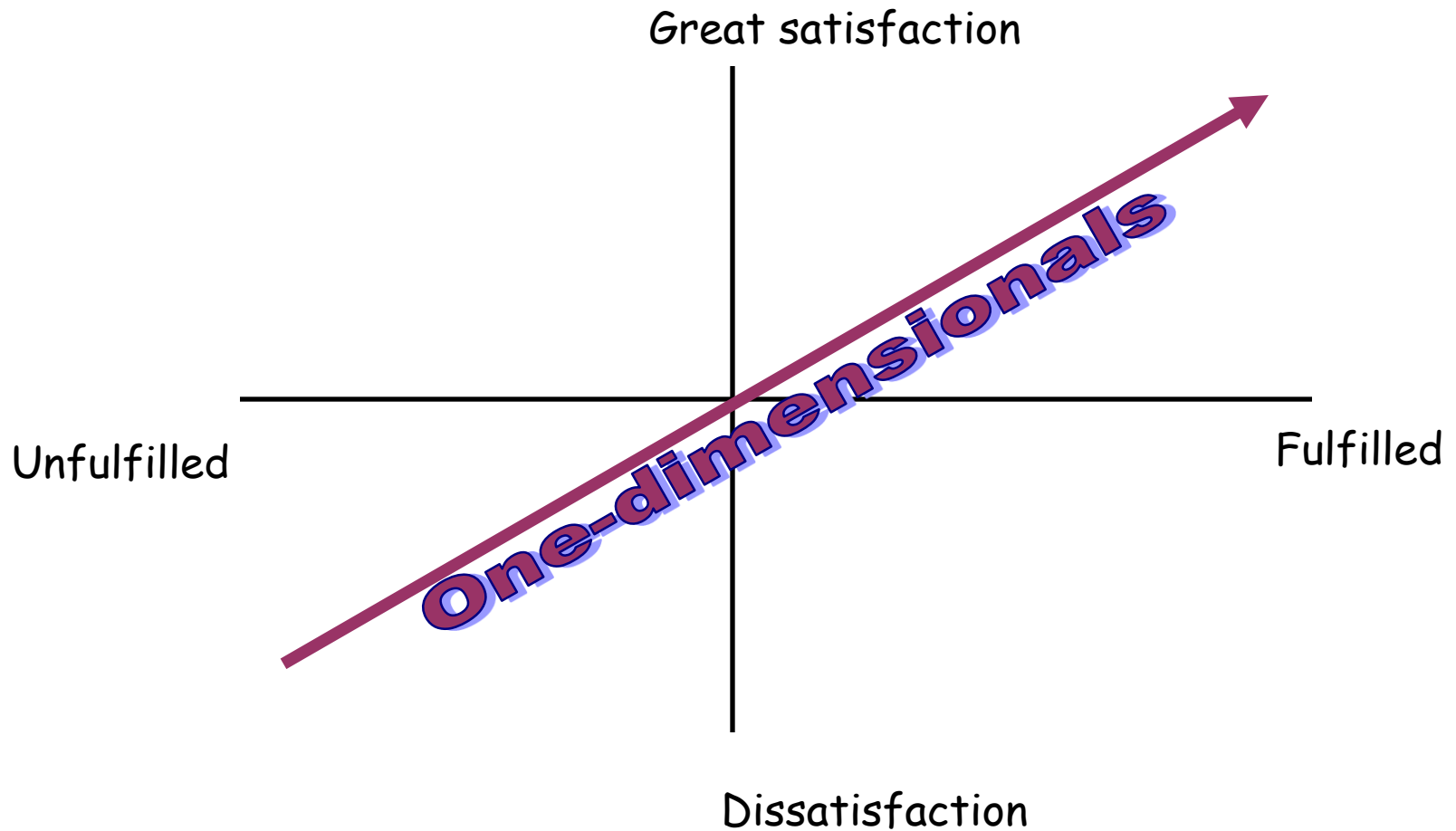
There are four primary classifications of functions

- Those that will delight customers beyond their expectations (e.g. self-driving car) → **Delighter or Attractive function**
- Those that please customers to the extent that we provide them more of that for which they asked (e.g. mpg) → **One-dimensionals**
- Those that displease customers if the function is not met (e.g. including instructions for assembly) → **Must-Bes**
- Those functions about which customers are indifferent (e.g. color of disk drive case) → **Indifferent**

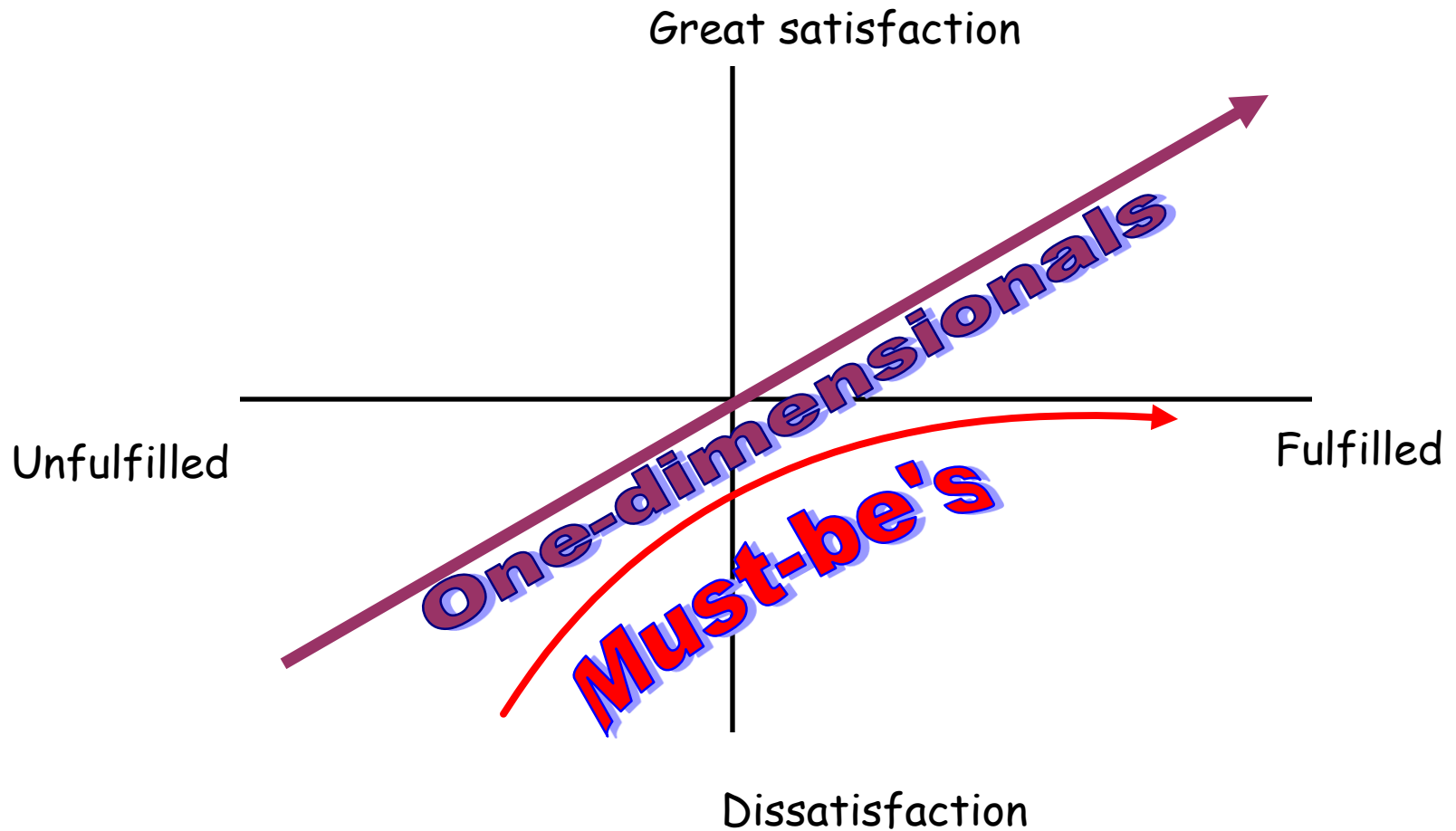
# Classify Functions - Kano



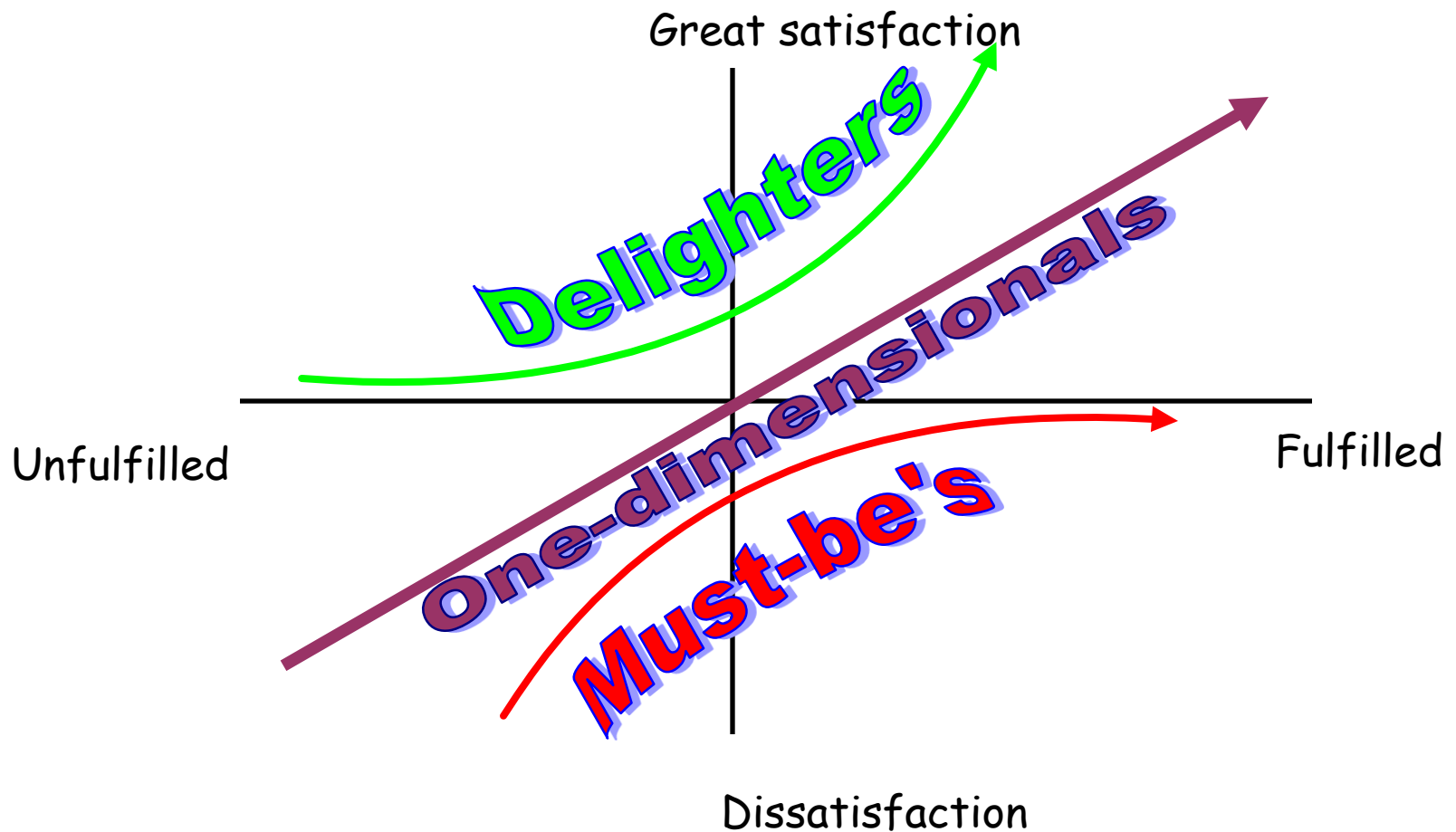
# Classify Functions - Kano



# Classify Functions - Kano



# Classify Functions - Kano





# Classify Functions - Kano

- Here, we consider **Kano classification** on functions
  - Kano classification can also be performed on *features* – after identifying the objects and substances that potentially define the design solution.

# Classify functions - Kano



- For each function, develop a dichotomous pair of questions
  - A “functional”<sup>1</sup> level
    - Example: Keep eyeglasses on to view
  - A “dysfunctional” level
    - Example: Remove eyeglasses to view
- Ask “how would you feel if you had the (functional/dysfunctional) level of function?”

*<sup>1</sup>an unfortunate choice of words because we speak of these activities taking place in the ‘functional’ domain*

# *Kano classification*

Responses to all questions must be one of the following five choices:

- 1 = I would be delighted to find it that way.**
- 2 = I expect it to be that way.**
- 3 = I don't care; I am neutral.**
- 4 = I would not like it that way but I can live with it.**
- 5 = It must not be that way.**

# VisoCaelus: Kano Classification

Each pair of responses is mapped onto a corresponding classification

(Details omitted).

D= Delighter

I = Indifferent

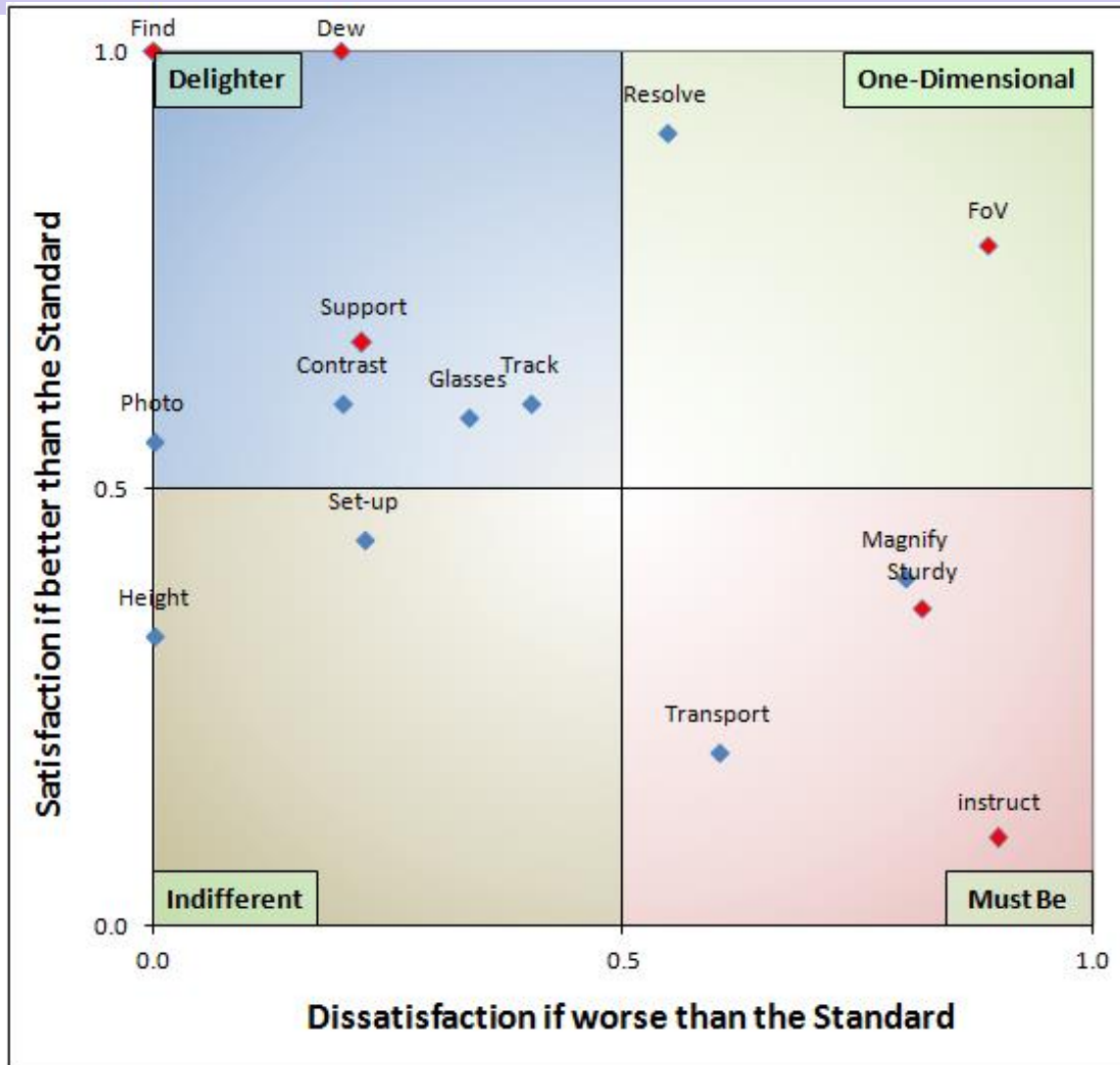
O = one-dimensional

M = Must be

Rolled up, results can be graphed:

[ 2 ] retain eyeglasses		
Functional: Can wear eyeglasses	Dysfunctional: Must take eyeglasses off	Evaluation
1	4	D
1	3	D
3	3	I
1	5	O
2	4	I
2	5	M
1	4	D
1	4	D
1	5	O
2	4	I
1	3	D

# VisoCaelus: Kano Classifications



Red markers denote functions that are statistically significantly within their quadrant.

# Why Kano?

- Insights into customer needs and values to minimize the fuzz – we get to know customers better than they know themselves!
- Optimize the allocation of limited product/process development resources:

# Using the results of Kano Classification

- **Must-Bes** – do these but no more than necessary
- **One-dimensionals** – do these at least as well as your competition
- **Delighters** – do these to create a competitive advantage in the marketplace
- **Indifferent** – do the least expensive alternative

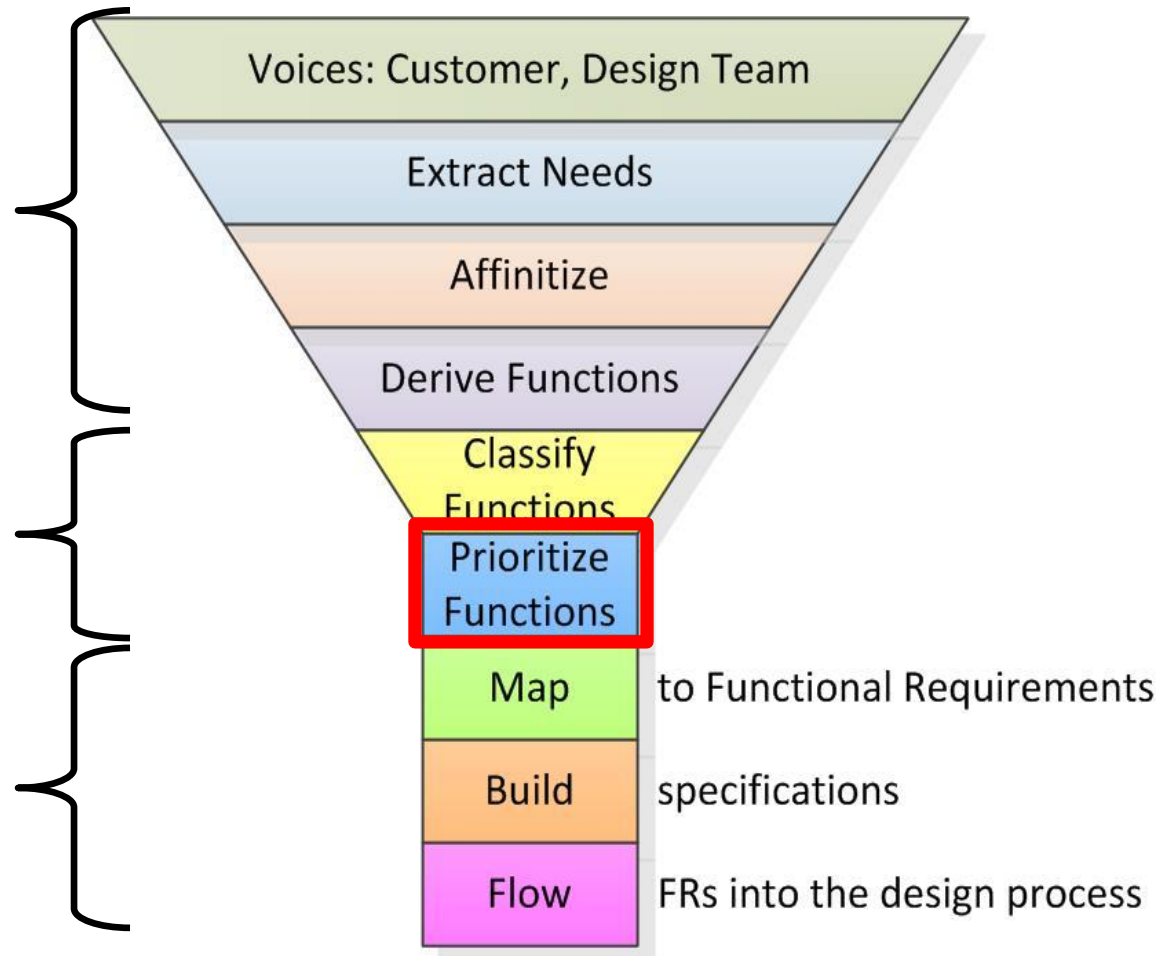
# The Flow of the VoC into Functional Requirements

## VoC Flow

-Fuzz to  
Functions

-Prioritize  
Functions

-Deploy Functions





# Prioritize Functions

- Not all functions (even within the same Kano classification) are equally pleasing
- For efficiency, flow only the higher priority functions into the next activities



- Prioritize functions with **Analytic Hierarchy Process** (AHP) (Saaty, 1970s)
- Make all possible paired comparisons among the functions

*Example for a car: “How important is (provide)(comfort) as compared to (increase)(mileage)?”*

# Prioritize Functions

Rate on a scale:

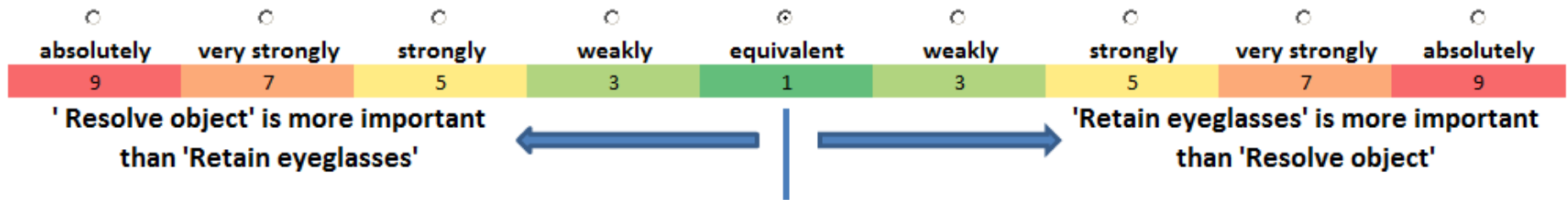
Rating	Degree of preference
1	Equivalent
3	Weakly
5	Strongly
7	Very Strongly
9	Absolute

# VisoCaelus: AHP

10 functions → 45 paired comparisons

How important is 'Resolve object' as compared to 'Retain eyeglasses' ?

(Comparison 1 of 45)



# VisoCaelus: AHP

First comparison from previous slide

...over...

Completed Matrix

Entries: VoC / Function

Resolve object	...is preferred
Retain eyeglasses	
Find objects	
Track objects	
Provide field-of-view	
Provide magnifications	
Facilitate transportation	
Provide support	
Support photography	
Prevent condensation	

VoC/Function	Weight	Weight
Resolve object	0.32	0.32
Retain eyeglasses	0.03	0.03
Find objects	0.04	0.04
Track objects	0.08	0.08
Provide field-of-view	0.09	0.09
Provide magnifications	0.20	0.20
Facilitate transportation	0.04	0.04
Provide support	0.11	0.11
Support photography	0.01	0.01
Prevent condensation	0.07	0.07

Sum of the weights = 1.00

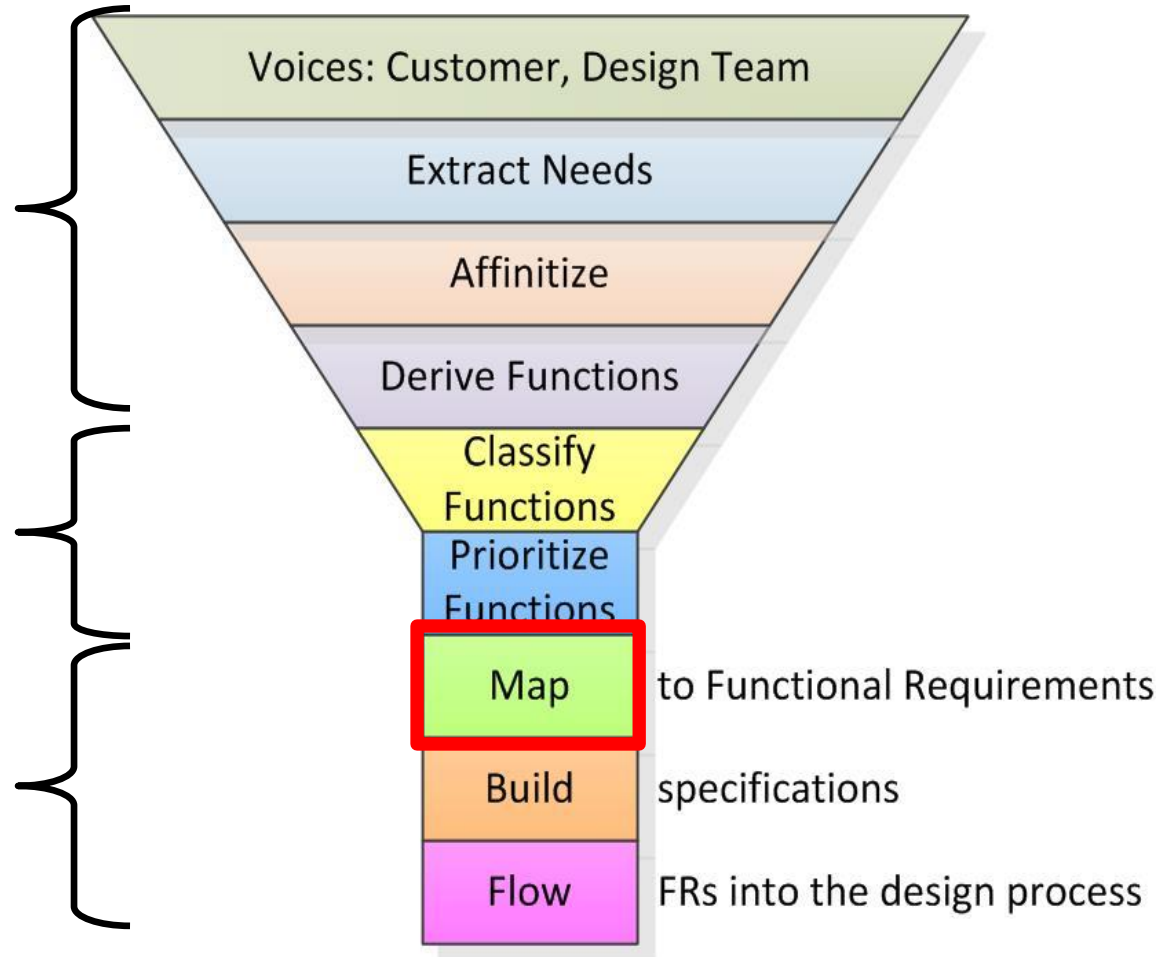
# The Flow of the VoC into Functional Requirements

## VoC Flow

-Fuzz to  
Functions

-Prioritize  
Functions

-Deploy Functions



# Map to Functional Requirements

- Customers want functions “...*under certain conditions and within certain limits.*” [Miles]
- Map functions to **functional requirements (FRs)**
  - Test method
  - Units

# VisoCaelus: Functional Requirements

## Functions to Functional Requirements

Save and Close

Function		Functional Requirement (FR)			Direction	Target	LSL	USL
verb	noun	measure	units	test method				
Resolve	object	resolution	arc seconds	bench	lower is better	0		2
Retain	eyeglasses	eye relief	mm	eyepiece supplier	target is best	tbd	tbd	tbd
Find	objects	error	arc minutes	field	lower is better	0		tbd
Track	objects	error	arc seconds per n	field	lower is better	0		tbd
Create	field-of-view	field-of-view	degrees	eyepiece supplier	higher is better	75		
Create	magnifications	power	x	eyepiece supplier	higher is better	200		
Facilitate	transportation	weight	kg	scale	lower is better	0		tbd
Support	optics	vibration dampening	seconds	tbd	lower is better	0		tbd
Offer	photography	cell phone adapter	yes/no	friction fit tbd	target is best	yes		
Prevent	condensation	moisture	grams	tbd	lower is better	0		tbd

Eye relief = the distance from the eyepiece<sup>1</sup> to the viewer's eyeball.

<sup>1</sup>an eyepiece is assumed here – it is a constraint of technology and not considered part of the creative design solution.



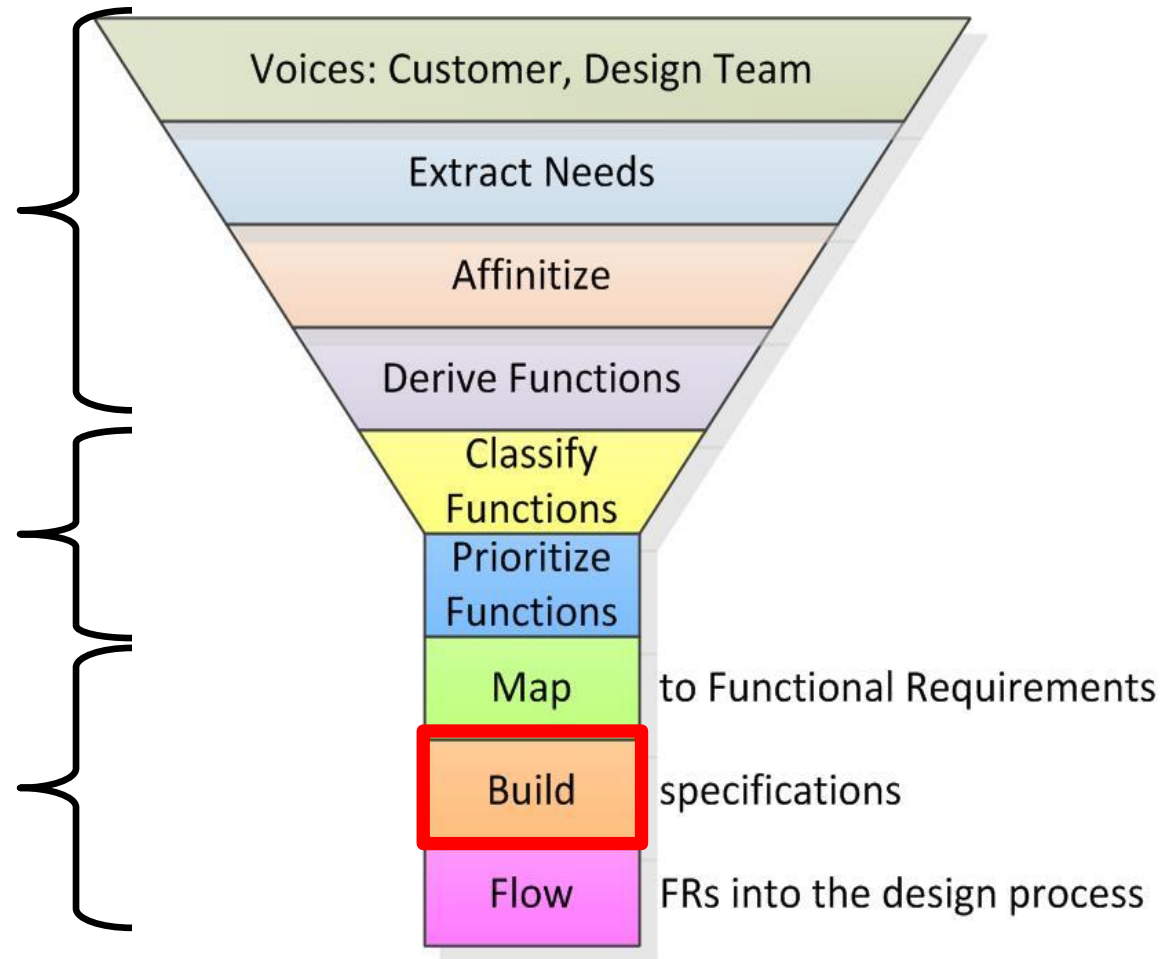
# The Flow of the VoC into Functional Requirements

## VoC Flow

-Fuzz to  
Functions

-Prioritize  
Functions

-Deploy Functions



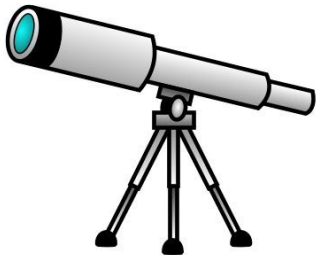


- A statistical exercise to maximize customer satisfaction *vis à vis* functional requirement
- General phrasing “How would you feel if we provided you ‘y’ much function?” Choice is limited to
  - That would be acceptable 😊
  - That would be unacceptable 😞

# VisoCaelus: Unilateral requirement

In practice...

Functional requirement:  
Telescope weight, kg

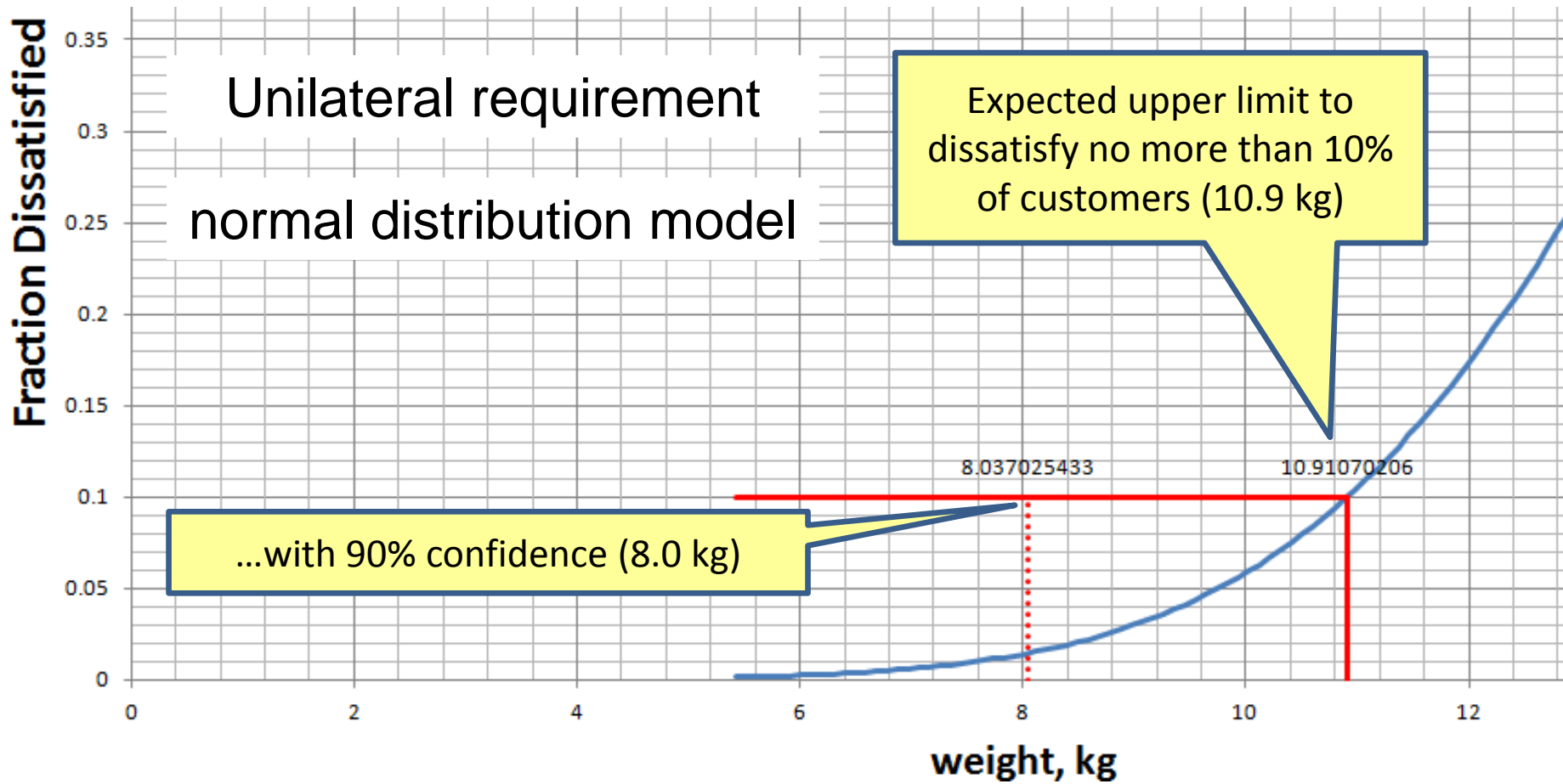


If the weight were \_\_\_ kg or more, that would be too high.

Data

15
17
11
16
18
12
10
17
14
18
20
12

# VisoCaelus: Weight



# VisoCaelus: Bilateral requirement

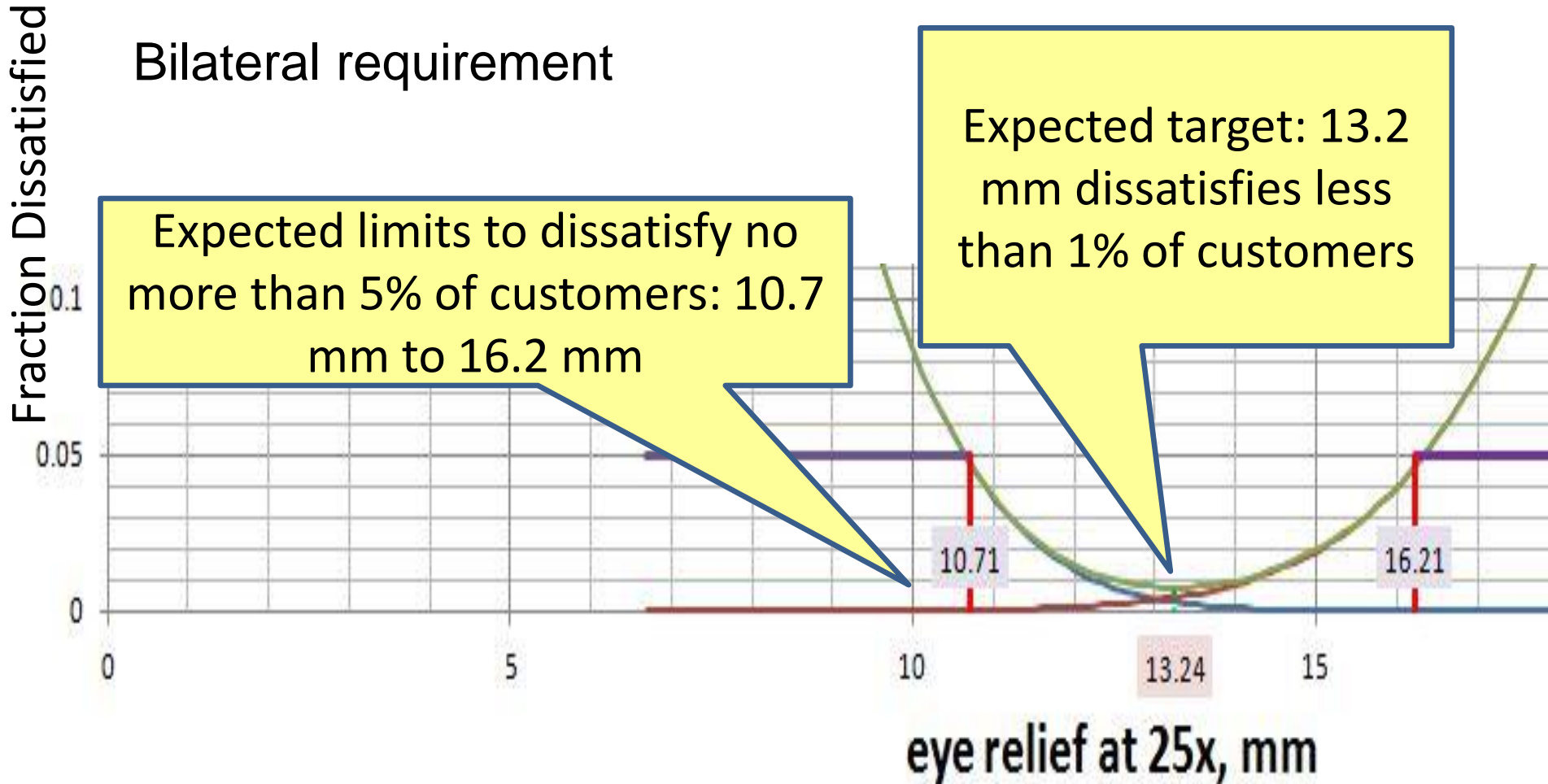
In practice...

Functional requirement:  
Eye relief at 25x, mm

<p>If the eye relief at 25x were ___ mm or less, that would be too low.</p>	<p>If the eye relief at 25x were ___ mm or more, that would be too high.</p>																																		
<p>Data</p> <table border="1"> <tr><td>5</td></tr> <tr><td>4</td></tr> <tr><td>7</td></tr> <tr><td>6</td></tr> <tr><td>5</td></tr> <tr><td>6</td></tr> <tr><td>7</td></tr> <tr><td>5</td></tr> <tr><td>9</td></tr> <tr><td>4</td></tr> <tr><td>10</td></tr> <tr><td>7</td></tr> <tr><td>12</td></tr> <tr><td>9</td></tr> <tr><td>11</td></tr> <tr><td></td></tr> <tr><td></td></tr> </table>	5	4	7	6	5	6	7	5	9	4	10	7	12	9	11			<p>Data</p> <table border="1"> <tr><td>25</td></tr> <tr><td>20</td></tr> <tr><td>20</td></tr> <tr><td>22</td></tr> <tr><td>25</td></tr> <tr><td>20</td></tr> <tr><td>18</td></tr> <tr><td>27</td></tr> <tr><td>25</td></tr> <tr><td>22</td></tr> <tr><td>21</td></tr> <tr><td>23</td></tr> <tr><td>18</td></tr> <tr><td>16</td></tr> <tr><td>14</td></tr> <tr><td></td></tr> <tr><td></td></tr> </table>	25	20	20	22	25	20	18	27	25	22	21	23	18	16	14		
5																																			
4																																			
7																																			
6																																			
5																																			
6																																			
7																																			
5																																			
9																																			
4																																			
10																																			
7																																			
12																																			
9																																			
11																																			
25																																			
20																																			
20																																			
22																																			
25																																			
20																																			
18																																			
27																																			
25																																			
22																																			
21																																			
23																																			
18																																			
16																																			
14																																			

# VisoCaelus: eye relief

Bilateral requirement



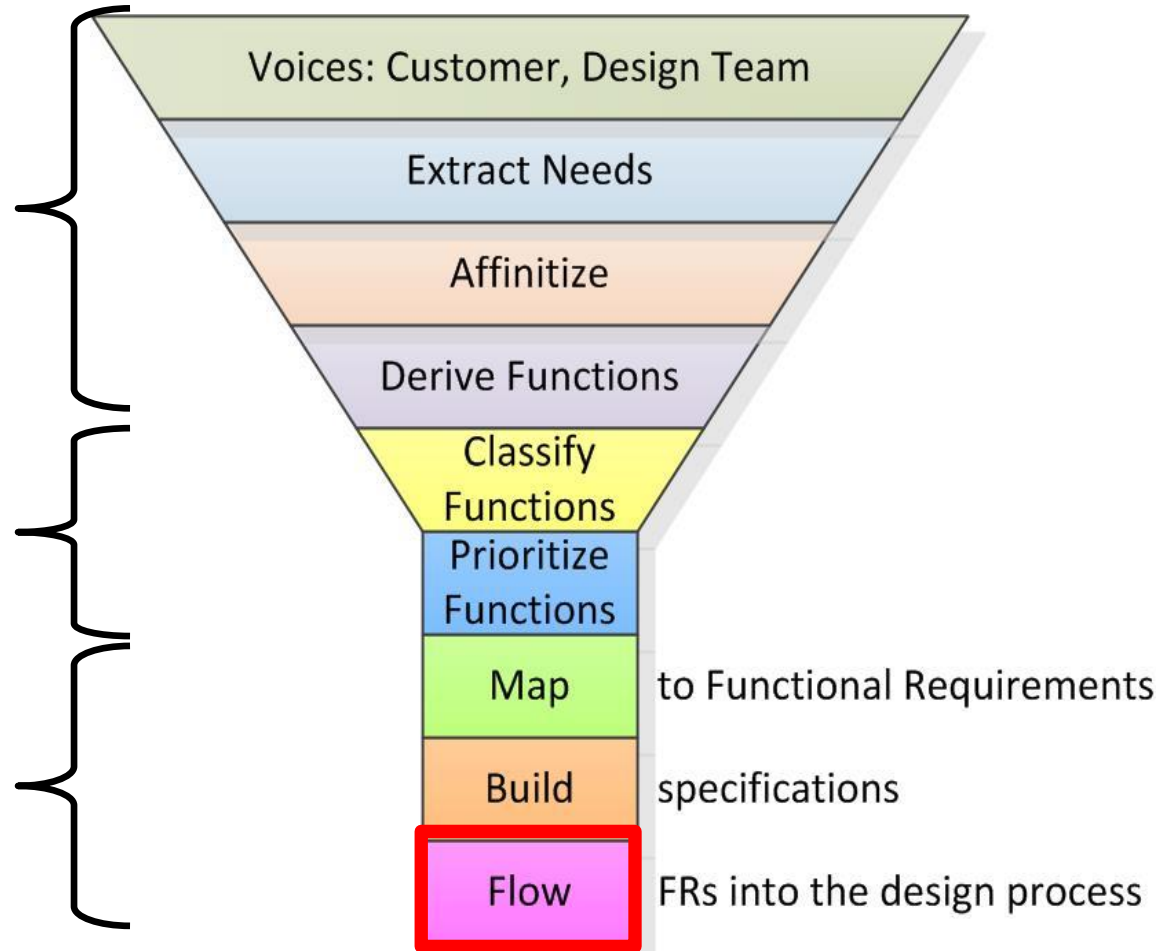
# The Flow of the VoC into Functional Requirements

## VoC Flow

-Fuzz to  
Functions

-Prioritize  
Functions

-Deploy Functions



# *Deploy Voices into Design Process*

- Functions to Functional Requirements
- Top-level House of Quality
- Relationship matrices tend to be more uncoupled (diagonal 9s) in this functional realm



# VisoCaelus: Top-Level House of Quality

Importance = Weight from AHP

HOWs / Functional Requirements (FRs) / CTQs	Kano Classification	Importance	resolution	eye relief	positioning error	tracking error	field-of-view	power	weight	vibration dampening	cell phone adapter	moisture
			L	H	L	L	H	H	L	L	T	L
<b>Direction for Improvement:</b>												
Resolve object	1d	0.32	9							1		1
Retain eyeglasses	D	0.03		9								
Find objects	D	0.04			9							
Track objects	D	0.08				9						
Provide field-of-view	1d	0.09					9					
Provide magnifications	MB	0.2						9				
Facilitate transportation	MB	0.04							9			
Provide support	MB	0.11								9		
Support photography	D	0.01									9	
Prevent condensation	D	0.07										9
<b>Importance of the HOWs (Calculated)</b>			2.9	0.3	0.4	0.7	0.8	1.8	0.4	1.3	0.1	1

# Functions to Fulfillment



As the design moves from the **functional** realm into the **physical/process** realm to identify creative solutions, the same tools can be useful:

- Kano classification on features (present/absent)  
e.g. an eyepiece tray
- AHP on design choices  
e.g. tripod versus post mount

# Conclusion

- Driving all voices (customer and design team) to their underlying **functions** and then using those functions to bridge the customer to the producer improves the **effectiveness** of the flow of the voices.
- **Prioritizing functions** via **Kano classification** and **Analytic Hierarchy Process** allows us to concentrate our limited resources on those functions that matter the most to the customer.
- Using statistically-based targets and tolerances assures the VoC all the way into the design.
- Using this methodical flow minimizes the **fuzziness** at the front end and allows the design team to design products and processes more efficiently. That is, **to market sooner** with **fewer problems**.

# Questions



**Thomas Scripps**  
*Scripps & Assoc., Principal*

*How have you handled ....*

*Have you ever encountered ....*

*Would you explain more how you've approached ....*



# Glossary

- **Feature** - Something that is either present or absent that has the potential to fulfill a function
- **Function** - Something for which a customer is willing to exchange resources (time /money) to please someone (possibly himself or herself). Functions are expressed on a higher level than objects and substances.
- **Need** – Something that pleases a customer expressed as “I need...”
- **Solution** - A characteristic of an object or substance (product) or a process that fulfills a function
- **Symptom** (of a need) - Something that pleases or displeases a customer expressed as a feeling
- **Value** - function / cost

# Master Black Belt Program

- Offered in partnership with Fisher College of Business at [The Ohio State University](#)
- Employs a [Blended Learning model](#) with world-class instruction delivered in both the classroom and online
- Covers the [MBB Body of Knowledge](#), topics ranging from advanced *DOE* to *Leading Change* to *Finance for MBBs*



# *Thank you for joining us*

## ***Questions? Comments about today's program?***

Thomas Scripps – Scripps & Assoc.  
[tom@tascripps.com](mailto:tom@tascripps.com)

Ellen Milnes – MoreSteam.com  
[emilnes@moresteam.com](mailto:emilnes@moresteam.com)



***Watch for info about our upcoming programs***

***Archived presentations and other materials:*** <http://www.moresteam.com/presentations/>