Does Experimental Design Work in Software?

Perry K. Parendo

651-230-3861

Perry@PerrysSolutions.com

Agenda

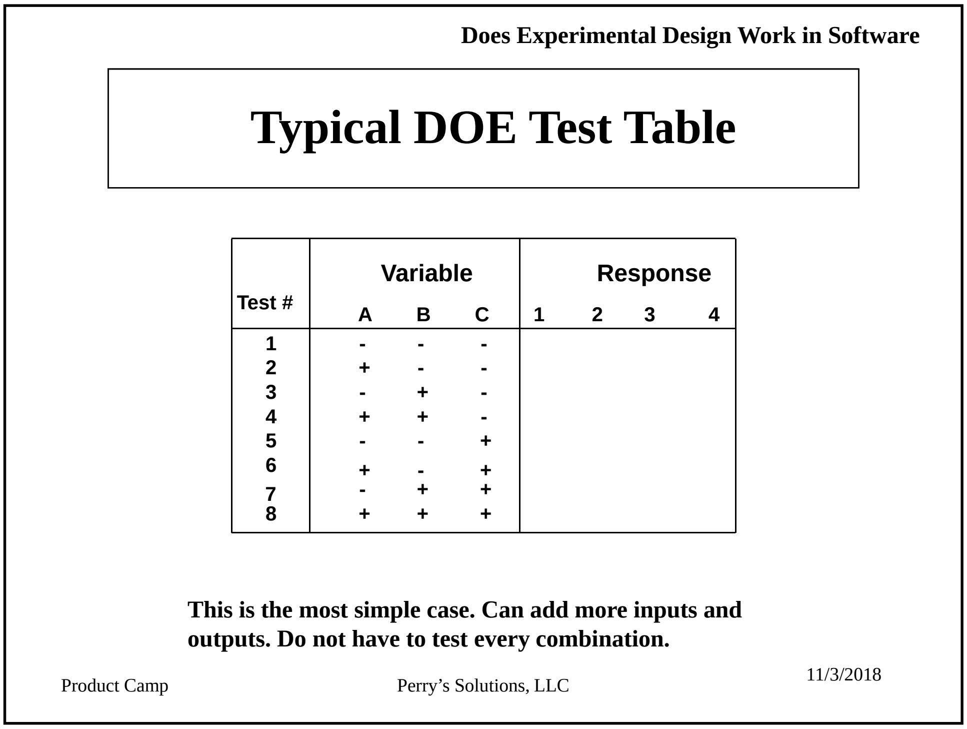
- **Comparing Experimental Design (also known as Design of Experiments, or DOE) for Hardware to DOE for Software**
- **Case Studies – for Software**
  - **User Experience**
  - **Controls**
- **Example Output**

Experimental Design

- Experimental Design (also known as Design of Experiments, or DOE) has been shown to benefit Systematic Innovation
  - http://bit.ly/2Bo7Dad
- However, does it have a place in Software Development?
  - It has been said software is the "Holy Grail" of DOE, if one can figure out how to apply it there
- Most case studies are in mechanical, chemical and electrical applications.
  - Links to these case studies are shown at the end of this presentation.

DOE For Hardware Design

- DOE works when we can identify input and output variables.
  - Input examples: dimensions, materials, timing, pressure, temperature, speed, voltage
  - Output examples: requirements, reliability, cost, schedule
- We need to scope our situation to ensure we have captured a related but manageable test case.



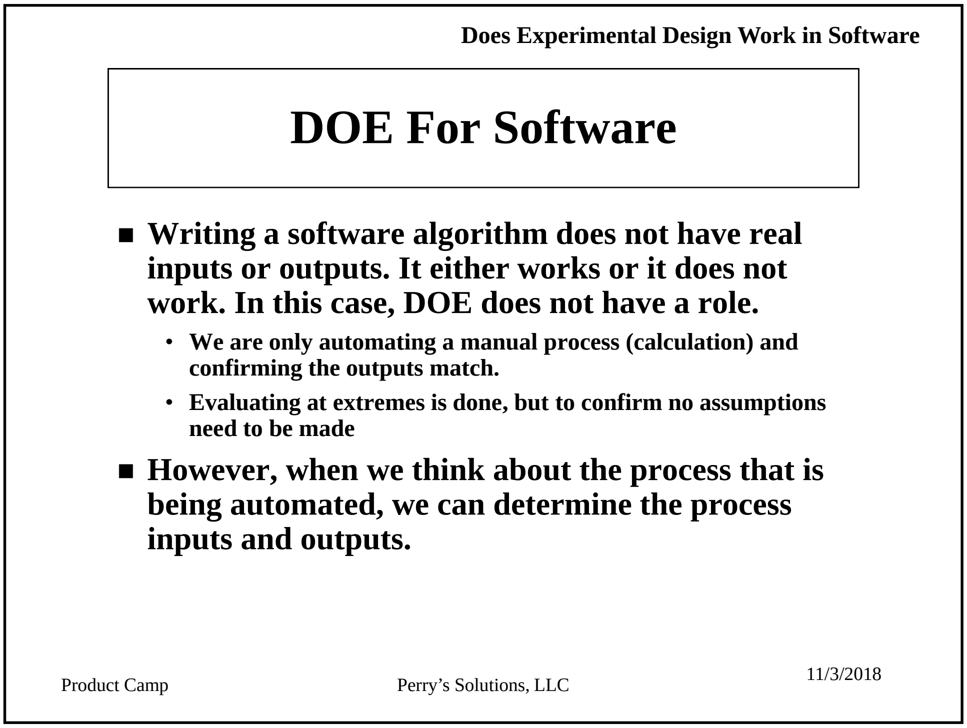
Does Experimental Design Work in Software

Typical DOE Test Table

Test #	Variable			Response			
	A	B	C	1	2	3	4
1	-	-	-				
2	+	-	-				
3	-	+	-				
4	+	+	-				
5	-	-	+				
6	+	-	+				
7	-	+	+				
8	+	+	+				

This is the most simple case. Can add more inputs and outputs. Do not have to test every combination.

Product Camp Perry’s Solutions, LLC 11/3/2018

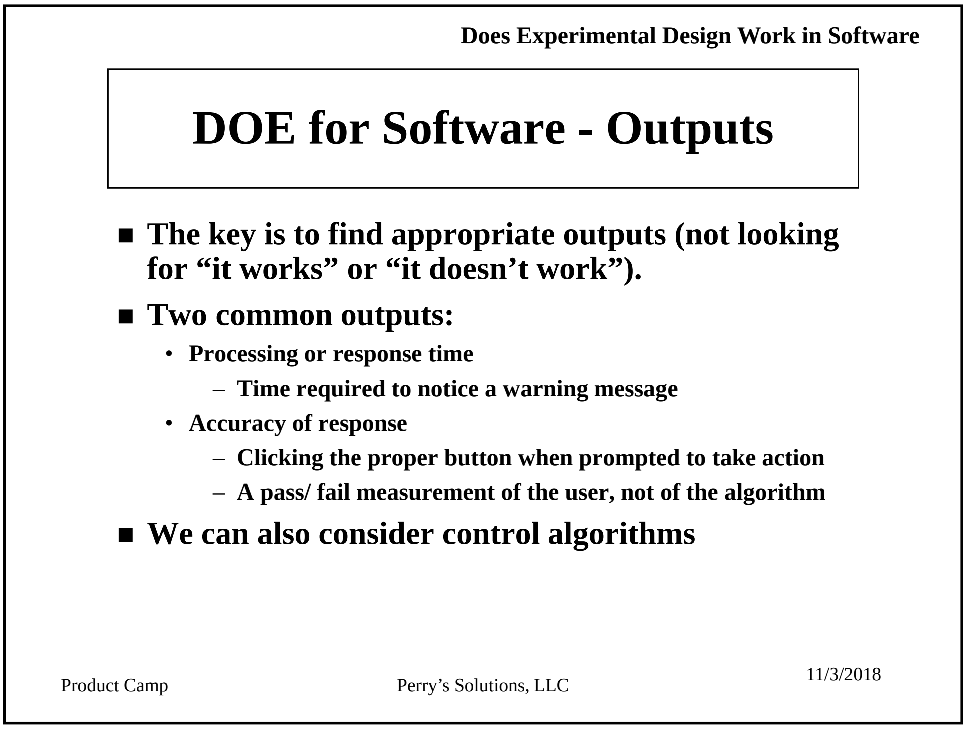


Does Experimental Design Work in Software

DOE For Software

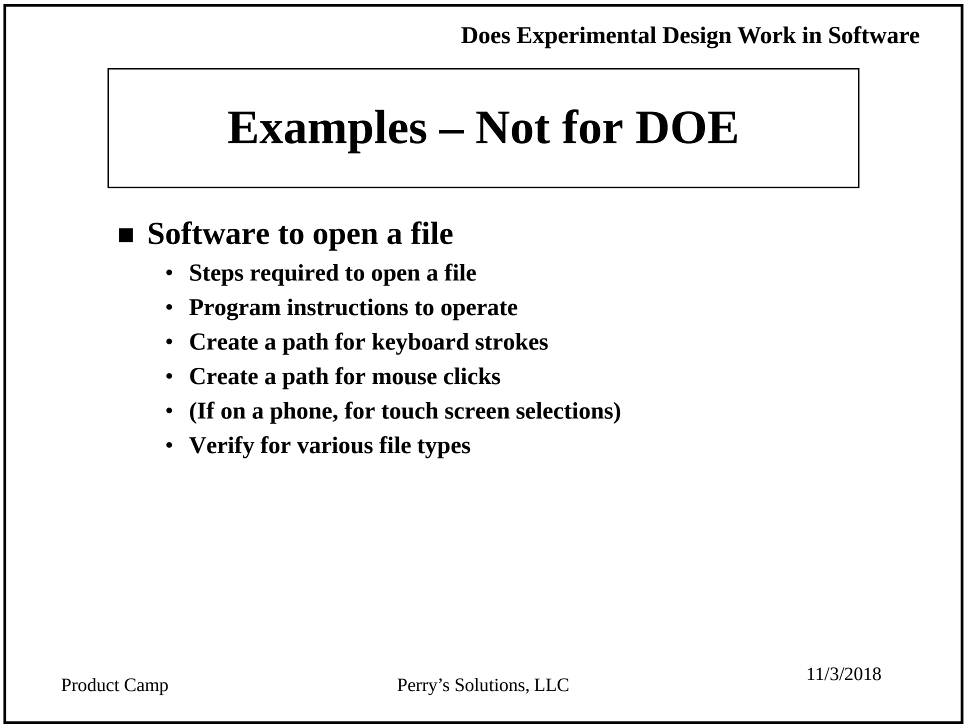
- **Writing a software algorithm does not have real inputs or outputs. It either works or it does not work. In this case, DOE does not have a role.**
  - **We are only automating a manual process (calculation) and confirming the outputs match.**
  - **Evaluating at extremes is done, but to confirm no assumptions need to be made**
- **However, when we think about the process that is being automated, we can determine the process inputs and outputs.**

Product Camp Perry’s Solutions, LLC 11/3/2018



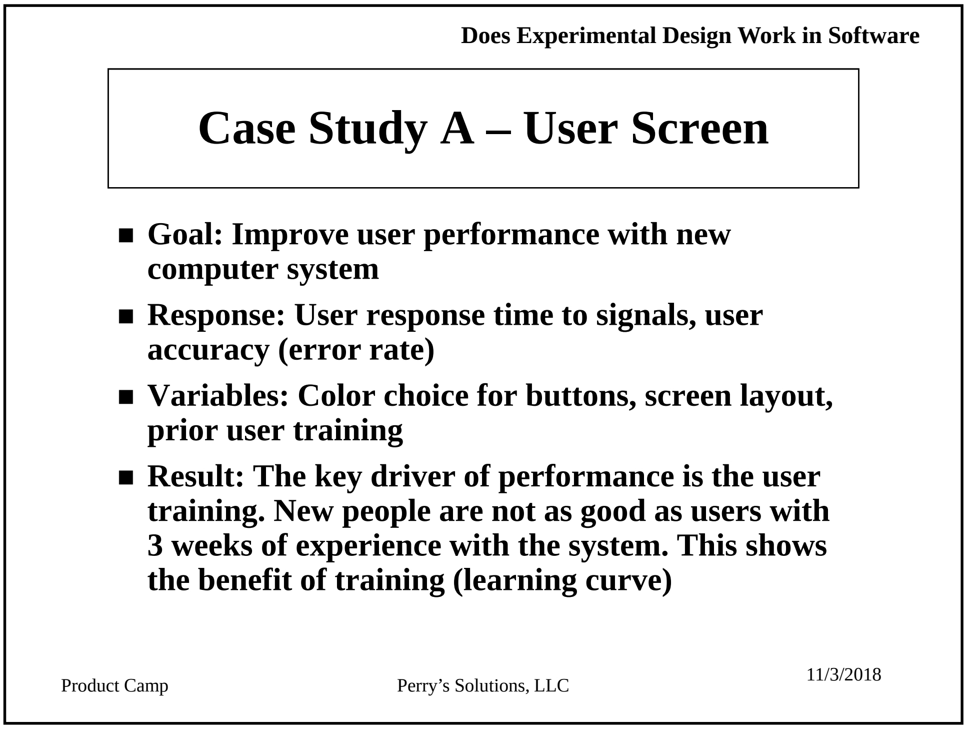
DOE for Software - Outputs

- **The key is to find appropriate outputs (not looking for “it works” or “it doesn’t work”).**
- **Two common outputs:**
  - **Processing or response time**
    - **Time required to notice a warning message**
  - **Accuracy of response**
    - **Clicking the proper button when prompted to take action**
    - **A pass/ fail measurement of the user, not of the algorithm**
- **We can also consider control algorithms**



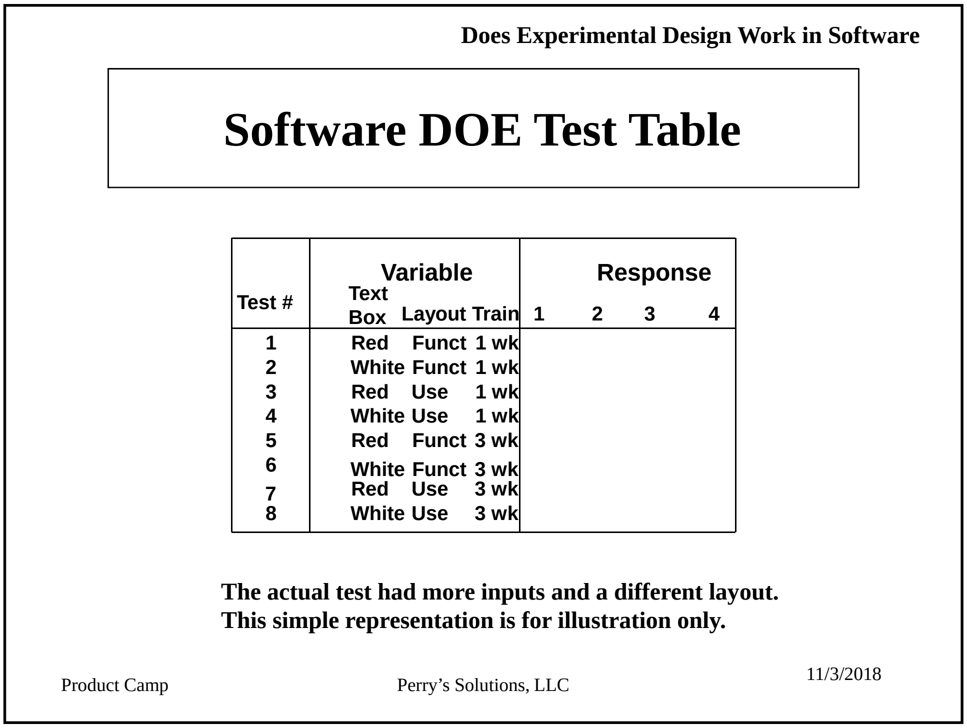
Examples – Not for DOE

- **Software to open a file**
  - **Steps required to open a file**
  - **Program instructions to operate**
  - **Create a path for keyboard strokes**
  - **Create a path for mouse clicks**
  - **(If on a phone, for touch screen selections)**
  - **Verify for various file types**



Case Study A – User Screen

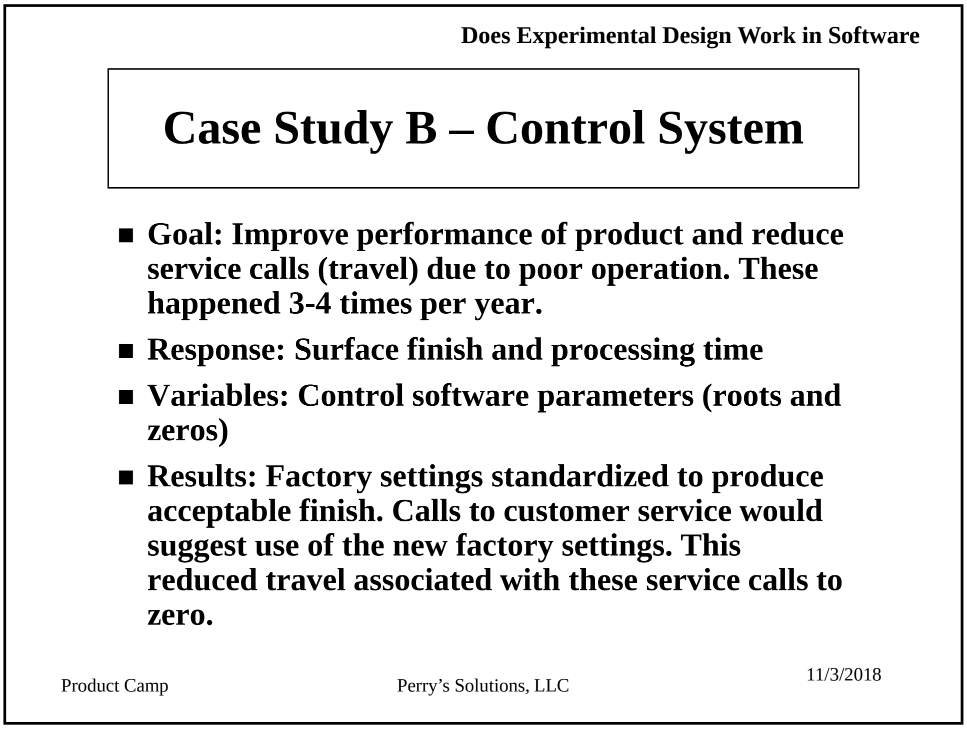
- **Goal: Improve user performance with new computer system**
- **Response: User response time to signals, user accuracy (error rate)**
- **Variables: Color choice for buttons, screen layout, prior user training**
- **Result: The key driver of performance is the user training. New people are not as good as users with 3 weeks of experience with the system. This shows the benefit of training (learning curve)**



Software DOE Test Table

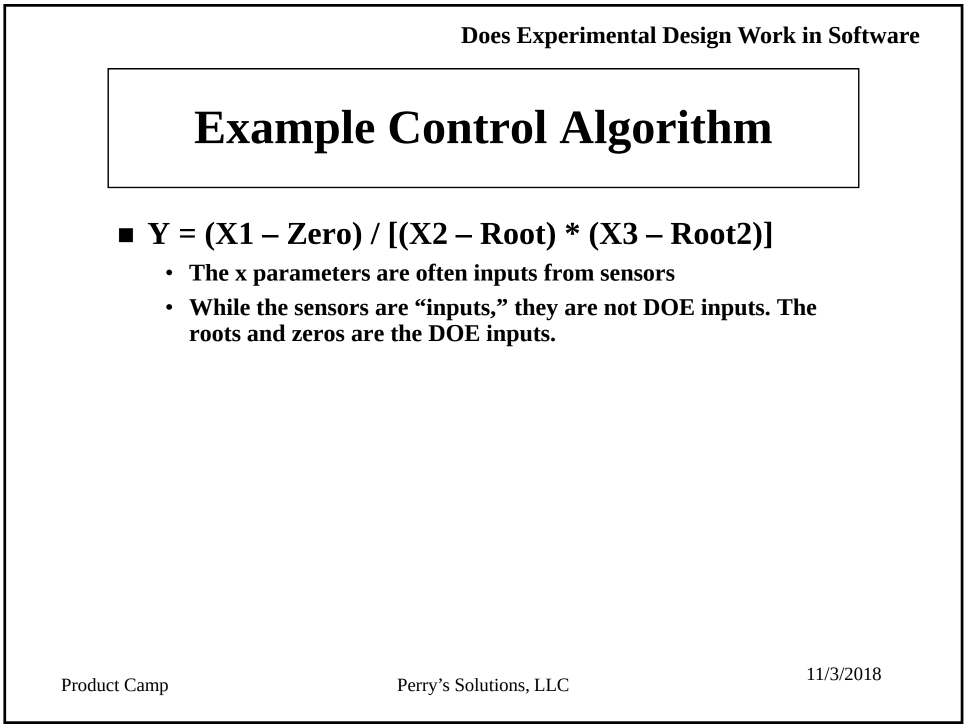
	Variable			Response			
Test #	Text Box	Layout	Train	1	2	3	4
1	Red	Funct	1 wk				
2	White	Funct	1 wk				
3	Red	Use	1 wk				
4	White	Use	1 wk				
5	Red	Funct	3 wk				
6	White	Funct	3 wk				
7	Red	Use	3 wk				
8	White	Use	3 wk				

The actual test had more inputs and a different layout. This simple representation is for illustration only.



Case Study B – Control System

- **Goal: Improve performance of product and reduce service calls (travel) due to poor operation. These happened 3-4 times per year.**
- **Response: Surface finish and processing time**
- **Variables: Control software parameters (roots and zeros)**
- **Results: Factory settings standardized to produce acceptable finish. Calls to customer service would suggest use of the new factory settings. This reduced travel associated with these service calls to zero.**



Example Control Algorithm

- **Y = (X1 – Zero) / [(X2 – Root) * (X3 – Root2)]**
  - **The x parameters are often inputs from sensors**
  - **While the sensors are “inputs,” they are not DOE inputs. The roots and zeros are the DOE inputs.**

Controls DOE Test Table

Test #	Variable			Response			
	Root	Zero	Root 2	1	2	3	4
1	1	2	7				
2	3	2	7				
3	1	5	7				
4	3	5	7				
5	1	2	8				
6	3	2	8				
7	1	5	8				
8	3	5	8				

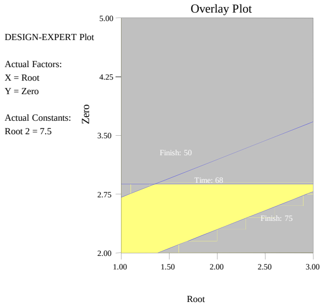
The actual test had more inputs and a different layout.
This simple representation is for illustration only.

How Does It Actually Work

- **After varying the inputs and recording the associated outputs while operating the system, we can observe our desired condition**
- **The example shows 2 outputs and the relevant acceptance criteria for each one**
- **The yellow zone reflects the conditions that achieve all requirements simultaneously**

Control Algorithm Example

- For example: The yellow space is a capable window of operation.



Conclusion

- Software development can benefit from DOE when used appropriately.
  - Design Case Study: http://bit.ly/1la6Olz
  - Manufacturing Case Study: http://bit.ly/1q5e7mk
  - Equipment Setup Case Study: http://bit.ly/15umlcu
- If you want more information, visit our website.
  - www.PerrysSolutions.com
  - If interested, email us to be on our quarterly newsletter where we share recent trends and learning points