Example: Dells "costumize it"

- Entry level laptop, 11 controls:
  (2, 2, 2, 2, 2, 3, 2, 2, 3, 4, 7)
  32,256 combinations

- High end laptop, 13 controls:
  (2, 2, 2, 2, 2, 3, 2, 2, 3, 4, 7, 4, 4)
  2,322,432 combinations

- Test everything in high end?!
  Every combinations takes $\frac{1}{10}$ second = 64 hours

When to test?

- "When testing costs more than not testing, then don't do it."
- Techniques to lower quality-test costs

Solving the problem

- Random sampling
- Pairwise sampling
- N-wise sampling

Random sampling

- Check random inputs = eventually find the bugs
- Level of confidence about a level of quality
- 99% confidence, 99% bug free

Combinations	Tests
100	99
1,000	943
10,000	6,247
100,000	14,627
1,000,000	16,369
10,000,000	16,613
100,000,000	16,638

99% confidence for one million combinations

Quality	Required samples
90%	166
99%	16,393
99,9%	624,639
99,99%	994,027
99,999%	999,940

Pairwise sampling

- Not individual variables, but combinations
- Example

CPU	Memory	Storage
Bargain	Minimal	Large
Consumer	Average	Very large
Power user	Excessive	Huge

- 27 unique combinations, 9 pairwise combinations
- Entry level laptop: 31 tests, High end laptop: 36 tests

N-wise sampling

- Extend pairwise testing to n-wise testing
- $N = 3 \approx 90\%$
- $N = 4$, rarely uncovers more bugs

Nummber of tests for a given N		
N	**Entry level**	**High end**
1	11	13
2	31	36
3	110	179
4	318	749
5	814	2812

Order relevance

- No assumptions of testing order $\rightarrow$ fixed input
- M controls = M! possible sequences
- Every n-wise testing must be tested in M! sequences
- Split up controls in multiple screens to lower number of sequences
- Example: 10 controls is split up in two screens. 10! = 3,6mio. 5! * 5! = 14,400

Order relevance cont'd

- Only need interdependency. N! instead of M!
- Instead look at unique sequences. The binomial coefficient:

$$\binom{n}{k} = \frac{n!}{k! \cdot (n-k)!}$$

Non-unique sequences		
N	**Entry level**	**High end**
1	11	13
2	62	72
3	660	1,074
4	7,632	17,976
5	97,680	337,440

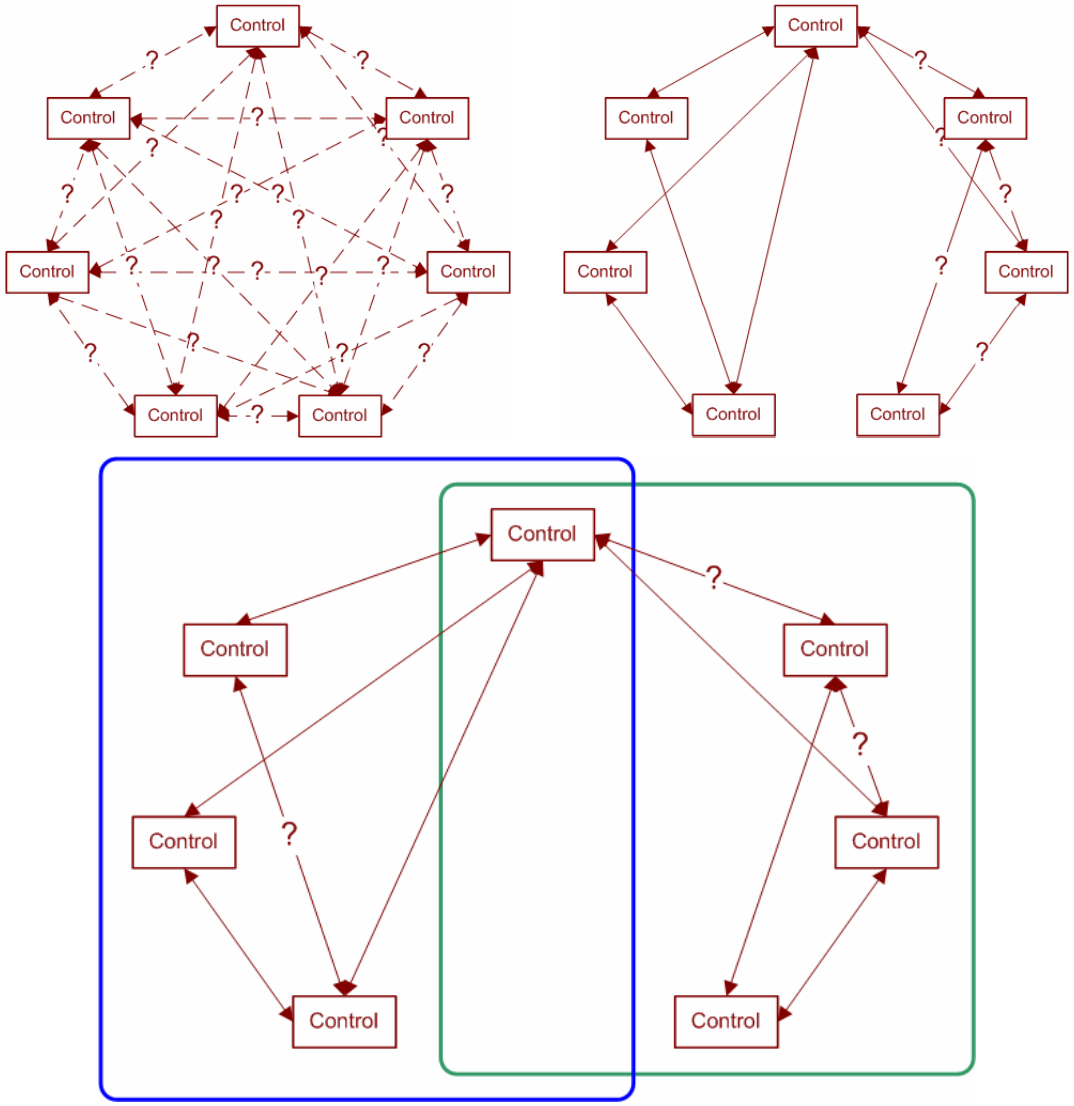
Unique sequences		
N	**Entry level**	**High end**
1	11	13
2	55	78
3	165	286
4	330	715
5	462	1,287

Applying knowledge

- Apply knowledge of application, find the dependencies for each control
- Ideally, this would result in two or more disconnected graphs
- Example:

  7 controls with each 5 options

N	7 controls		4 + 4 controls	
	Order doesn't matter	Order matters	Order doesn't matter	Order matters
3	236	8,260	306	1,440


Control
?