Understanding Property-Based Testing

by Talking to People

Benjamin C. Pierce

WG 2.8 May 2022

Partners in Crime









Joseph W. Cutler 1st Year PhD

Harrison Goldstein 3rd Year PhD

Adam Stein 1st Year PhD Andrew Head Asst. Prof

Property-Based Testing For All

Everyone, loves PBT!

We want the other ~7.753 billion people to love it too.

How do we find out what would help them love it better?



A Preliminary User Study

(in preparation for a bigger one to come...)

Preliminary User Study

Focused on "interviews for need finding."

Recruited 7 industry users of *Hypothesis* (in Python).



Interview Questions

- Tell us about your most memorable time doing PBT. (To get subjects thinking about a specific experience.)
- 2. How did you come up with the properties that you tested?
- 3. Did you need custom generators? If so, what did they generate?

Analyzing Qualitative Data

- 1. Recruit Informants
 - a. We recruited from Twitter (@alpha_convert and @hgoldstein95 🙂)
- 2. Informants Sign Consent Form
- 3. Conduct Interviews
 - a. 30 minute interviews over Zoom
 - b. Calls were recorded, transcribed using Otter.ai
- 4. "Code" Data
 - a. Tag interesting paragraphs of the transcript with labels
 - b. Codes depend on the study goals
 - c. May need to compute "agreement"
- 5. Analyze Coded Data

More detail on coding (if requested)

Code categories:

- "Usage": Specific details about how informants had used PBT
- "Benefits": Reasons the informants liked using Hypothesis
- "Challenges": Issues the informants faced during their use of Hypothesis.
- "Missing features": Features that the informants either explicitly or implicitly asked for
- "Adoption": Grab bag of codes related to the informants adoption and advocacy for Hypothesis in their workplace

Example codes:

- "Round-Trip property" (Usage)
- "Data Ingest Testing" (Usage)
- "Confidence in correctness" (Benefits)
- "Time/Effort" (Challenges)
- "Unclear Mental Model" (Challenges)
- "Cache Failed Tests" (Missing Features)
- "Worries about Maintenance" (Adoption)

Analyzing Qualitative Data

| | A | В | C | D | E | F |
|----|---|---------------|--------------------|----------------------|------------------------|------------------|
| 1 | Quote = | Usage | Benefits = | Challenges = | Missing Features = | Adoption = |
| 2 | | , | Catches My Mist - | - | * | Manager Enf - |
| 3 | | Math/Science | - · · · | - | | - |
| 4 | to their secule - prior secule if even as our the facer of it estimates a ris if the right if | t Data Ingest | · · | ÷ | . | + |
| 5 | to the star of star 1, byth the time shall also and set and path the sharp be sent the | | - · · · | * | * | Manager Enf 👻 |
| 6 | No. Not a new releasing order, a fails, driving price and pricessary was so if it in a it of | | · · | Left Something on 👻 | * | * |
| 7 | to reasonable have said under her des fins, much gamming, sorts program druggle is an alloch- | | · · | Creating an Oracle - | Ŧ | · • |
| 8 | the Property generation and the books has seen taken and seen edged. The dupon white all the | 1 . | · · | Seeing Properties * | • | • |
| 9 | They be been one of such of shake he rept or here of here long. But as or new store of | Math/Science | Example Covera: * | * | | * |
| 10 | the out-of-her and down to be exceeded, to i say that to share, he shares formation, she is | Math/Science | · · | * | * | * |
| 11 | and any sole of strength, and the is not at strength its all agents a control, if has not a | : • | Catches My Mist - | - | - | |
| 12 | No series to been one respective or study on other and an experimental react the open an | ۰ · | Example Covera: * | ÷ | | · · |
| 13 | that i would server be available to developer to no heads, if some, and her as an if a 1 galles as | Use Everywh | · · | * | | • |
| 14 | and requiring encoded requiring a fider with darkney have body of a to put | Async SM Te | · • | * | * | * |
| 15 | has allough syndhese data. I fram, sugger new or if the vestine bing, has of the secting for- | SM Testing | · · | Not Enough Exam 👻 | Better Documentati - | |
| 16 | for the and that a method table and the lightness is its rest with rest shortsgreen in | . , | · · | Requires Mental N - | | • |
| 17 | | Worked from | | + | Async SM Testing 👻 | * |
| 18 | | | Example Covera: * | * | * | * |
| 19 | the own of partnesses a new of sublegibles its regions, and the partness of these by | Simple Comp | · | * | | |
| 20 | work have preferred if you could any 1 h or finger with 1 fings, made have an effective and if you | E . | · · | × | Integration with Exi v | * |
| 21 | for face in second charter, where, for an application is price using the or other or potential of | Check for Ob | | * | - | - |
| 22 | to a rank as for \$10, or \$6, and \$6, analyzing income, showing the page \$1.5, for party | Simple Comp | | | • | * |
| 23 | | | · · | * | - | Insufficient C 👻 |
| 24 | | , | · · | * | Better Documentati - | • |
| 25 | The series of peak do not not sell ingly peak to by other studies on the back range | | · · | Ŧ | - | Science+Mat - |
| 26 | | | Anthropomorphiz * | * | • | |
| 27 | | Math/Science | Т | - | | |
| 28 | compare and a fulfing of the property loady being talk for values styling, showing the di-talk of | Test Runner | · · | × | · · | * |
| 29 | the south has an own applies also and one o an call to rack as the set off, race, | | Confidence in Cc - | * | - | • |
| 30 | had not experie toward weing an earth first sold, and this first path of the related by tags in a result of | ۰ · | Confidence in Cc - | - | - | - |
| 31 | for the presence of the part of the provide the provide sector and the sector sector and the sector | 1 . | · · | Abstraction - | * | • |
| 32 | | | · · | Abstraction * | * | * |
| 33 | | | Catches My Mist - | Abstraction ~ | | - |

What Have We Learned (So Far)?

1. People who use PBT **really** like it!

"I've found probably half a dozen major corner case bugs"

"[PBT] was 1000 times better than the alternative"

"QuickCheck appealed to me as someone who makes a lot of mistakes and wanted the computer to find them for me"

2. There are two surprisingly distinct classes of users...

Power Users

- Fully "bought in"
- Tend to have strong mathematical backgrounds (often PhD in Math/CS)

- Need better generators
- Care about speed and efficiency

Everyday Users

- Use PBT occasionally
- More traditional SE background

- Need help "seeing" properties
- Tend to test simple, "extremal" properties:
 - "Program doesn't crash"
 - "Program behaves exactly like oracle"

These groups can teach us different things!

3. PBT is hard when code is poorly abstracted

- Some reported that "carving out" an interface was much of their testing effort
- Others reported resorting to "end-to-end" properties
- "I can't see any properties to test" was a common refrain

4. We need to do a better job of *teaching* people PBT!

- Several subjects cited lack of examples / experience as a problem
- PBT documentation often uses terminology unfamiliar to engineers
- We should all teach PBT in our courses!

Shriram Krishnamurthi has written a bunch about how to do this!

What's Next?

This is just the beginning!

We are planning a full-scale user study this summer.

Big Questions to Answer:

Where are the points of highest leverage for getting PBT out into the world?

Concretely: What is the best way to move the needle with a PhD dissertation or two over the next 2-3 years?

Discussion...

What would you want to learn from such a study?

Who should we interview? How do we contact them?

How do we find places in industry where PBT can help?

How can we help developers "see the properties"?

How do we help make PBT an ongoing part of a company's culture?

Are today's "property DSLs" sufficiently expressive? Aimed at the right set of users?

Besides interviews, where else can we look for insight into these issues? (e.g. GitHub?)

Watch This Space!











