

The Serpentine Career Change

David Prager Branner

New York Python Meetup Group

2013.07.09

Knewton (New York City)

My background

The quandary

The solution

Training

Tactics

Main points again

Advice for the career-changer

My background

My background

- Career academic

My background

- Career academic
 - Chinese philology and language teaching

My background

- Career academic
 - Chinese philology and language teaching
 - early tenure, extensive publications and public speaking

My background

- Career academic
 - Chinese philology and language teaching
 - early tenure, extensive publications and public speaking
 - managed academic programs for about ten years

My background

- Career academic
 - Chinese philology and language teaching
 - early tenure, extensive publications and public speaking
 - managed academic programs for about ten years
- Decided to work in industry (giving up tenure).

My background

- Career academic
 - Chinese philology and language teaching
 - early tenure, extensive publications and public speaking
 - managed academic programs for about ten years
- Decided to work in industry (giving up tenure).

2008: Took job with machine translation company in New York!

My background

- Career academic
 - Chinese philology and language teaching
 - early tenure, extensive publications and public speaking
 - managed academic programs for about ten years
- Decided to work in industry (giving up tenure).

2008: Took job with machine translation company in New York!

- Using my existing skills (lexicography, academic management) in the service of an business research project.

My background

- Career academic
 - Chinese philology and language teaching
 - early tenure, extensive publications and public speaking
 - managed academic programs for about ten years
- Decided to work in industry (giving up tenure).

2008: Took job with machine translation company in New York!

- Using my existing skills (lexicography, academic management) in the service of an business research project.

2009: Economic crisis! Company closed! Jobs dry up!

The quandary

The quandary

As of 2009:

The quandary

As of 2009:

- To myself, I seemed to have already switched fields from academia to tech.

The quandary

As of 2009:

- To myself, I seemed to have already switched fields from academia to tech.
- To others, I did not appear hireable in tech because of inexperience.

The solution

The solution

- Gradual adaptation to coding world, while continuing to teach and do research.

The solution

- Gradual adaptation to coding world, while continuing to teach and do research.
- Began learning Python and Ubuntu.

The solution

- Gradual adaptation to coding world, while continuing to teach and do research.
- Began learning Python and Ubuntu.
- At first, mostly self-study, supplemented by

The solution

- Gradual adaptation to coding world, while continuing to teach and do research.
- Began learning Python and Ubuntu.
- At first, mostly self-study, supplemented by
 - periodic informal training sessions run by a Python activist;

The solution

- Gradual adaptation to coding world, while continuing to teach and do research.
- Began learning Python and Ubuntu.
- At first, mostly self-study, supplemented by
 - periodic informal training sessions run by a Python activist;
 - help sessions with experienced programmer friend.

The solution

- Gradual adaptation to coding world, while continuing to teach and do research.
- Began learning Python and Ubuntu.
- At first, mostly self-study, supplemented by
 - periodic informal training sessions run by a Python activist;
 - help sessions with experienced programmer friend.
- Constant use of

The solution

- Gradual adaptation to coding world, while continuing to teach and do research.
- Began learning Python and Ubuntu.
- At first, mostly self-study, supplemented by
 - periodic informal training sessions run by a Python activist;
 - help sessions with experienced programmer friend.
- Constant use of
 - David Beazley's *Python Essential Reference*;

The solution

- Gradual adaptation to coding world, while continuing to teach and do research.
- Began learning Python and Ubuntu.
- At first, mostly self-study, supplemented by
 - periodic informal training sessions run by a Python activist;
 - help sessions with experienced programmer friend.
- Constant use of
 - David Beazley's *Python Essential Reference*;
 - Python docs;

The solution

- Gradual adaptation to coding world, while continuing to teach and do research.
- Began learning Python and Ubuntu.
- At first, At first, mostly self-study, supplemented by
 - periodic informal training sessions run by a Python activist;
 - help sessions with experienced programmer friend.
- Constant use of
 - David Beazley's *Python Essential Reference*;
 - Python docs;
 - StackOverflow.

Training

Training

- City College undergraduate Computer Science (minor), taken part-time as second-degree student. (2010-12)

Training

- City College undergraduate Computer Science (minor), taken part-time as second-degree student. (2010-12)
 - cheap for NYC residents

Training

- City College undergraduate Computer Science (minor), taken part-time as second-degree student. (2010-12)
 - cheap for NYC residents
 - very good education — hard-working teachers, serious classmates

Training

- City College undergraduate Computer Science (minor), taken part-time as second-degree student. (2010-12)
 - cheap for NYC residents
 - very good education — hard-working teachers, serious classmates
 - principal benefit: exposure to the important math and theory groundwork of programming

Training

- City College undergraduate Computer Science (minor), taken part-time as second-degree student. (2010-12)
 - cheap for NYC residents
 - very good education — hard-working teachers, serious classmates
 - principal benefit: exposure to the important math and theory groundwork of programming
 - learned LaTeX in order to take math notes quickly

Training

- City College undergraduate Computer Science (minor), taken part-time as second-degree student. (2010-12)
 - cheap for NYC residents
 - very good education — hard-working teachers, serious classmates
 - principal benefit: exposure to the important math and theory groundwork of programming
 - learned LaTeX in order to take math notes quickly
- Hacker School (2013 “Winter” [Spring] batch)

Training

- City College undergraduate Computer Science (minor), taken part-time as second-degree student. (2010-12)
 - cheap for NYC residents
 - very good education — hard-working teachers, serious classmates
 - principal benefit: exposure to the important math and theory groundwork of programming
 - learned LaTeX in order to take math notes quickly
- Hacker School (2013 “Winter” [Spring] batch)
 - like a gym or collaborative art studio for self-driven programmers seeking to improve;

Training

- City College undergraduate Computer Science (minor), taken part-time as second-degree student. (2010-12)
 - cheap for NYC residents
 - very good education — hard-working teachers, serious classmates
 - principal benefit: exposure to the important math and theory groundwork of programming
 - learned LaTeX in order to take math notes quickly
- Hacker School (2013 “Winter” [Spring] batch)
 - like a gym or collaborative art studio for self-driven programmers seeking to improve;
 - “instant network”, like being in a business-school cohort;

Training

- City College undergraduate Computer Science (minor), taken part-time as second-degree student. (2010-12)
 - cheap for NYC residents
 - very good education — hard-working teachers, serious classmates
 - principal benefit: exposure to the important math and theory groundwork of programming
 - learned LaTeX in order to take math notes quickly
- Hacker School (2013 “Winter” [Spring] batch)
 - like a gym or collaborative art studio for self-driven programmers seeking to improve;
 - “instant network”, like being in a business-school cohort;
 - principal benefits:

Training

- City College undergraduate Computer Science (minor), taken part-time as second-degree student. (2010-12)
 - cheap for NYC residents
 - very good education — hard-working teachers, serious classmates
 - principal benefit: exposure to the important math and theory groundwork of programming
 - learned LaTeX in order to take math notes quickly
- Hacker School (2013 “Winter” [Spring] batch)
 - like a gym or collaborative art studio for self-driven programmers seeking to improve;
 - “instant network”, like being in a business-school cohort;
 - principal benefits:
 - acculturation to the habits of speech and thinking of working programmers;

Training

- City College undergraduate Computer Science (minor), taken part-time as second-degree student. (2010-12)
 - cheap for NYC residents
 - very good education — hard-working teachers, serious classmates
 - principal benefit: exposure to the important math and theory groundwork of programming
 - learned LaTeX in order to take math notes quickly
- Hacker School (2013 “Winter” [Spring] batch)
 - like a gym or collaborative art studio for self-driven programmers seeking to improve;
 - “instant network”, like being in a business-school cohort;
 - principal benefits:
 - acculturation to the habits of speech and thinking of working programmers;
 - exposure to criticism and best-practices of the field;

Training

- City College undergraduate Computer Science (minor), taken part-time as second-degree student. (2010-12)
 - cheap for NYC residents
 - very good education — hard-working teachers, serious classmates
 - principal benefit: exposure to the important math and theory groundwork of programming
 - learned LaTeX in order to take math notes quickly
- Hacker School (2013 “Winter” [Spring] batch)
 - like a gym or collaborative art studio for self-driven programmers seeking to improve;
 - “instant network”, like being in a business-school cohort;
 - principal benefits:
 - acculturation to the habits of speech and thinking of working programmers;
 - exposure to criticism and best-practices of the field;
 - pair-programming and code-review.

Tactics

Tactics

- Gradualism

Tactics

- Gradualism
 - continued to support self by teaching and consulting;

Tactics

- Gradualism
 - continued to support self by teaching and consulting;
 - also published two books (2011, 2012), and am near completing three others;

Tactics

- Gradualism
 - continued to support self by teaching and consulting;
 - also published two books (2011, 2012), and am near completing three others;
 - remained actively engaged with network in previous career

Tactics

- Gradualism
 - continued to support self by teaching and consulting;
 - also published two books (2011, 2012), and am near completing three others;
 - remained actively engaged with network in previous career
- Concentrated on writing small projects that would help me with research projects or teaching, rather than working through tutorials of little interest.

Tactics

- Gradualism
 - continued to support self by teaching and consulting;
 - also published two books (2011, 2012), and am near completing three others;
 - remained actively engaged with network in previous career
- Concentrated on writing small projects that would help me with research projects or teaching, rather than working through tutorials of little interest.
- Self-study first, formal study afterwards.

Tactics

- Gradualism
 - continued to support self by teaching and consulting;
 - also published two books (2011, 2012), and am near completing three others;
 - remained actively engaged with network in previous career
- Concentrated on writing small projects that would help me with research projects or teaching, rather than working through tutorials of little interest.
- Self-study first, formal study afterwards.
- Keep personal notebook (electronic, not paper) of everything I research, every problem I solve (or try to).

Tactics

- Gradualism
 - continued to support self by teaching and consulting;
 - also published two books (2011, 2012), and am near completing three others;
 - remained actively engaged with network in previous career
- Concentrated on writing small projects that would help me with research projects or teaching, rather than working through tutorials of little interest.
- Self-study first, formal study afterwards.
- Keep personal notebook (electronic, not paper) of everything I research, every problem I solve (or try to).
- Set myself to learn one language deeply rather than succumbing to the temptation to study many.

Advice for the career-changer

Advice for the career-changer

1. Question: Immediate or gradual?

Advice for the career-changer

1. Question: Immediate or gradual?

– If immediate, 6-12 months is almost certainly enough time.

Advice for the career-changer

1. Question: Immediate or gradual?

– If immediate, 6-12 months is almost certainly enough time.

2. Question: Back-end or front-end?

Advice for the career-changer

1. Question: Immediate or gradual?

– If immediate, 6-12 months is almost certainly enough time.

2. Question: Back-end or front-end?

– Front-end probably faster but it seems a question of temperament.

Advice for the career-changer

1. Question: Immediate or gradual?
 - If immediate, 6-12 months is almost certainly enough time.
2. Question: Back-end or front-end?
 - Front-end probably faster but it seems a question of temperament.
3. Cultivate doggedness and attempt hard projects.

Advice for the career-changer

1. Question: Immediate or gradual?
 - If immediate, 6-12 months is almost certainly enough time.
2. Question: Back-end or front-end?
 - Front-end probably faster but it seems a question of temperament.
3. Cultivate doggedness and attempt hard projects.
4. Put shame aside and ask for help as much as possible. Mistakes are of no consequence unless you fail to learn from them.

Advice for the career-changer

1. Question: Immediate or gradual?
 - If immediate, 6-12 months is almost certainly enough time.
2. Question: Back-end or front-end?
 - Front-end probably faster but it seems a question of temperament.
3. Cultivate doggedness and attempt hard projects.
4. Put shame aside and ask for help as much as possible. Mistakes are of no consequence unless you fail to learn from them.
5. Ignore the nagging question “am I hireable” — you will be.

Main points again

Main points again

1. Decide whether gradual/immediate and front/back-end.

Main points again

1. Decide whether gradual/immediate and front/back-end.
2. Self-study first; try to work toward the point where you can get some training.

Main points again

1. Decide whether gradual/immediate and front/back-end.
2. Self-study first; try to work toward the point where you can get some training.
3. Cultivate doggedness and attempt hard projects; avoid easy-entry classes.

Main points again

1. Decide whether gradual/immediate and front/back-end.
2. Self-study first; try to work toward the point where you can get some training.
3. Cultivate doggedness and attempt hard projects; avoid easy-entry classes.
4. Ask for help wherever you can and don't be ashamed of being stupid or making mistakes.

If I can help you by listening, please feel free to contact me.

If I can help you by listening, please feel free to contact me.

Go to **brannerchinese.com** and look for my email in the “email hider” at the bottom of the page.

If I can help you by listening, please feel free to contact me.

Go to **brannerchinese.com** and look for my email in the “email hider” at the bottom of the page.

END