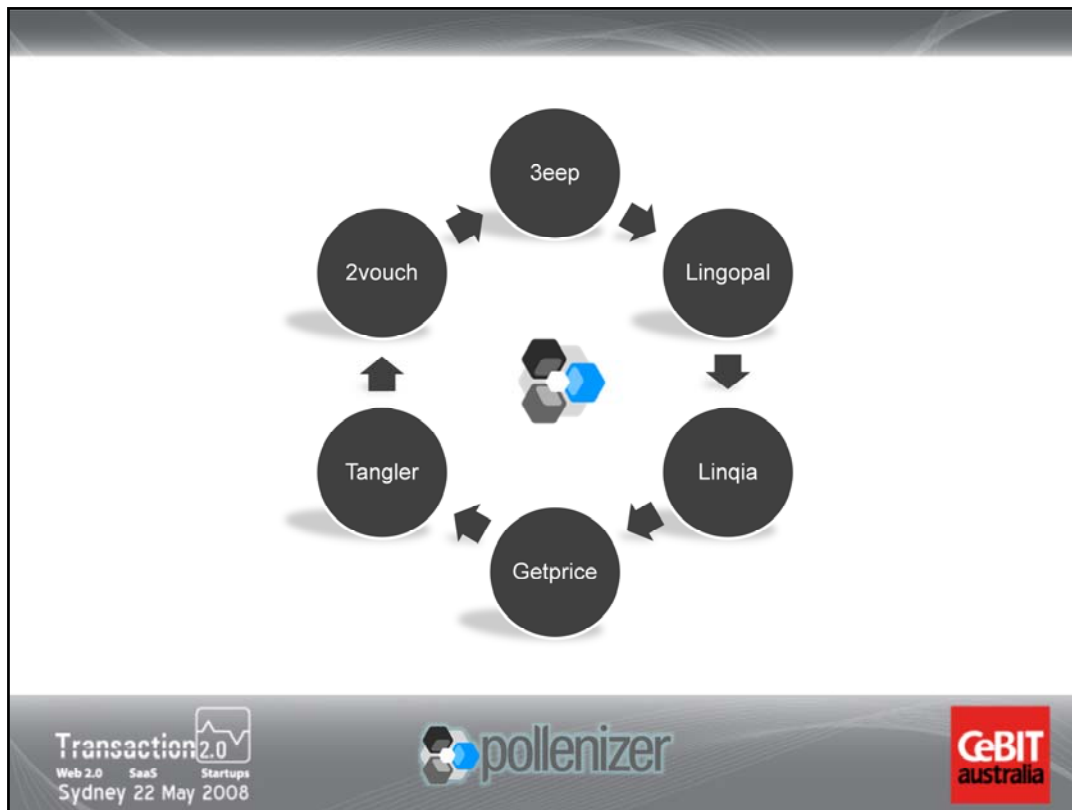




- Hi
- I am Phil Morle.
- Co-Founder of Start-up Management Firm - Pollenizer**
- Hell's kitchen is what it feels like (sometimes) to make a dream into a reality. You've made something cool ... Now you are a few months or years into executing on a technology business.
- Things get tough in the middle
 - Releases take too long
 - Can't depend on the schedule – everyone disappointed – tech team feel like failures
 - Cost 4 x what it did when you started
 - Conflicts between business development and engineering
 - Its hard getting users
 - The application won't scale
- First Hell's Kitchen: I was Kazaa CTO for 4 years – 5 million users at the same time, very complex customer relationships, international teams in Estonia, Sweden, Israel and Sydney. Business development in Los Angeles. We needed to design systems to cope. When I was at Kazaa, I thought that we were the only company in the world that struggled...
- But I now know it is common.**



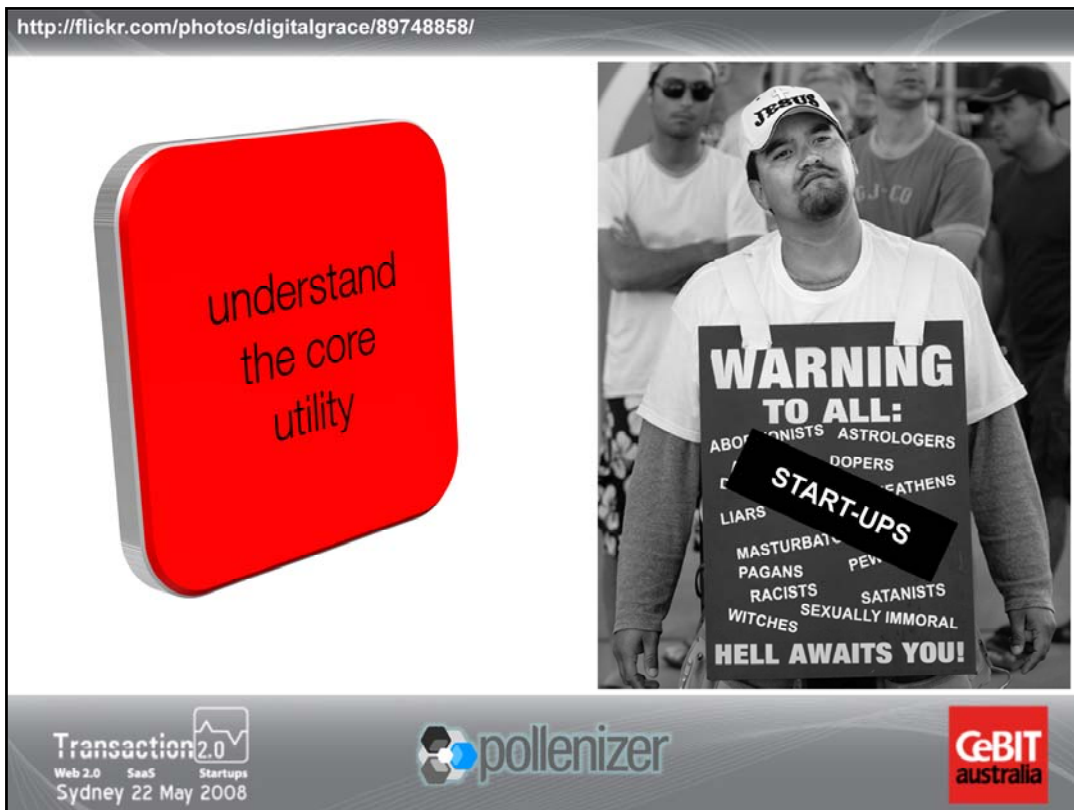
Today I am one of the founders of Pollenizer which is a start-up management firm.

My job is a privilege because of the opportunities I get to learn across organisations. I can see common problems.

- A way for users to make real money through placing people in jobs – about to launch
- Social network for grass-roots sport
- 2-way, user generated translation tool for mobile devices
- Community search engine
- Australia's biggest comparison shopping service
- Live discussions that can be distributed all over the web.

• I see Hell's Kitchen everyday and am inspired by the ways these companies learn to work through the problems that mid-stage start-ups attract.

• **WHAT FOLLOWS ARE NOT 'RULES' BUT THEY WORK FOR ME...**



- **Understand the core utility.**
- **In Hells Kitchen the team doesn't know what they are building and why.**
- **Not knowing this:**
 - Disappoints users
 - Wastes resources
 - Extends timetables
- **Common to think you know this... But often people don't...**
- **PROCESS:**
 - Relationship between different 'users' - nodes
 - What each node needed and how they related
 - **Drawing on the meeting room wall**
 - Apply decisions to... How does this add to the core utility?
 - Living document
 - Hard to say 'Can we add a widget?'
 - Easy to ask 'Is our search algorithm right?'
- De-engineering



- **A successful business does not equal more features.**
- **In Hell's Kitchen there is a belief that the next feature will make the product good.**
- But it doesn't.
- Great thing about modern tech is that seems cheap – is cheap...
 - E.g. Plug-ins, Ajax Frameworks, APIs for instant features.
- Let's add video sharing, social networking, Google maps!
- **"When we have feature X..."**
- **"Just one more feature!"**
- Pollenizer has a whole programme called 'de-engineering' because this is endemic.

RATHER

- **Magic**– a thousand tiny units of magic is harder to justify, can take longer, is less visible but makes magic software.
- Throw it against the 'core utility'
- E.g. 3eep
 - Google Maps? Or Printable sheet for Mums to get to the match?



- Code informs strategy informs code
- In Hells Kitchen cool technology dominates the business or business development designs the software
- Story: Gordon Ramsey – chefs and service in the room – working as one – finding solutions together
- Make the tech part of the solution when an idea is initiated... But don't be focussed on tech.
 - Lingopal card exchange idea? Engineers found actual ways to bring this to life?
 - Compression technology: what can business development bring? The first real use?
- Talk about outcomes not about technology. Coach Phil getting where he needs to go not 'Google Maps'
- Find solutions together.



- Caution (A camel is a horse designed by committee)
- Know when to shut-up. Especially CEOs.
- Find a Product Manager to obsess about the whole experience and trust them.



- You need Talent AND Process
- In Hells Kitchen, talent without process hides problems... For now.
- You need talent. People that understand the product and obsess about it. Fight for it. Know it more than you but...
- Talent without process... **Talent can absorb no process for quite a while..** Then it breaks badly.
- Companies that have a talented engineer as a founder can suffer from this.
- The Hopeful 'YES'
- The Myth of 2 days, 2 weeks or 2 months.
- Find a process that exposes pockets of weakness ...
- **AGILE:** All of our clients use this process because it celebrates talent, is not over bureaucratic, but overlays a structure.
- Briefly...
- Sprints – 2 weeks – can be estimated.
 - The science of estimation. Small – predictable sprints.
 - Minor course corrections.
 - The third iteration will be better.
- Huddles – daily 10 minute calls (yesterday, today and getting in your way)
- Feedback loops – why were we late/early?
- Metrics – Look at the stats – what did we change?
- Living Roadmap – throw decisions against it.
- Morle's Law – Can you apply it?



- **Cheap can be expensive**
- **In Hell's Kitchen, cheapness is seductive.**
- People and Tools
- People
 - Treat offshore teams equally and intimately – no black box. Sprints with internal deliverables.
- Tools
 - Free tools (that no one uses) – Google docs, Media Wiki... Spend some money on a good task system and IP system.
 - Worse – allow email to naturally store your company's IP and TODO list.
 - **Pollenizer bought confluence and Jira on day one.** How we replace email



- **Don't do you own QA.**
- **In Hell's Kitchen, the founders still do their own QA.**
- You are bad at QA – because you know what it should do
- You don't have the time to QA – so you rush it.
- **Hire an offshore team for \$14 / hour who are amazing at it.**
- Also:
 - **DEVELOPERS:** Unit testing (even though it feels like a waste)
 - **QA:** Machines are cheaper than people so automate

<http://flickr.com/photos/nerdtech/48823238/>



Transaction 2.0
Web 2.0 SaaS Startups
Sydney 22 May 2008

 pollenizer

 CeBIT
australia

- Know what to leave till later
- In Hell's Kitchen, the engineers are readying the platform for 100,000 users before their are any users.
- Its harder to get users than it is to scale. So many initial meetings are about how to scale not core utility.
- But do have a plan...



- Use your product
- In Hell's Kitchen, founders don't use their own product.
- The simplest rule of all. No matter how busy you are.
- Stop worrying about things on the surface (the logo colour) and start worrying about the utility. **Core utility again.**
- Try this for an exercise: The next social site that goes down – go look for the profile of the CEO.
- Friendster as an example.



You can escape Hell's Kitchen, but its hard.

Escape Velocity.

Ideas are simple, execution is hard.

Discipline.

- Understand the core utility
- Engineering and business – work together as one
- Measure what happens – learn.
- Have a shared, known process.
- Automate and outsource testing
- Use your product
- Use your product

Good luck.

<http://flickr.com/photos/photochiel/204790709/>

phil@pollenizer.com
www.pollenizer.com

blog: philmorle.com
twitter: [philmorle](https://twitter.com/philmorle)

Transaction 2.0
Web 2.0 SaaS Startups
Sydney 22 May 2008

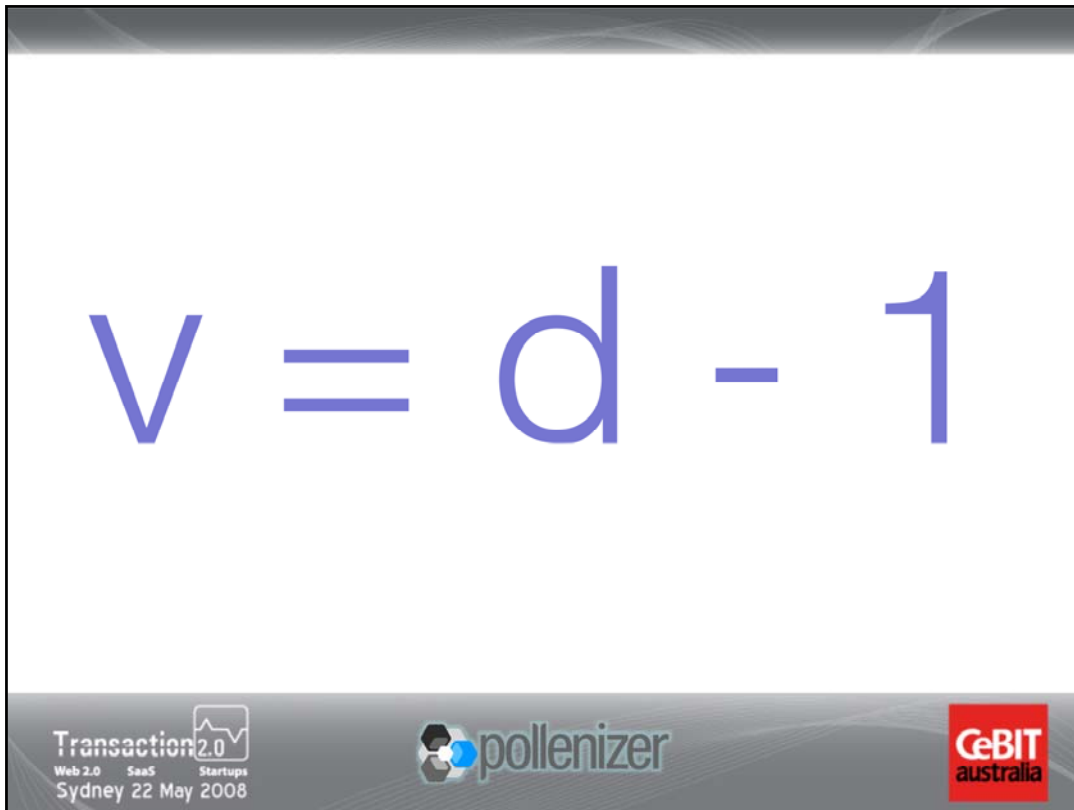


Thanks



Transaction 
Web 2.0 SaaS Startups
Sydney 22 May 2008





I'm not going to talk about rules or laws today. Just some things that have worked for me and the companies I work with.

Only one law in my life – 'Morle's Law'

V = Phil's Vacation Start Date

D = Release date

I can do that because I put systems in place...

Anyway, here's what I have learned...



- Twitter is a good example of what to do right.
 - Ev Williams on Simplicity: <http://light.vpod.tv/?s=0.0.392057>
 - Learned from Odeo – a podcasting client that tried to be too much
- Google – organising the world’s information
 - Notice how search is becoming more sophisticated:
 - E.g. GPS on my phone search results
- LinkedIn – didn’t try to be Facebook
- Facebook – Did have it right but
 - May lose? Lost focus in apps?
 - What if Facebook stayed on core...? Connect people around you...?
 - No apps?
 - Data Portability – Friend connect?