**The Three Ways**

In The Phoenix Project, Bill discovers The Three Ways - a trio of principles used to improve the relationship between Dev and Ops, and to improve the processes used to control the throughput of work throughout the organization. The First Way describes Systems Thinking: Bill grows to realize that in order to meet the business’ objectives and produce valuable change faster, local optimization of the operations group or development group may come at the expense of optimization of the whole system. The best way to have positive impact is to take a whole-system view.

This in my mind ties back into avoiding optimizing for local maxima as described by Deming, Anderson, the Poppendiecks and others. Competition between departments kills co-operation and it is absurd to accept a situation where one group can ‘win’ when the other group ‘fails’ - the entire business must either win or fail and responsibility for considering how your process affects others must expand upwards to take this into account.

The Second Way as used by Bill is to implement feedback loops. Bill fairly rapidly starts having retrospective meetings after deployment issues, sets up deployment and incident drills to learn from practiced responses and solicits feedback from both the developers and operations staff on the ground as well as the managers of the business departments his work impacts upon. This is a core agile principle and the [fifth core practice in Kanban](http://willhamill.com/2013/11/20/kanban-beyond-the-board/) and also corresponds to the Plan-Do- Check-Adjust cycle described by Deming/Shewart for continuous improvement.

The Third Way is creating a culture that promotes both mastery of the current process through repetition (for example, repeatedly doing and documenting deployments until they become less unknown and risky) and continual experimentation (taking small risks in an attempt to learn new things to improve the process). In Bill’s case, pushing the IT organization from quarterly releases to multiple releases per day focuses on the repetition and also ties in to the concepts mentioned by Jez Humble & David Farley in their book Continuous Delivery when they say that the best way to become good at something that is risky and painful is to embrace the pain and reduce the risk by doing it more often and forcing yourself to learn how to do that more effectively - through automation.

**Summary**

Overall I found that The Phoenix Project was an enjoyable and interesting read and I’m glad that it coherently brings together the many associated facets of modern management and IT practices. In my head it seems more clearly that fundamentally all of the things I’ve read on agile development, lean, Kanban, Theory of Constraints, Deming’s System of Profound Knowledge and even [motivation](http://www.amazon.co.uk/Drive-Surprising-Truth-About-Motivates/dp/184767769X/) are all facets of the same concept of making people work more effectively. A good start to my 2014 to-read list!

**Materials Referenced in The Phoenix Project**

[The Goal:](http://www.amazon.co.uk/The-Goal-Process-Ongoing-Improvement-ebook/dp/B002LHRM2O/) A Process of Ongoing Improvement by Eliyahu Goldratt  
[The Five Dysfunctions of a Team](http://www.amazon.co.uk/The-Five-Dysfunctions-Team-Leadership/dp/0787960756/) by Patrick Lencioni  
[Continuous Delivery](http://www.amazon.co.uk/Continuous-Delivery-Deployment-Automation-Addison-Wesley/dp/0321601912/) by Jes Humble and David Farley  
[Kanban](http://www.amazon.co.uk/Kanban-David-J-Anderson/dp/0984521402/) by David J Anderson  
[10 Deploys Per Day at Flickr](http://www.slideshare.net/jallspaw/10-deploys-per-day-dev-and-ops-cooperation-at-flickr) by John Allspaw  
[Toyota Kata:](http://www.amazon.com/Toyota-Kata-Managing-Improvement-Adaptiveness/dp/0071635238) Managing People for Improvement, Adaptiveness and Superior Results by Mike Rother