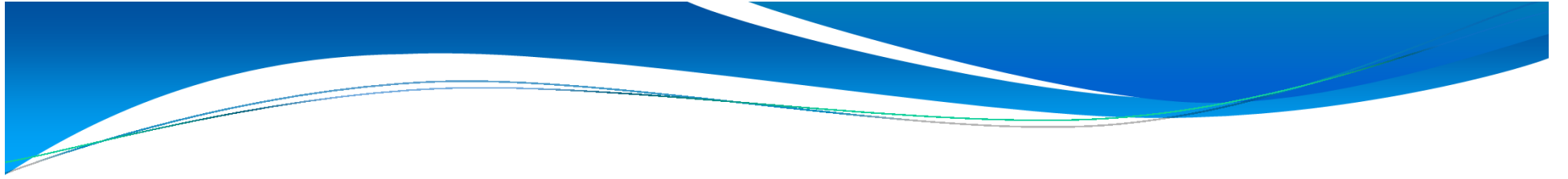




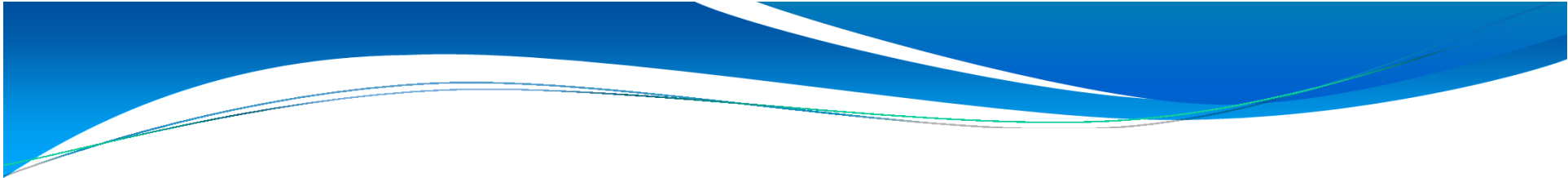
Agile Development

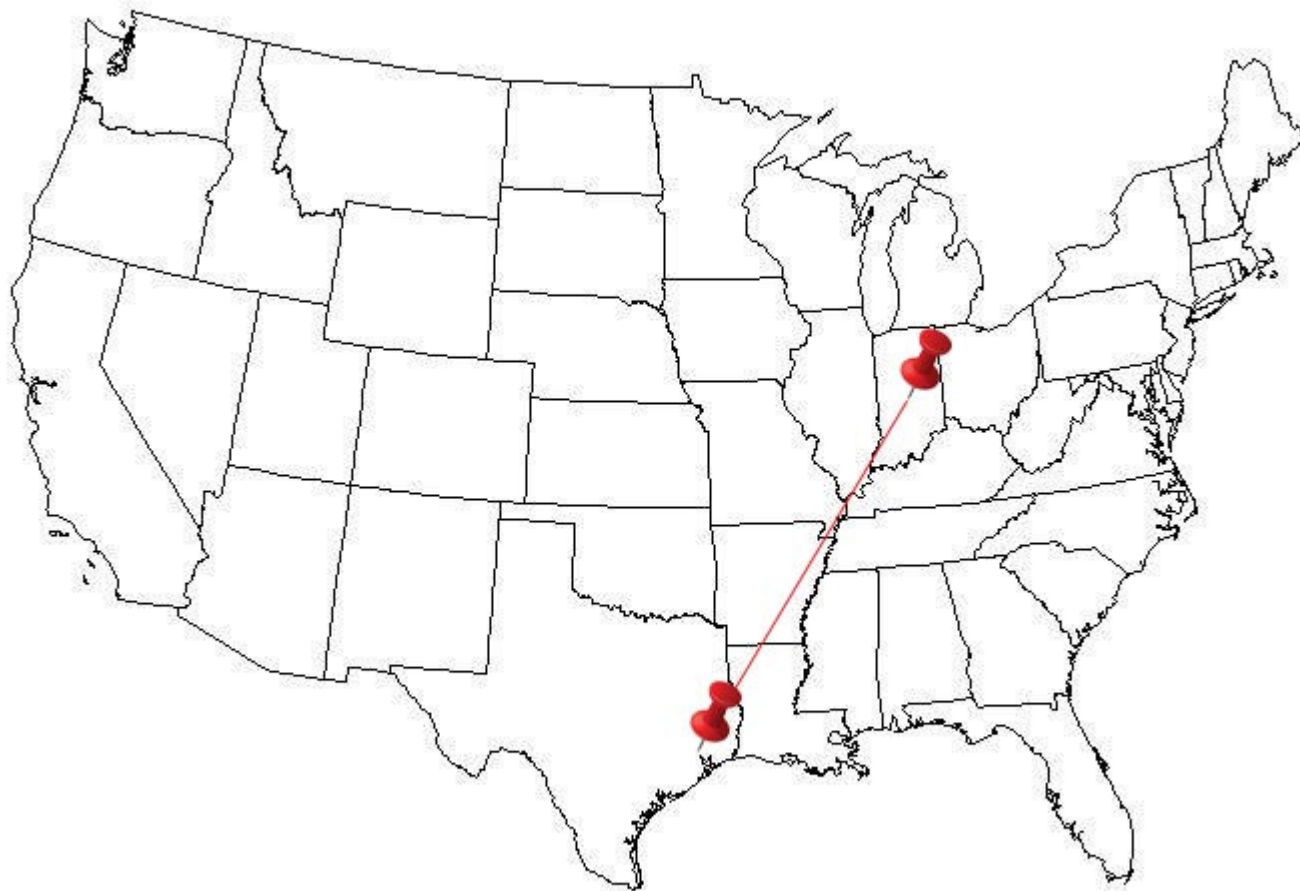
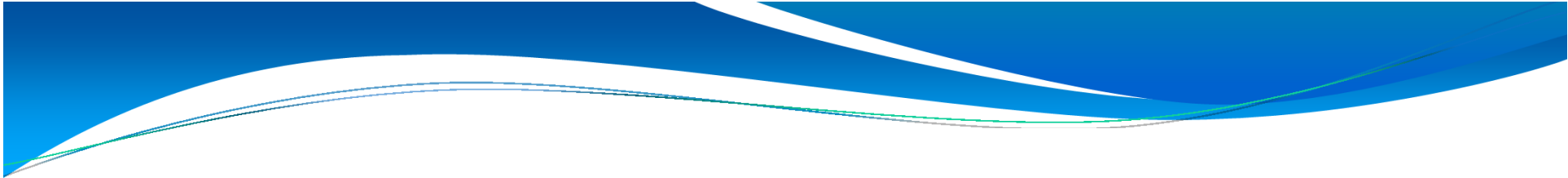
A Journey & How it Works for Clients

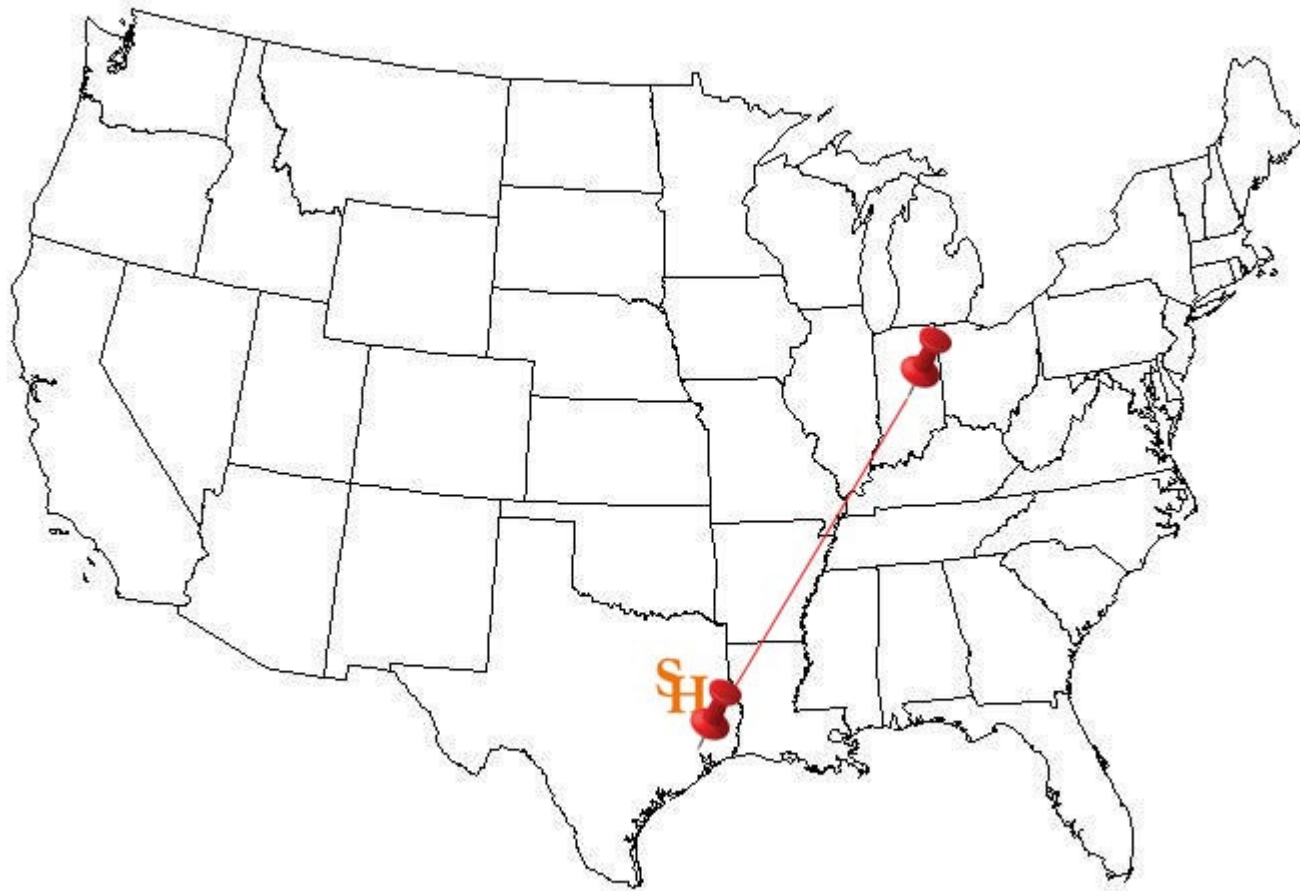
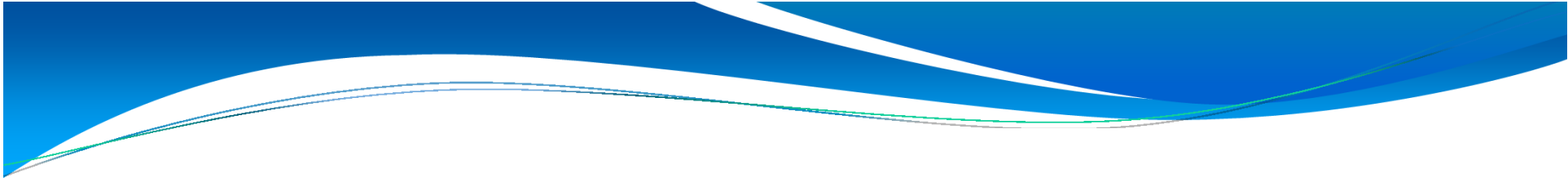


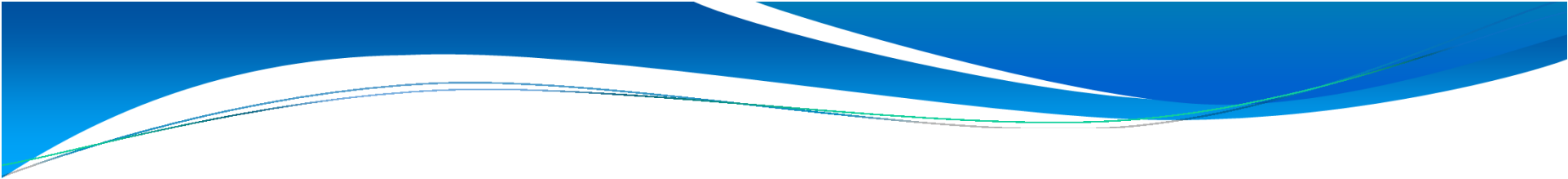
Background

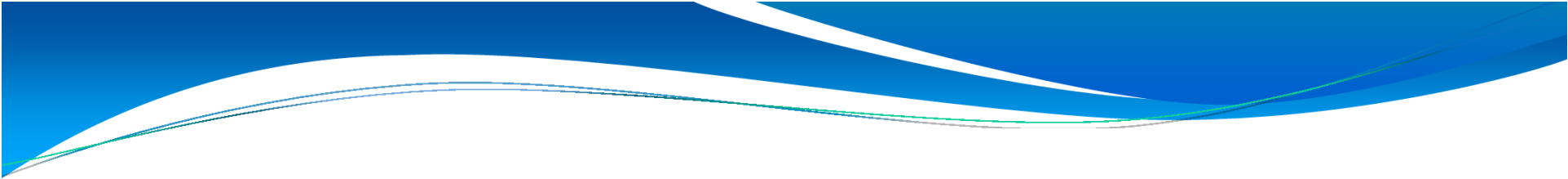
My “QUICK” Journey

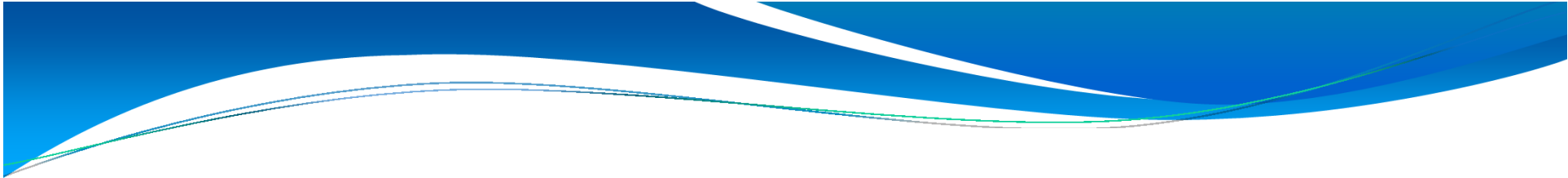


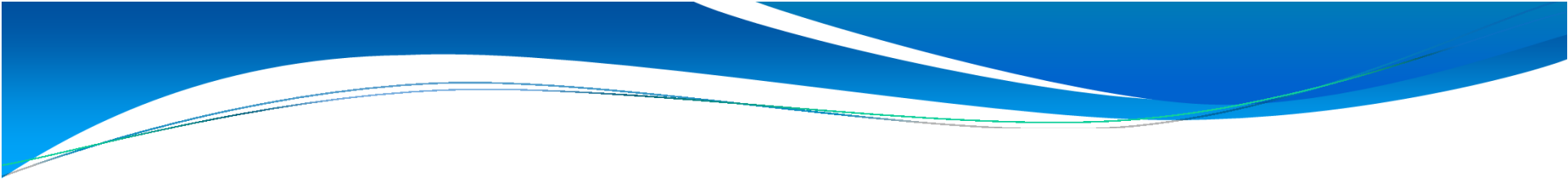


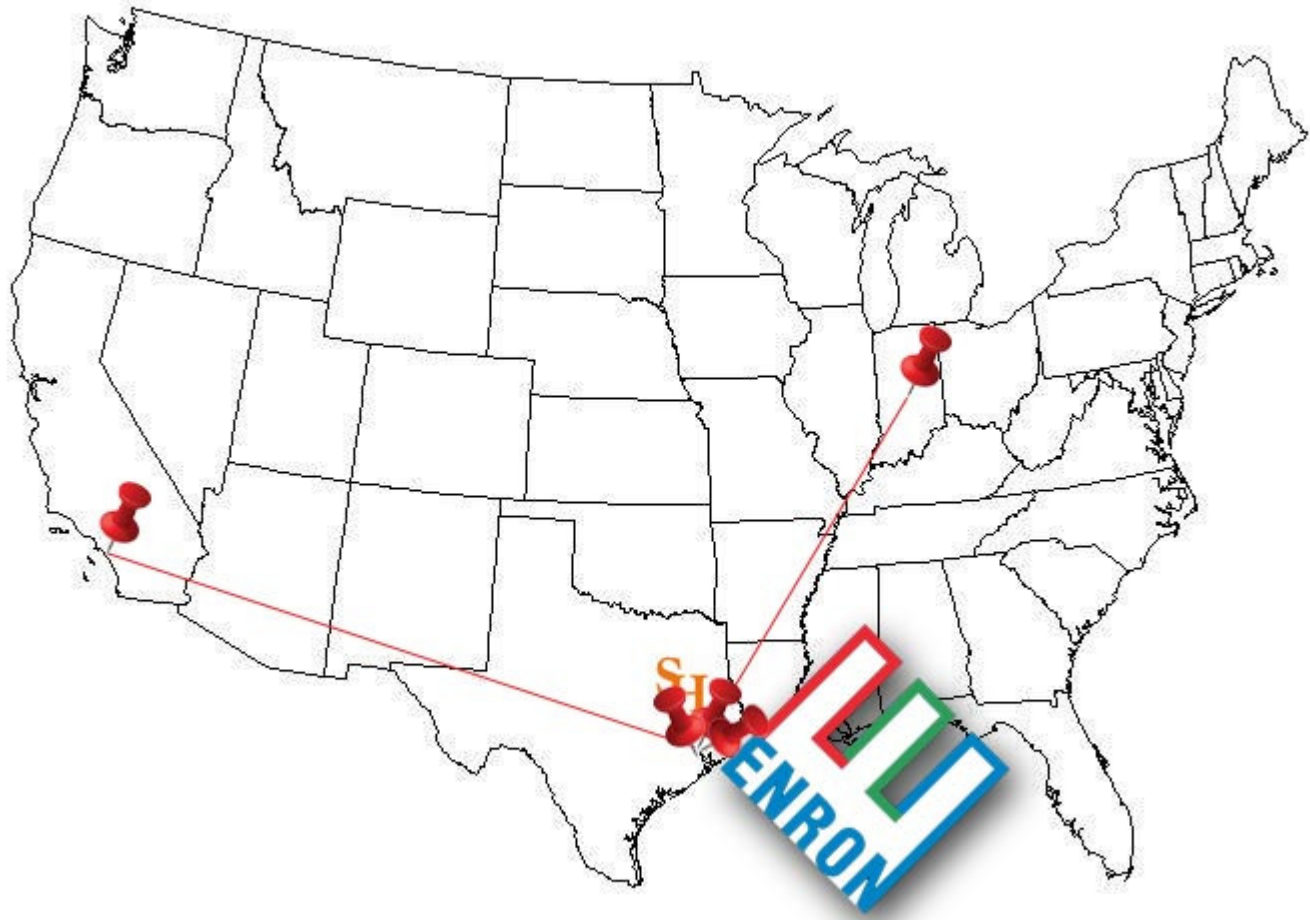
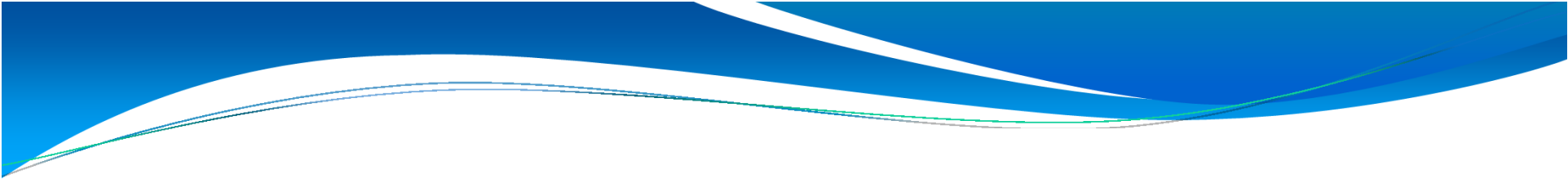


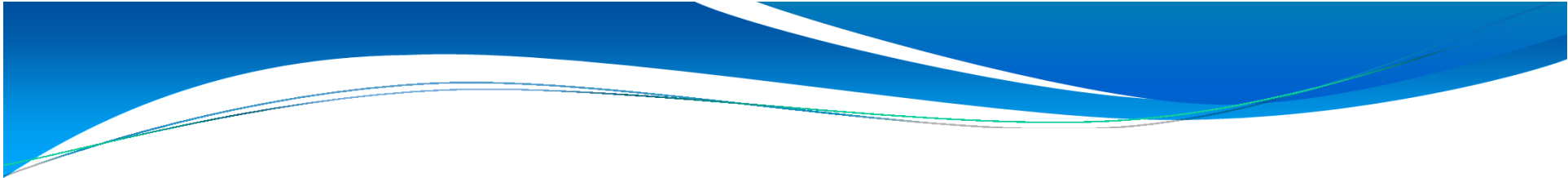


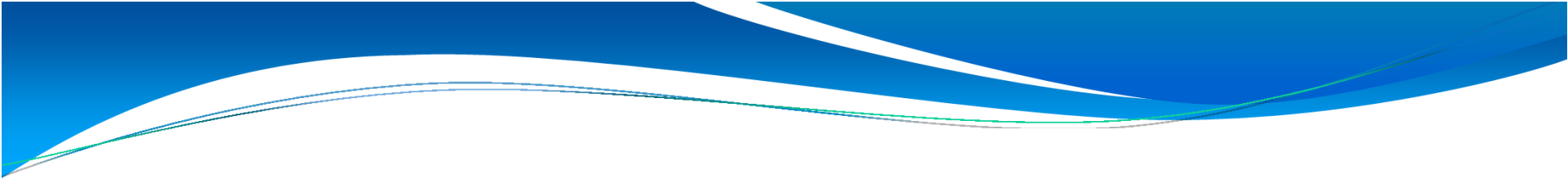


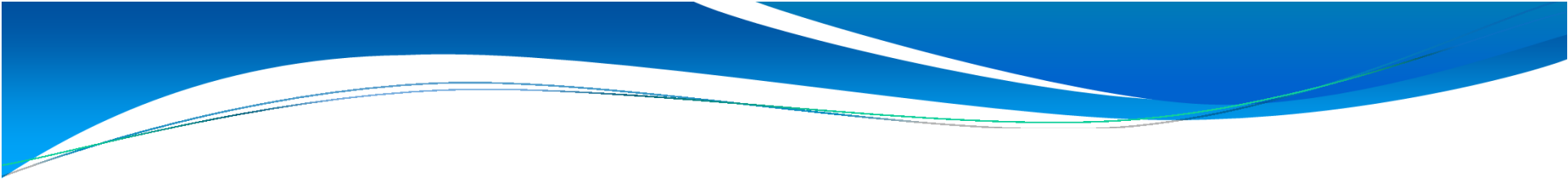


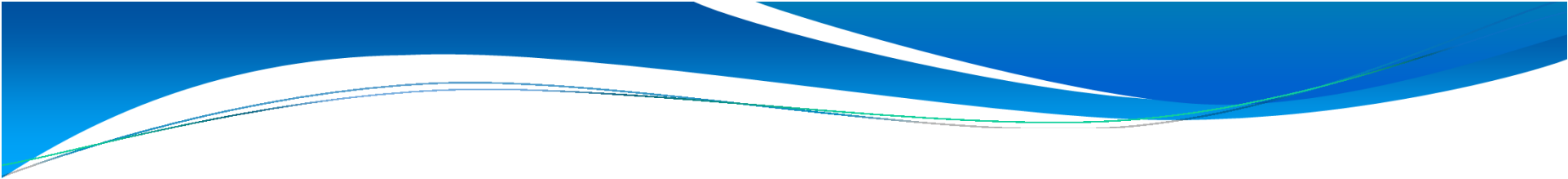


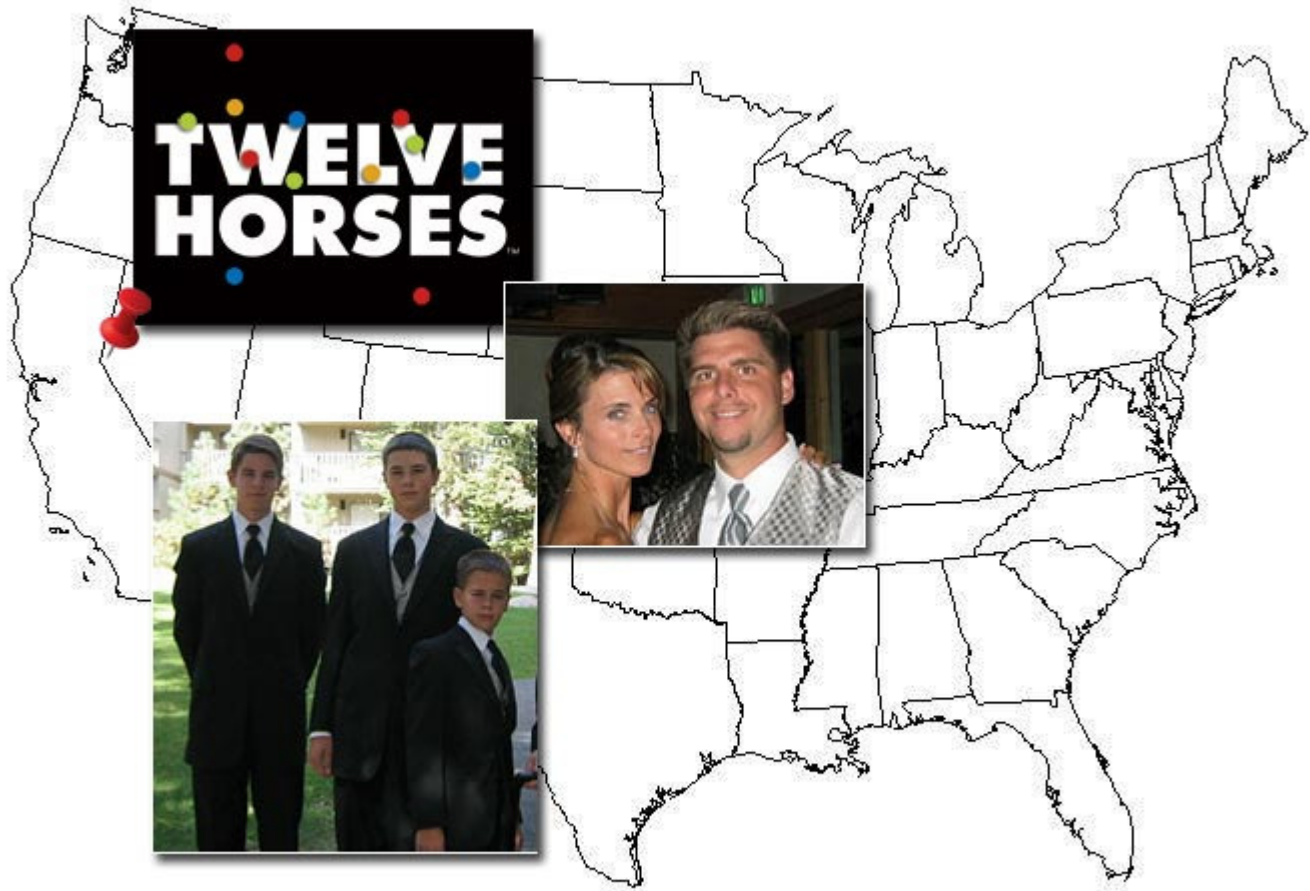
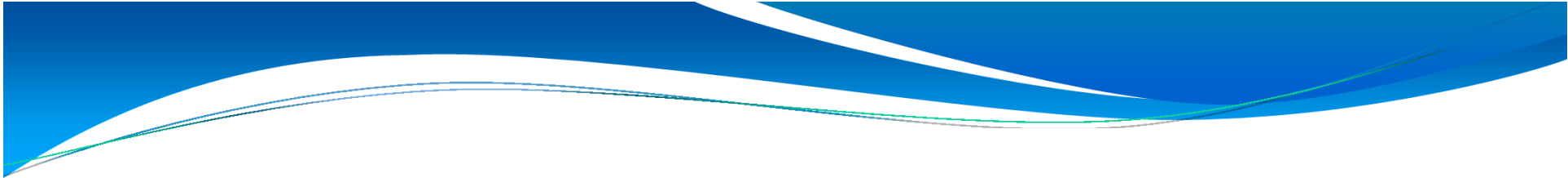






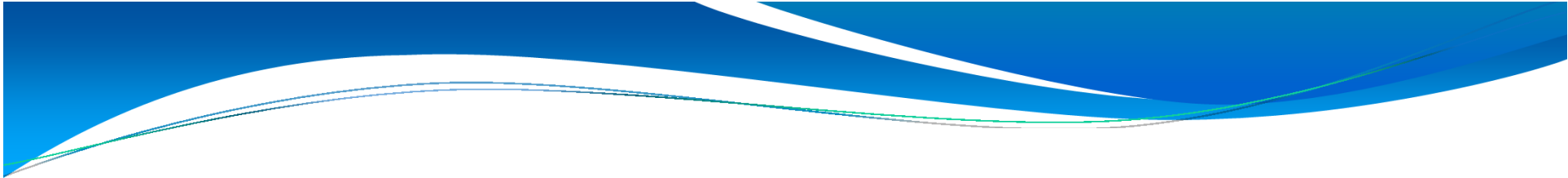






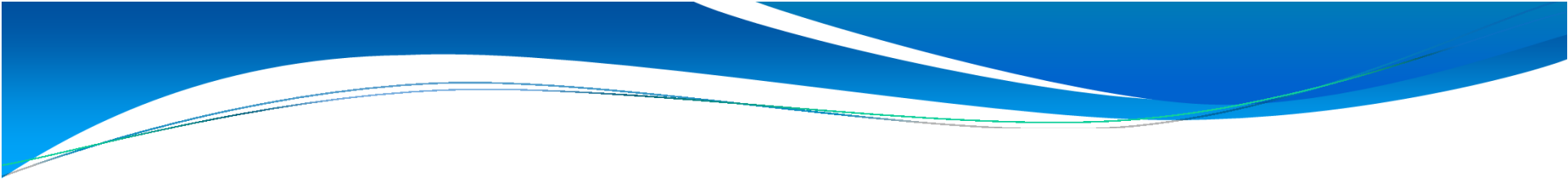
**TWELVE
HORSES**

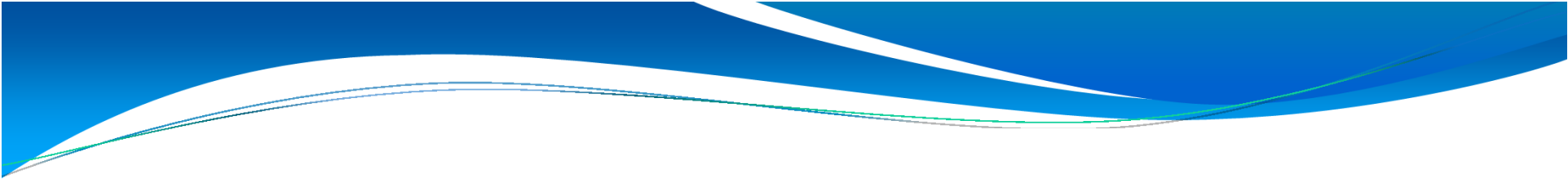




**TWELVE
HORSES**







**TWELVE
HORSES**

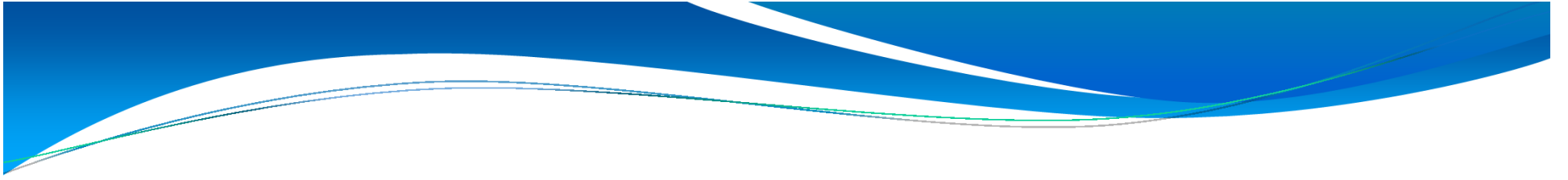

One to One connect

 **MessageMaker**
Social

We had issues with...

- Responding to a rapidly shifting market?
- How to get functionality to sales & clients faster
- How to better deliver what clients want
- How to maximize our limited resources
- ROI was too far out





Agile

Background



Manifesto

- Individuals & Interactions over process & tools
- Working software over comprehensive documentation
- Customer collaboration over contract negotiation
- Responding to change over following a plan

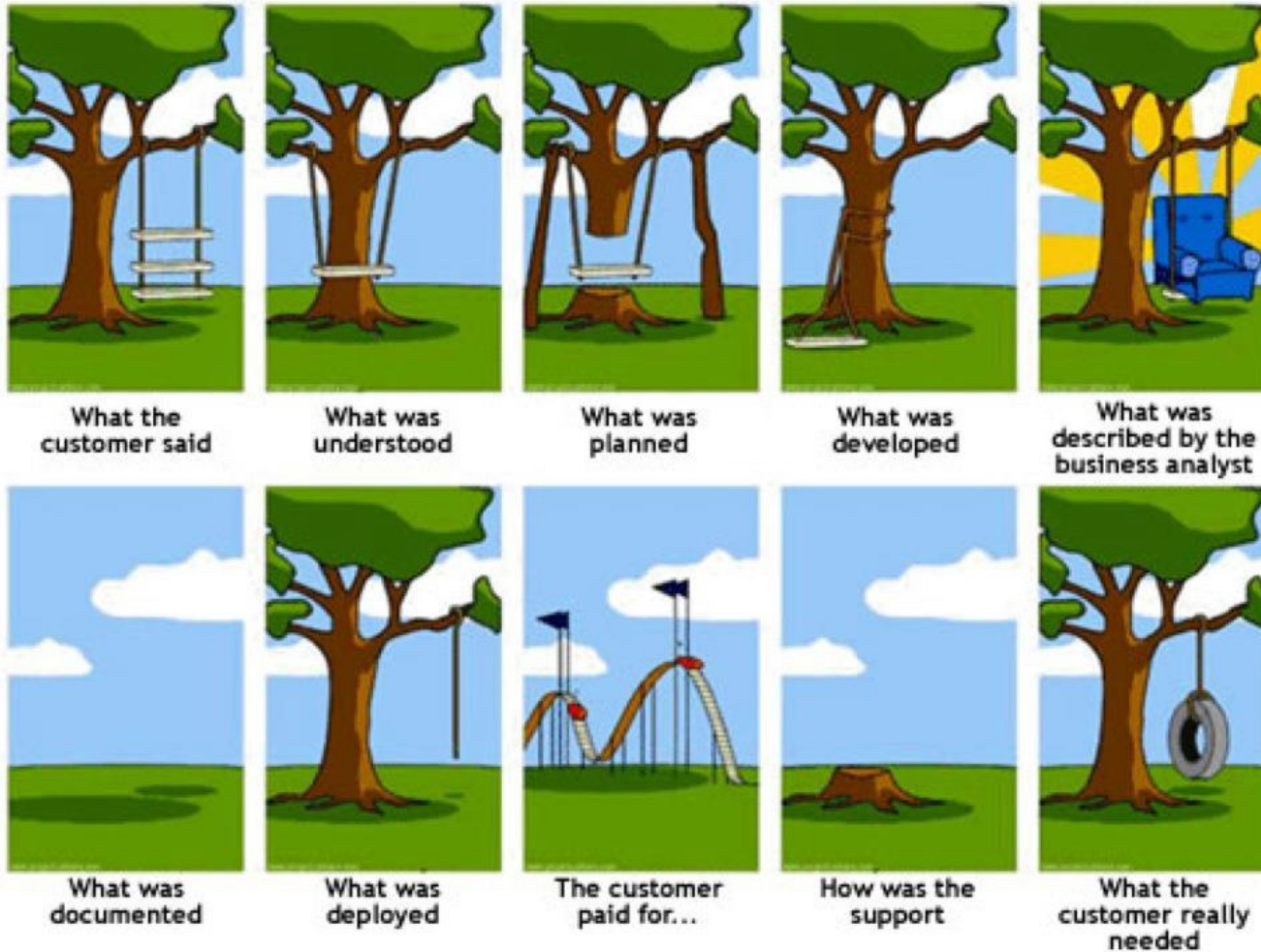


Twelve Principles

- Customer satisfaction by rapid delivery of useful software
- Welcome changing requirements, even late in development
- Working software is delivered frequently
- Working software is principle measure of progress
- Sustainable development, able to maintain a constant pace
- Close, daily co-operation between business people & developers
- Face-to-face conversations is the best form of communication
- Projects are built around motivated individuals, who should be trusted
- Continuous attention to technical excellence and good design
- Simplicity
- Self organizing teams
- Regular adaptation to changing circumstances

Communication is Key

www.blogcmmi.com



Several Methodologies

- Agile Modeling
- Agile Unified Process
- Dynamic Systems Development Method
- Essential Unified Process
- Exia Process
- **Extreme Programming**
- **Feature Driven Development**
- Open Unified Process
- **Scrum**
- Crystal Clear
- Velocity Tracking
- Kanban
- GSD



Roles

- **Product Owner** = Voice of Customer
- **Development Team** = Designers, Developers, Testers, Documentation, etc.
- **Scrum Master** = Rules & Removes Impediments
- **Stakeholders** = Customers & Vendors
- **Managers** = Environment





Other Terms

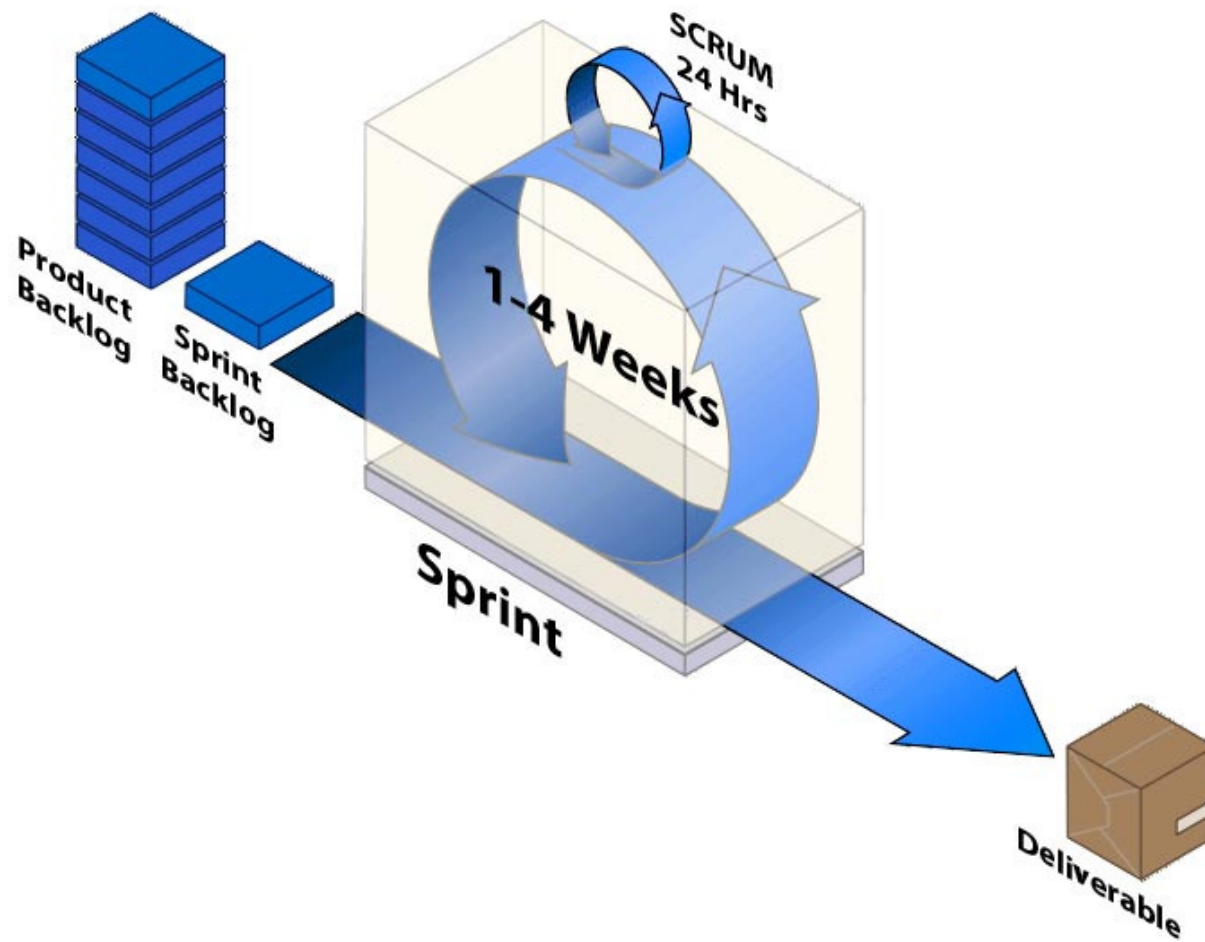
- Sprint
- User Story
 - Written Description
 - Acceptance Criteria
- Story Points
- Backlog
 - Product
 - Sprint
- Tasks
- Planning Poker
- Acceptance Tests
- Epic
- Theme
- Velocity
- Cards
- Burndown
- SPIKE
- Retrospective
- Timebox
- Abnormal Termination



The Activities

- Daily Scrum
 - Backlog Grooming
 - Sprint Planning
 - Sprint Review
 - Sprint Retrospective
-
- Scrum of Scrums – Multiple Teams

What it looks like



Features

- Title & Author
- Purpose & Scope
- Stakeholders
- Market Assessment & Target
- Product Overview
- Use Cases
- Requirements
 - Functional Requirements
 - Usability Requirements
 - Technical Requirements
 - Environmental Requirements
 - Support Requirements
 - Constraints
 - High Level workflow
- Constraints
- Workflow Plans
- Timelines & Milestones
- Eval Plan & Metrics



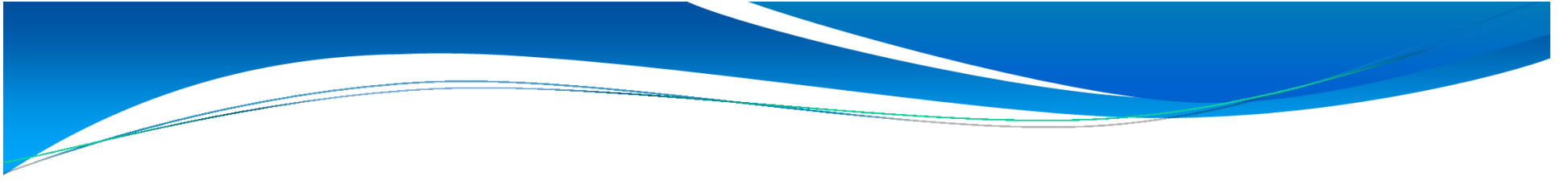


Stories

As a user who consumes reports I would like to be able to view the number of followers my LinkedIn Company has at any given time.

- How often will we be pulling this?
- Need to handle rate limiting

- Login and run report for today & 30 days in past
- Cross reference report with followers on LinkedIn
- Delete LinkedIn Company & ensure report maintains historic values



Agile

Pain & Pleasure



Implementation Issues

- Shift in working style
- Some of the team had issues converting their thinking from well documented Features to semi-vague stories
- Every team has a personality and as such a specific preference on how a story should be written
- No one had a basis for story point estimations
- Story pointing debates took FOREVER
- Stories need pre-grooming & sessions timeboxed



Implementation Issues

- How to model work for components that are in more than one story
- Integration of Documentation & Help
- Inclusion of maintenance activities as stories
- How does Support interact with Dev vs Product Owner
- Deciding what stories get added/axed in 4th quarter
- Devs feeling involved in product decisions
- Use of single vs buddy system - Ownership



Ongoing Issues

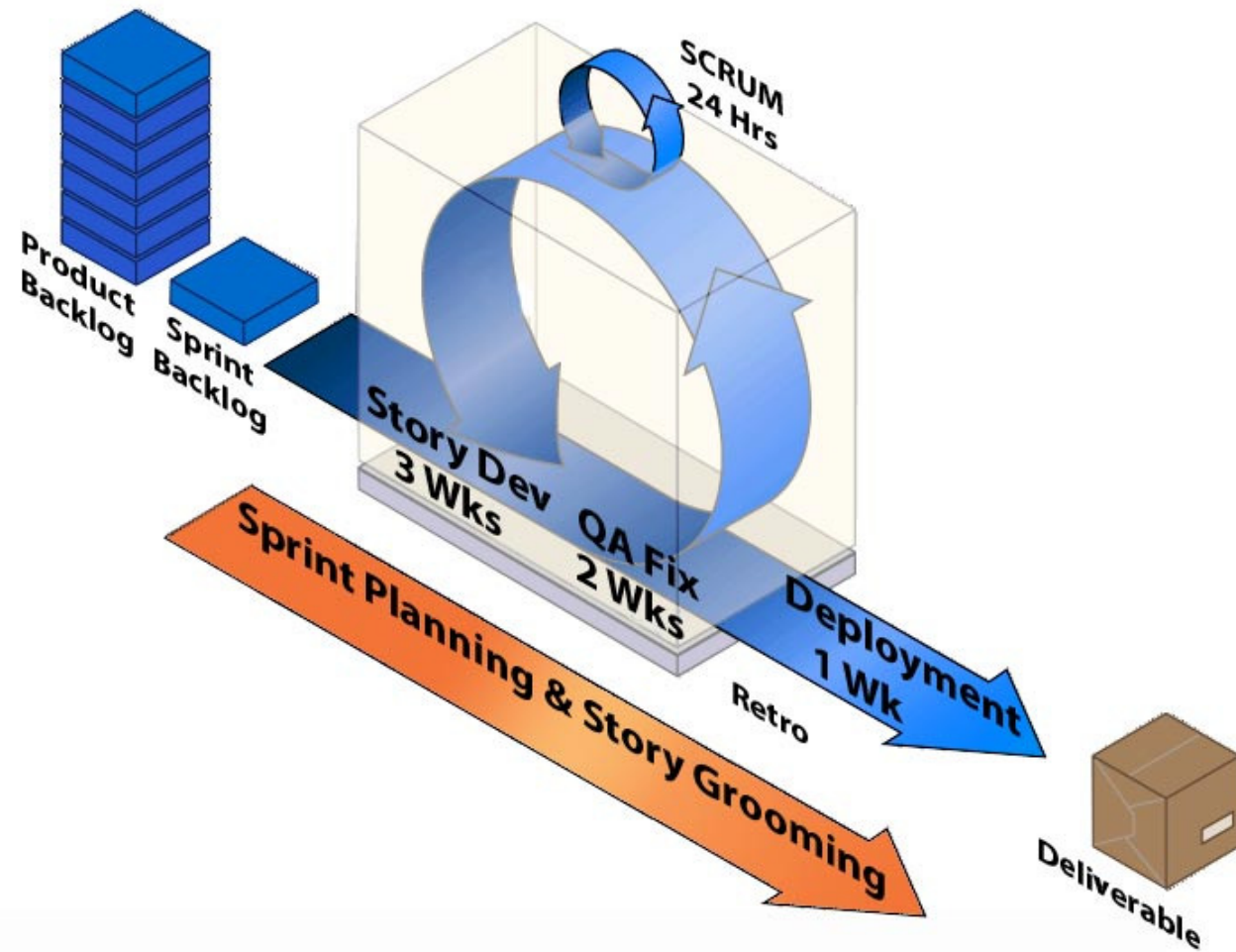
- Distribution of team
- More time to discuss/brainstorm stories and story details
- Visuals no matter what quality, Integration of UI/UX
- Epics needed spikes 1 to 2 sprints ahead of needed release
- Maximizing time was important but conflicted with one of the principles of communication
- Everyone not knowing everything but including everyone was unproductive
- Some of the questions devs would ask in story pointing sessions would need more info from stakeholders

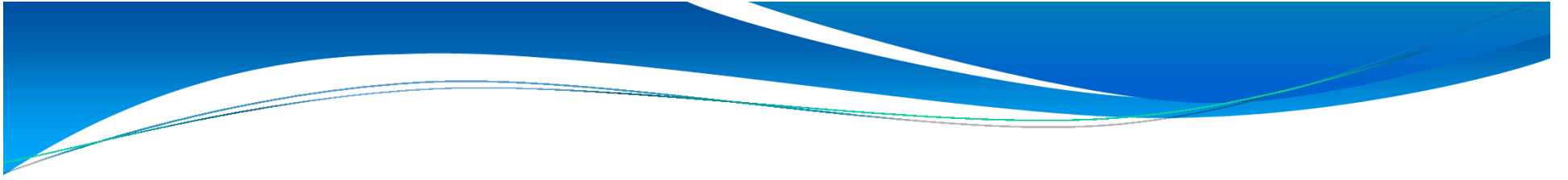


Ongoing Issues

- Product owner reviewing test plans
- Ordering of Stories – Interdependencies & QA
- Devs disconnected/non-empathetic to client wants
- Sprint vs release timing
- Devs writing their stories (technical debt)
- Devs becoming specialized
- Show & Tell

What ours looked like





Agile

How Clients WIN

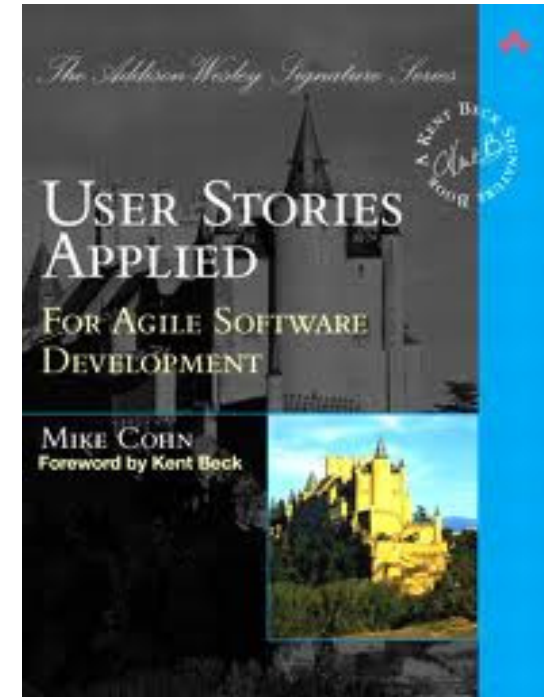
How Our Clients Benefit

- More often & consistent releases
- Feature request to use < less time
- Most important features first
- On target features
- Much better listening & collaboration
- Better relationships



Resources

- Martin Fowler – MartinFowler.com
- Mike Cohen
 - User Stories Applied - For Agile Software Development
 - MountainGoatSoftware.com
- agile.dzone.com
- Wrike.com/ProjectManagement
- SVProduct.com
Silicon Valley Product Group
- Wikipedia





Any Questions?

TJ Crawford
tj@tjcrawford.com
(775) 813-4506