Lean Philosophy adapted to Software Development in *"Lean Software Development"* Poppendieck (2003)
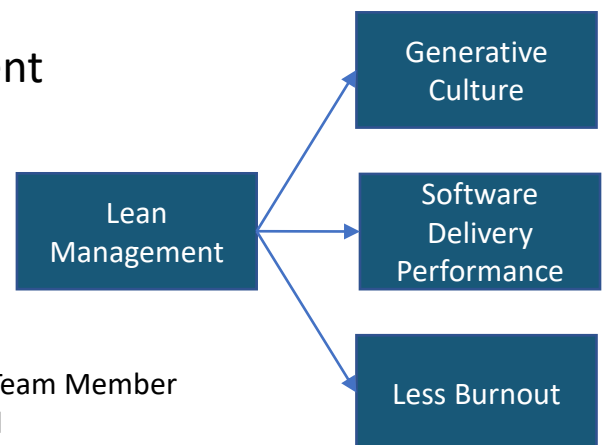
Lean Management Practices

1. Limiting Work In Progress (WIP)
2. Visual Displays for:
   - Key Quality Metrics
   - Key Productivity Metrics
   - Current Status of Work
3. Using Performance and Infrastructure Data
   - To make Daily Business Decisions

Lean Management

- Limit Work-in-Progress (WIP)
- Visual Management
- Feedback from Production
- Lightweight Change Approval

Lean Management for Software Development

- Limit WIP to prevent over burdening
- Visual Displays enable High Quality Communication
- Implement lightweight Change Management Process
  - None or Peer Review works best
  - External (CAB) does not work
  - Pair programming or intra-Team review enough
- Overcome Segregation of Duties
  - Code should be reviewed AFTER commit by another Team Member
  - Deployment to Production should be fully Automated

Lean Management → Generative Culture

Lean Management → Software Delivery Performance

Lean Management → Less Burnout

Faux Agile Practices

- Follow Common Practices
- Ignore Organizational Culture and Practices
- Grouping work into Larger Projects
- Not gathering Customer Feedback until the end

Focus on Lean Startup

- Frequently test Product with User Research
  - Its Design
  - Its Business Model
- Emphasis Experimentation into the product development process

> - Work in small batches
> - Make Flow of Work visible
> - Gather and Implement Customer Feedback
> - Team Experimentation

Lean Product Development Practices

1. Focus on working on Features that can be Completed within less than 1 week
   - Including MVPs
2. Ensure that the Team has good Visibility in the Flow of Work through to the Customer
3. Seek Customer Feedback and incorporate it into Product Design
4. Give Teams Authority to make changes to the Product

Virtuous Cycle

- Improved Software Delivery enables Better Lean Management Practices
- Working in Small Batches is key
  - Reduced cycle times leads to faster learning