



# Traditional Project Management Meets Agile: Can't We All Get Along

**Dr. Harry Koehnemann**  
*Director of Technology*  
*Rocket Gang*  
*[harry@rocketgang.com](mailto:harry@rocketgang.com)*

**Mark Coats**  
*Chief Software Engineer*  
*General Dynamics C4S*  
*[Mark.Coats@gdc4s.com](mailto:Mark.Coats@gdc4s.com)*

# Report Documentation Page

Form Approved  
OMB No. 0704-0188

Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.

1. REPORT DATE <b>MAY 2011</b>		2. REPORT TYPE		3. DATES COVERED <b>00-00-2011 to 00-00-2011</b>	
4. TITLE AND SUBTITLE <b>Traditional Project Management Meets Agile: Can't We All Get Along</b>				5a. CONTRACT NUMBER	
				5b. GRANT NUMBER	
				5c. PROGRAM ELEMENT NUMBER	
6. AUTHOR(S)				5d. PROJECT NUMBER	
				5e. TASK NUMBER	
				5f. WORK UNIT NUMBER	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) <b>Rocket Gang,14362 N. Frank Lloyd Wright Blvd,Scottsdale,AZ,85260</b>				8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)				10. SPONSOR/MONITOR'S ACRONYM(S)	
				11. SPONSOR/MONITOR'S REPORT NUMBER(S)	
12. DISTRIBUTION/AVAILABILITY STATEMENT <b>Approved for public release; distribution unlimited</b>					
13. SUPPLEMENTARY NOTES <b>Presented at the 23rd Systems and Software Technology Conference (SSTC), 16-19 May 2011, Salt Lake City, UT. Sponsored in part by the USAF. U.S. Government or Federal Rights License</b>					
14. ABSTRACT					
15. SUBJECT TERMS					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			
<b>unclassified</b>	<b>unclassified</b>	<b>unclassified</b>	<b>Same as Report (SAR)</b>	<b>14</b>	

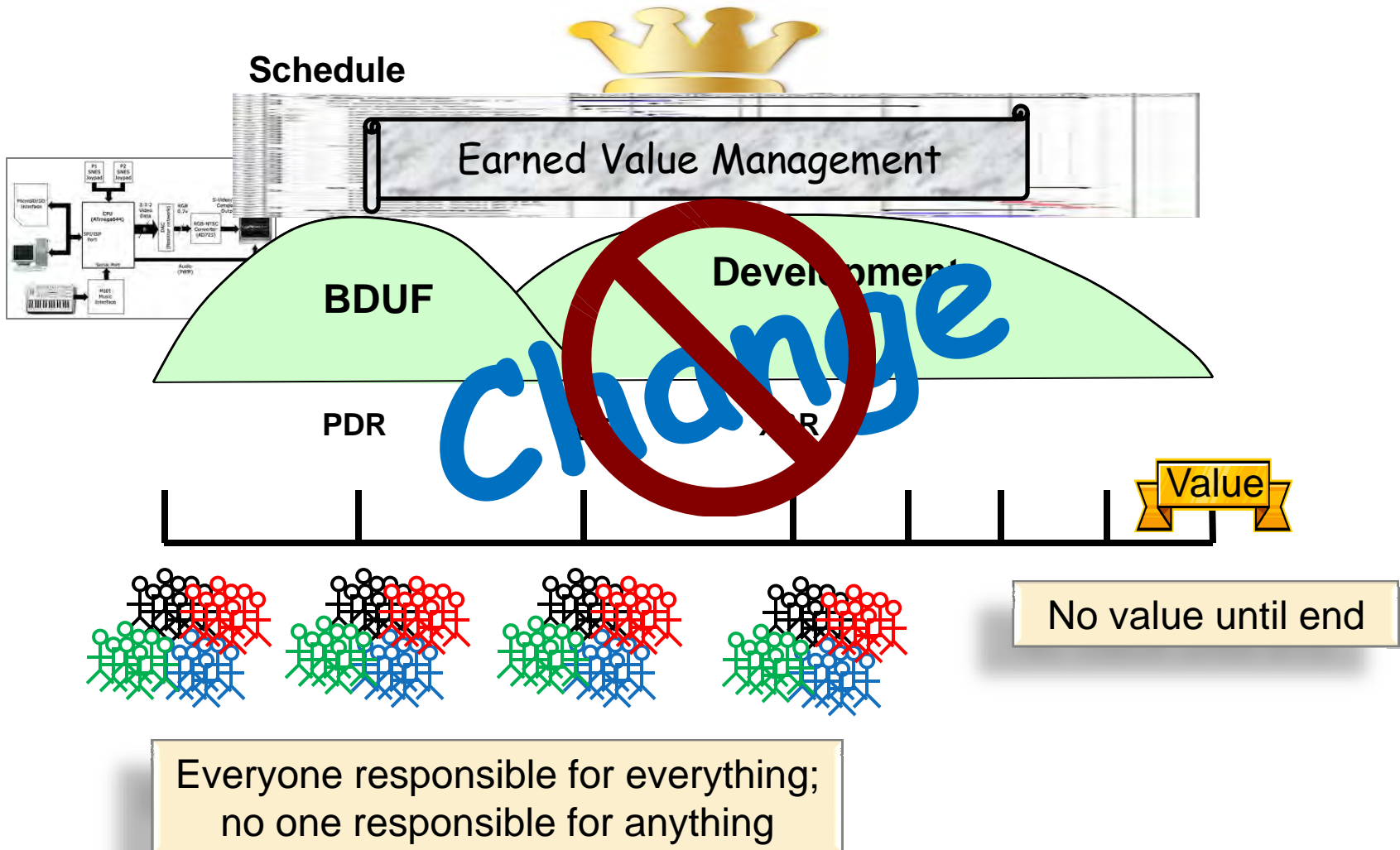
# Agenda

---

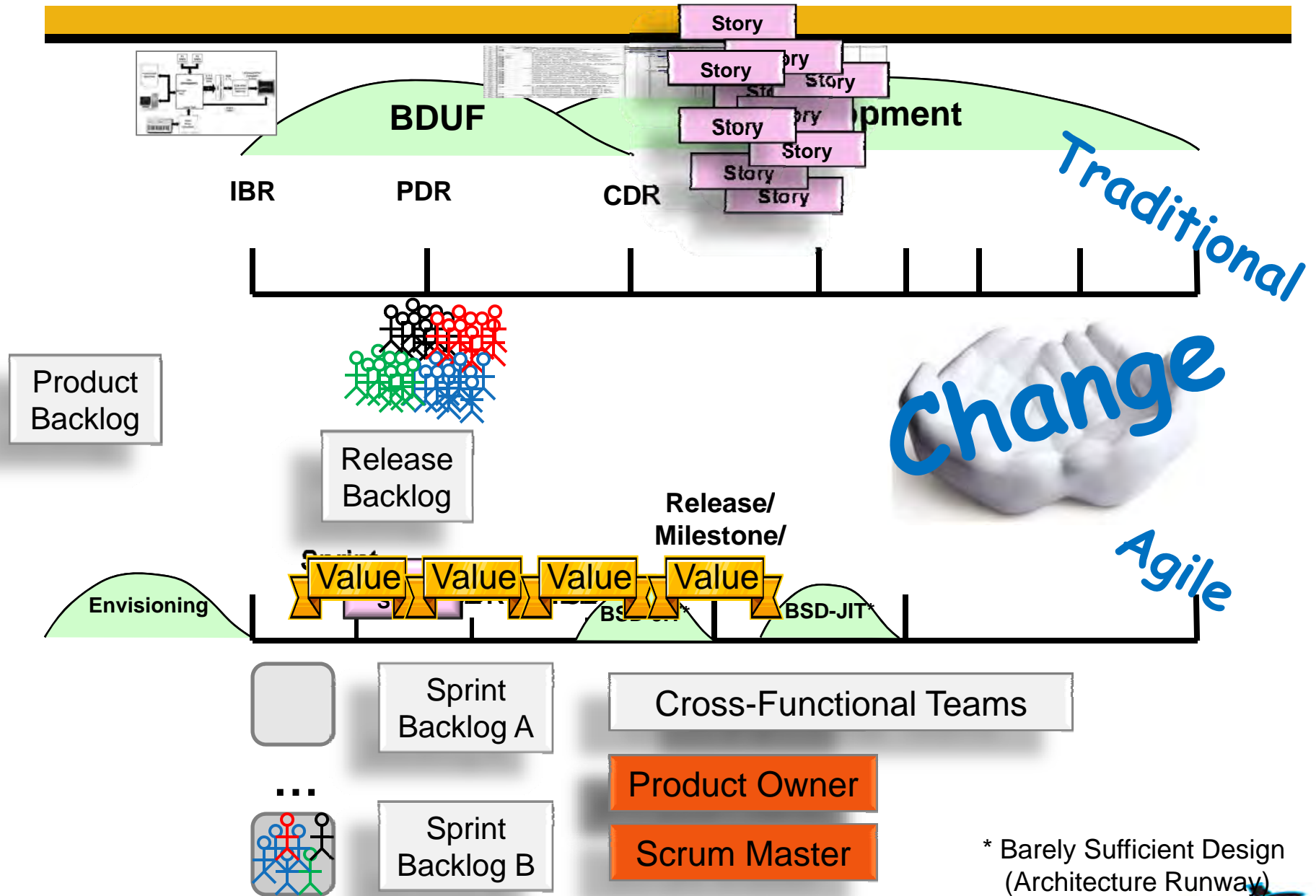
- Traditional planning and it's challenges
  - ↳ Focus on team organization, planning, requirements, and architecture
- Making traditional planning agile
- Transitioning from traditional to agile



# Traditional Development



# Agile Development



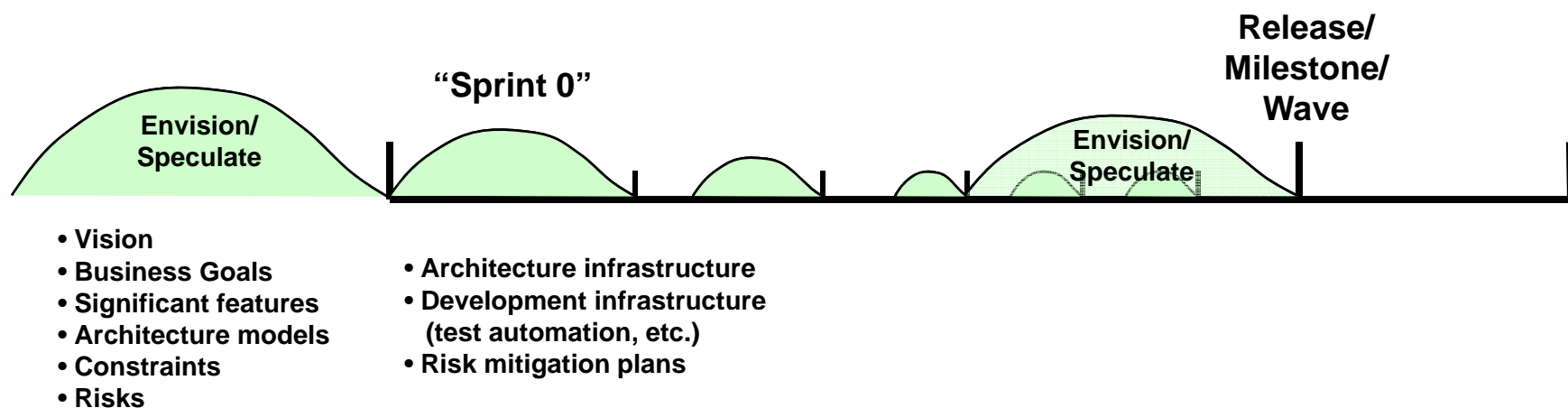
# Agile Value Points

- Defines success as delivering business value, vs. meeting schedule
- Continuous, frequent collaboration points between dev and business
- Encourages frequent feedback and learning
  - Allows projects to fail fast
  - Quickly adapts to changes
- Accommodates uncertainty, risk, change
- Plans work and organizes teams around system behavior
- Details requirements, design, schedule at right time



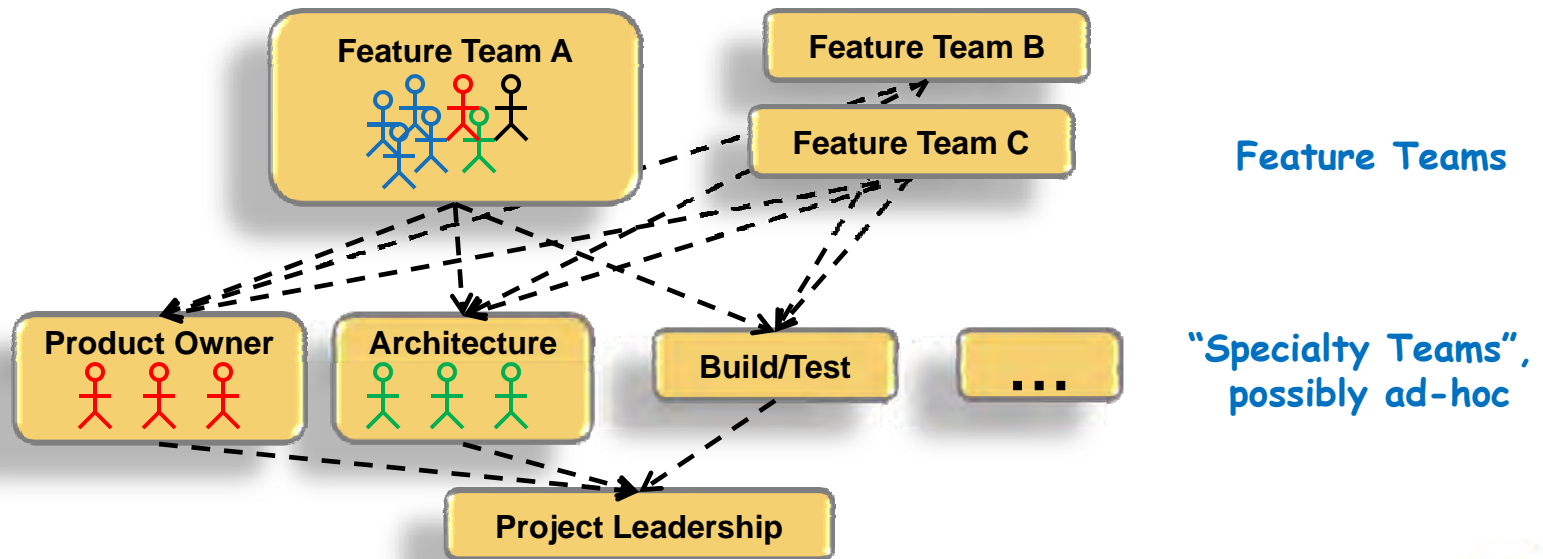
# Transitioning to Agile

- Addresses team organization, planning, requirements, architecture
  - Continually define barely-sufficient requirements and architecture runway for feature teams
- Envision/Speculate (vs. Initiate/Plan)
  - Brainstorm on product vision and promising implementation approach
- Explore (vs. Execute)
  - Use iterations to explore solution and adapt to new discoveries
  - Vs. preserve original baselines at all costs



# Transitioning to Agile Team Organization

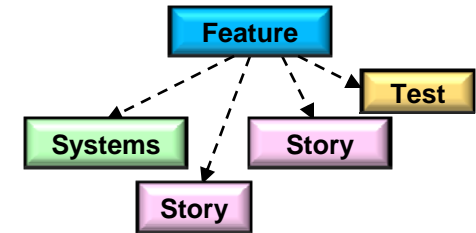
- Organize around behavior, not structure or domain
- Teams self-manage
  - Commit to features, decompose work, assign tasks
  - Desire to keep co-located!!
- Strive to push decisions out to feature teams
  - Specialty teams responsible for broader system issues and decisions
- Coupled features may require high collaboration between teams



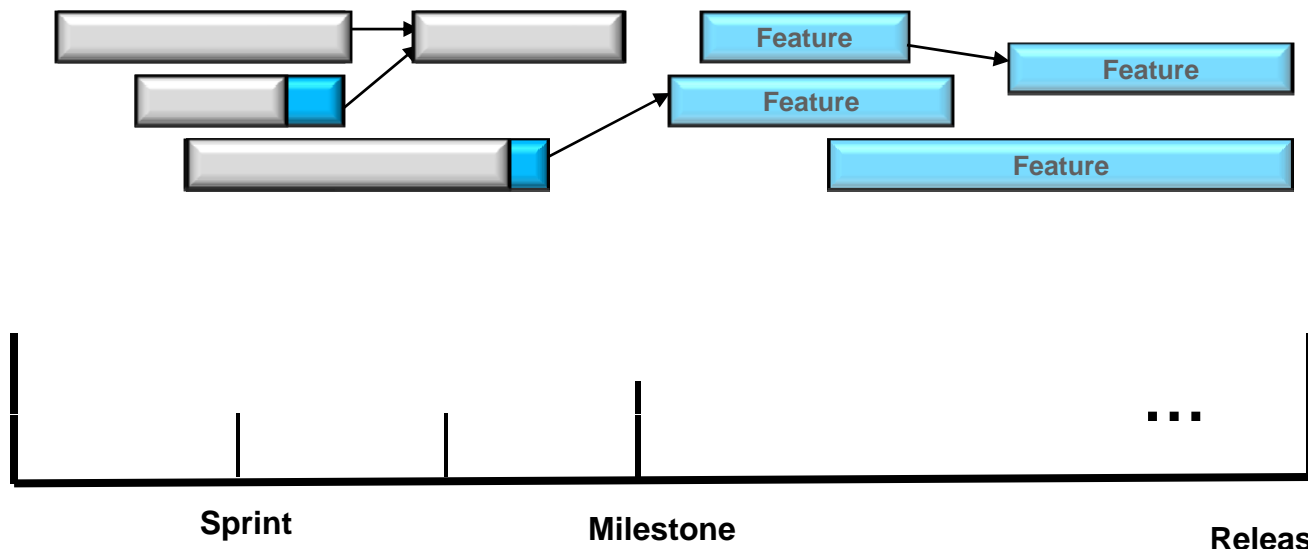
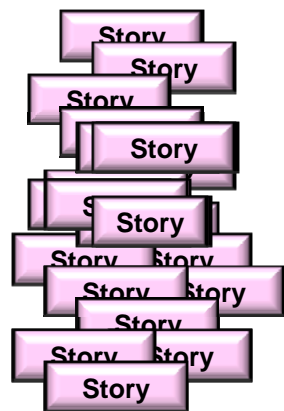


# Transition to Agile Planning

- Plans (WBS) define behavior
  - Releases plan features (20-100 day effort)
  - Decompose into stories (2-10 day)
- Items become more detailed closer to scheduling
- Use milestones/waves as synchronization points
  - Each iteration we learn more about team's ability to meet schedule

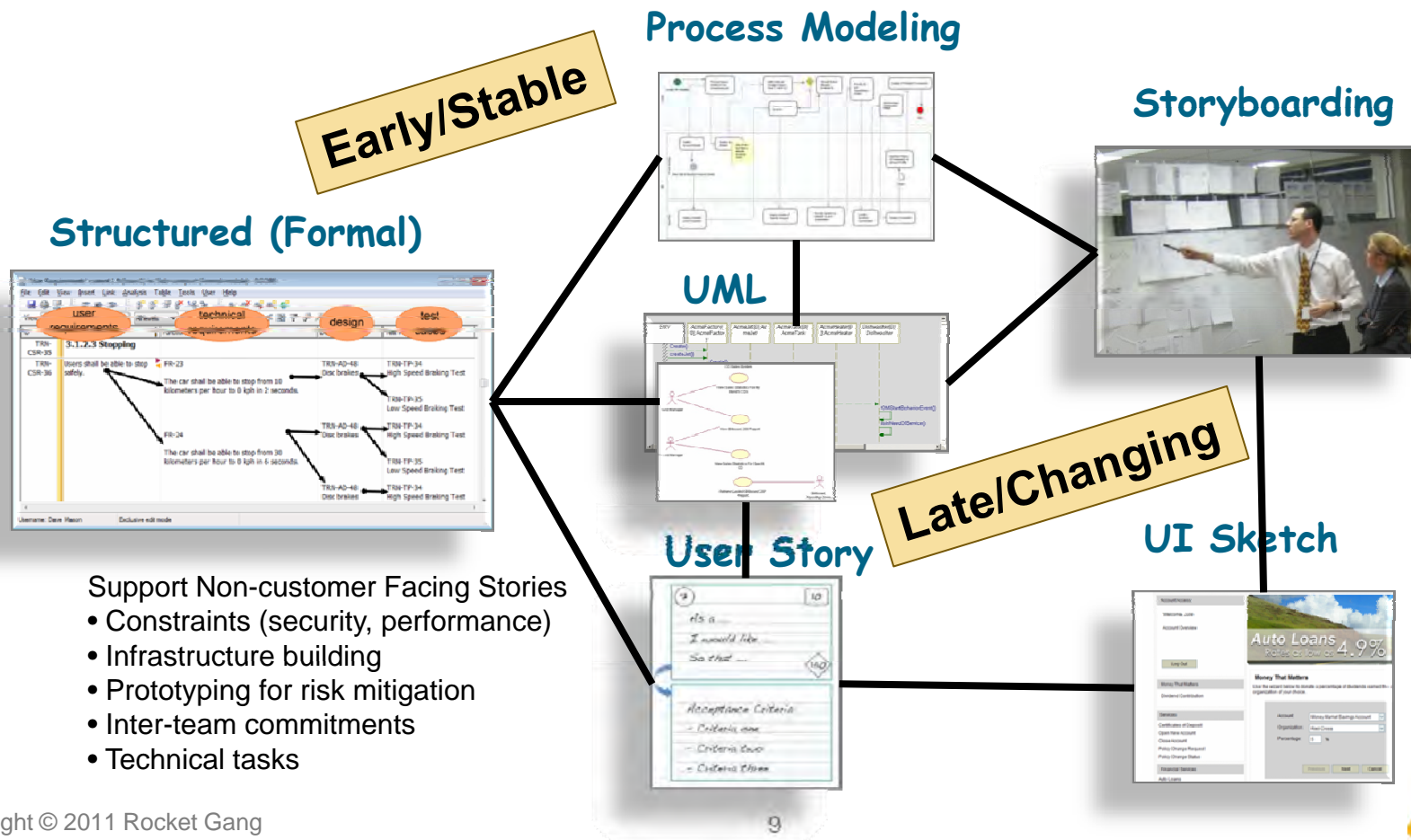


## Milestone Backlog



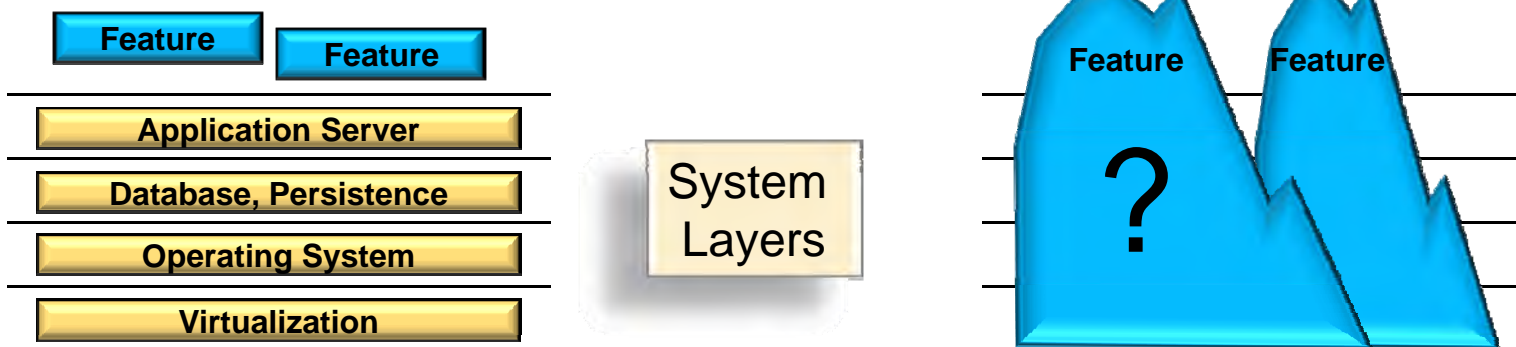
# Transitioning to Agile Requirements

- Envisioning defines constraints, significant features
- Exploring plans and details requirements at right time
- Need additional story types to support large, complex systems



# Transitioning to Agile Architecture

- Complex systems also develop their infrastructure
  - Responsible for significant functionality *below the waterline*
- Prioritize stories that extend and validate infrastructure
  - Includes rich set of automated tests
  - Similar in spirit to the “Sprint 0” agile practice (“Milestone 0”)



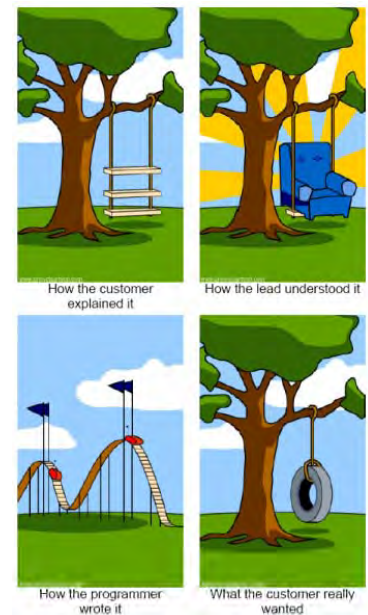
# Caution on Organizational Change

- Great process cannot make up for having the wrong people
- Need capable people with self-discipline
  - Self-motivated, passionate about solving problems
  - Multi-skilled, adaptive, continuous improvers
  - Proactive, sense of ownership
  - Communicate, collaborate
- Coaching/modeling is key to organizational change

“Most organizations build their bureaucratic rules to manage the few wrong people on the bus, which drives away the right people on the bus, which in turn increases the need for more bureaucratic rules, and so on” -- Jim Collins (2001)

# Recommendations

- Blend strengths
  - Agile: continuous feedback, learning, adaptive planning, behavior focus
  - Traditional: up-front assumptions
- Consider agility principles
  - Knowledge and running system are the goals, not documents and schedules
  - Focus on feedback and learning – problem, solution, development process
  - Use continuous, inspect-adapt cycles
  - Plan, track, execute small work
  - Organize teams around features
  - Lead teams, let teams manage themselves
- Repeatable wants to schedule, reliable wants to deliver value
  - Focus more on execution, less on management/control



# Conclusions

- Fill the gap between project plans and developer tasks
  - Ensure plans are based on reality
  - Automate progress from development work to project-level plans
- Agile changes how we:
  - Plan Behavior-focused vs. structure/document-focused
  - Measure Business value vs. percent complete
  - Manage Self-managed teams vs. Project Manager
  - Track Feature-centric vs. plan-centric
  - Build & Test Continuously vs. at end
  - Define success Make business smile (value 😊) vs. execute the plan

---

# Thank You

**Dr. Harry Koehnemann**  
*Director of Technology*  
*Rocket Gang*  
*harry@rocketgang.com*

**Mark Coats**  
*Chief Software Engineer*  
*General Dynamics C4S*  
*Mark.Coats@gdc4s.com*