



## Working as a Team



Back in May, I was watching the Giro d'Italia and the incredible selfless sacrifice of American cyclist David Zabriskie. Zabriskie was arguably one of the strongest riders in the peloton, but his teammate Andy Schleck was even stronger and stood the best chance of winning the race. Despite Zabriskie's chance to place high in the overall standings, he rode to help his teammate conserve energy and finish the race second on the podium. He and every other member of the team were focused on the performance of the team, not the performance of the individual. Effective software development teams need to have the same mutual commitment to work together towards a common goal. This level of mutual commitment doesn't happen just by telling people that we want them to work as a team. Such commitment requires several enablers. Many of the articles in this month's addition will touch upon processes for creating teams that truly work together. I'd like to offer my brief thoughts on four key enablers for creating mutual commitment of team members.

**1. Measure the project.** Over and over again I see proof that *what gets measured gets done*. If you focus your measurements on individuals, then individuals will perform. If you focus your measurements on project performance, the project will perform. Measures should focus on what is important to the customer and customers don't care about the performance of individuals. We all have internal measures and targets, but leaders should insulate their teams so they can focus on those that matter to their customer.

**2. Reward project performance not the individual.** Although easier said than done, especially in the public sector, it is the performance of the project that should be incentivized not the performance of the individual. If reward incentives focus on the individual, there is little incentive to make someone else look good. Project incentives will help individuals focus on the critical path, not just their piece of the project.

**3. Limit specialization.** Rather than have developers narrowly focused on one specialty which supports many projects, utilize people broadly to avoid bottle necks and encourage ownership. One person can not have the same level of commitment to many projects as the team members who are dedicated to one project. When possible it is better to have them committed to one or two teams and utilized in areas outside of their expertise to supplement the project.

**4. Institute Team Member Radio.** In my earlier cycling analogy, each team member is equipped with a radio that is in contact with the team director (and other coaches) in the team car. When a key member of the team suffers from a flat or has a mechanical problem, the team director signals the team to drop back to help their team member rejoin the peloton. The director also keeps close watch on the details of the race. The director warns the team of obstacles and watches for threatening tactics from the other teams. The team members themselves are communicating to let the director know what is going on from inside the peloton that can't be seen from the team car. When a team member might be suffering more than they care to admit, his team members can communicate information back to the director so she can decide what to do. This is exactly how a cohesive software development team should work. When a key area or key member of a software team gets in trouble the team needs to come to the rescue. The only way for the team to avoid obstacles, capitalize on opportunities, and react to trouble is to have constant communication up, down, and through the team.

Speaking of teamwork, we have a great team of authors lined up for this month. Starting with LTC Nanette Patton and Allan Shechet, they address achieving project success in *Wisdom for Building the Project Manager/Project Sponsor Relationship: Partnership for Project Success*. Jennifer Tucker and Hile Rutledge describe the necessity of maximizing team performance in *Shaping Motivation and Emotion in Technology Teams*, and Kasey Thompson and Tim Border teach managers to build more efficient teams in their article *The Gauge That Pays: Project Navigation and Team Building*.

Kevin Stamey  
Oklahoma City Air Logistics Center, Co-Sponsor