**4.3. Grow Small, Stable Teams**

4.3.发展小而稳的团队

Harvard Business Review studies by Wheelwright and Clark and others have shown that our productivity is highest when we work on one or two things at a time, as illustrated in Figure 13. The Effect of Multitasking on Productivity.

哈佛商业评论已经借由惠尔赖特和克拉克以及相关人士的研究表明，当我们每次致力于一件或者两件事物时，我们的生产效率最高，如图13所示，多任务对生产效率的影响。

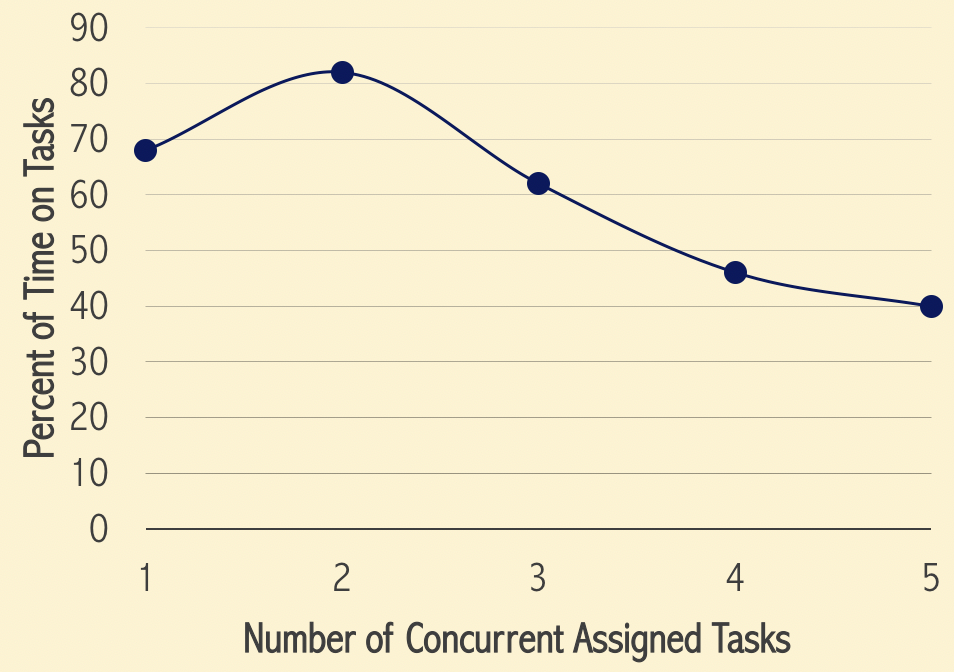


Figure 13. The Effect of Multitasking on Productivity

图13.多任务对生产效率的影响

So, Agile teams ought to focus on a single project at a time, and by working closely with our business sponsors, should be able to finish it much more quickly and accurately than in traditional environments. When the team completes the project and delivers the system into production, it will then be available to start its next project.

因此，敏捷团队理应每次专注于一个单一的项目，并且与我们的商业赞助人紧密合作，相比传统的工作环境，敏捷团队应能够更快更准确的完成项目。当团队完成项目并且把系统作为产品交付给客户时，团队将能够启动下一个项目。

Additionally, the Project Management Institute (PMI)’s Guide to the Project Management Body of Knowledge (PMBOK) tells us that to reach a high performance state, teams must go through the Tuckman model of team formation: forming, storming, norming and performing. That is, it takes time and considerable effort to get a team into the performing state. So, why on earth do we break them up and try to recreate them again for the next project?

此外，美国项目管理学协会（PMI）的指南，项目管理知识体系（PMBOK）告诉我们，如果要达到一个高绩效的状态，团队必须历经团队构建的塔克曼模型：组建期、激荡期、规范期、执行期。也就是说，要让团队进入高绩效的状态，需要花费时间和相当大的努力。那么，我们究竟为什么要把老团队拆散而且试图在下一个项目时，重新组建新团队呢？

Instead of sinking costs into forming teams for projects only to break them up on completion, we need to create stable teams as high performance units that focus on one product or project at a time. So, we create integrated cross-functional teams with team members from different departments; including business analysts, designers, developers, testers, and a project manager (or ScrumMas- ter). We allocate these core team members at least 80 percent or more to the project. Now, to maximize project throughput, we only start as many projects as there are available dedicated teams.

我们需要建立稳定的团队作为每次专注于某个产品或项目的高绩效单位，而不是为了当前的项目，投入沉没成本去构建新的团队，结果却在项目完成时解散他们。因此，我们创建和谐的跨职能团队，团队成员来自不同的职能部门：包括了业务分析师、设计师、研发人员、测试人员和项目经理（或者Scrum Master）。我们把至少80%或者更多的这些核心团队成员分配到项目中去。现在，为了最大化项目的生产力，我们只根据现有可用的专用团队，去启动尽可能多的项目。

We can then pull projects from a limited backlog of high-priority projects and allocate them as teams become available. Figure 14. Bringing Prioritized Demand to Stable Agile Teams illustrates this key concept.

接着，我们可以把高优先级的项目从有限的待办事项列表中，划拨出来，当团队资源可用时，分配给他们。图14，将优先需求带给稳定的敏捷团队，表明了这个关键概念。

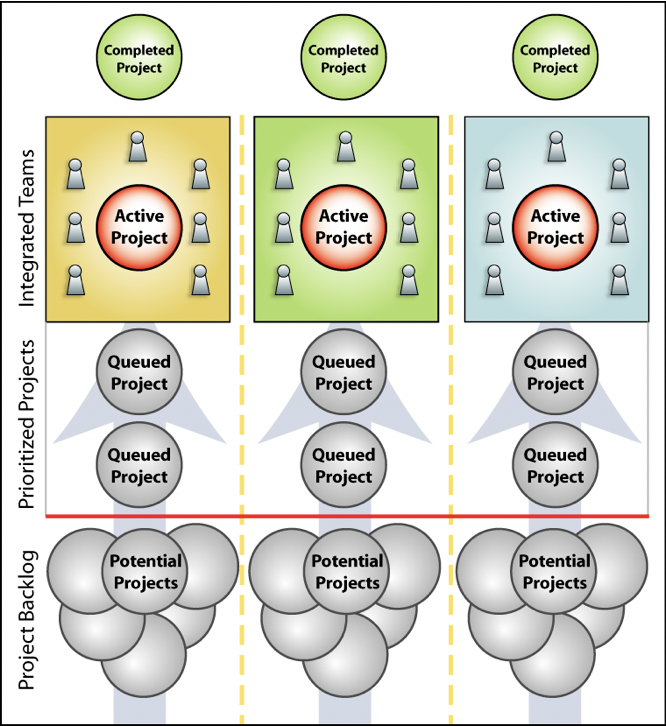


Figure 14. Bringing Prioritized Demand to Stable Agile Teams

图14.将优先需求带给稳定的敏捷团队

The bottom line here is that our stable teams form the basic unit of our allocation.

这里最重要的一点是我们稳定的团队构成了我们资源分配的基本单位。

**4.4. Build a Network of Small Teams**

4.4建立小团队网络

“... for a large organization to work it must behave like a related group of small organizations.” - E. F. Schumacher, Small is Beautiful

“…对于一个运转的大型组织而言，它必须表现的像个具有团体相关性的小组织一样”-E.F.舒马赫，《小的是美好的》

Once an Agile team is functioning well in its high-performance state, but reaches a size limit, say around nine people, we form a new team and do not continue to expand the current team beyond its size limit. To ensure that the new team is properly set up, a small seed group may break off from the original team to form its core. This core group, typically a manager, lead developer and business analyst, ensures that the team vision and culture are propagated intact to the new team. Importantly, the core original team continues intact as a stable team.

一旦敏捷团队在高绩效的状态下运行良好，却达到了组织的规模限制，比如说团队大约有9名成员了，我们将不会继续扩大现有团队的规模限制，并且会构建一个新的团队。为了保证新团队被正确建立，一个小的种子组可能会从原团队中脱离，形成新团队的核心。这个核心组，通常包括了项目经理、最重要的研发人员和业务分析师，确保把原团队的愿景和文化完整的传播给新的团队。重要的是，作为稳定的团队，核心原始团队继续保持不变。

Following this approach avoids team bloat, and also ensures that teams retain their agile qualities. Figure 15. Conquer-and-Divide Approach to Scaling Teams illustrates this approach.

这种方式避免了团队的臃肿，也确保了团队保持住了他们的敏捷性。图15，“分治法定标团队”表明了这种方式。

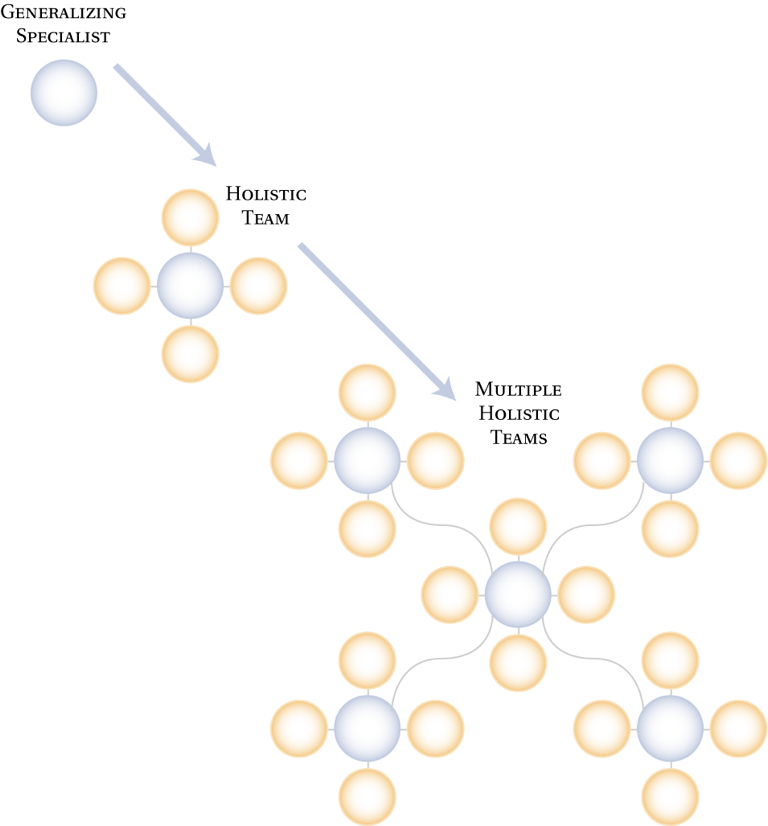


Figure 15. Conquer-and-Divide Approach to Scaling Teams

图15.分治法定标团队

Now, how can we ensure that this team structure remains flexible enough to adapt to rapid change? An excellent way is employ a variation of the feature teams invented by Jeff De Luca for teams implementing the Feature Driven Development (FDD) Agile methodology.

现在，我们如何确保团队结构保持住灵活性以能够适应快速的变化呢？有个很棒的方法是利用特性团队的变化来执行Jeff De Luca所创造的特性驱动开发（FDD）的敏捷方法。

In FDD, a chief programmer assumes responsibility for delivering specified features during an iteration. She then identifies the class owners – owners of specific code modules and pulls them together for the 1-3 week duration of the iteration to deliver the specified features. Besides the core group maintained for consistency and continuity, some members in the team may change from Sprint to Sprint depending on the functionality to be delivered, as illustrated in Figure 16. Dynamic Team Membership.

在FDD中，主程序员承担在迭代过程中交付指定特性的责任。她识别出类的所有者——特定代码模块的所有者，并在迭代的1-3周内将它们组合在一起，以交付指定的特性。除了核心组保持一致性和连续性，团队中的一些成员的“位置”，可能会根据所交付的功能，在不断的“冲刺”中发生变化，如图16所表明的“动态团队成员关系”。

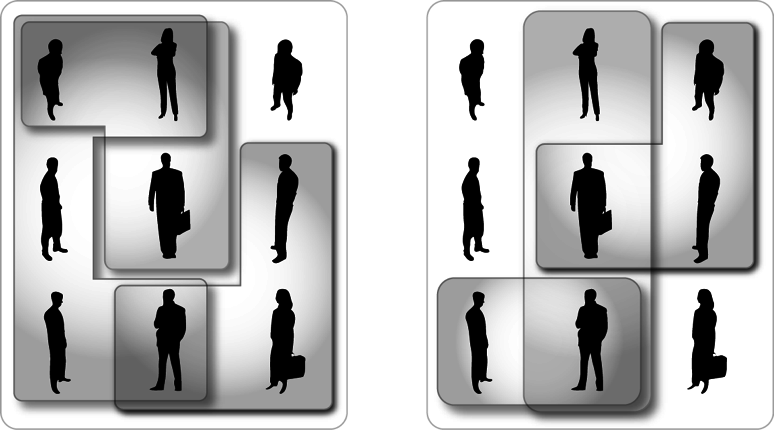


Figure 16. Dynamic Team Membership

图16.动态团队成员关系

To create connections between teams, noted Agile expert and author Johanna Rothman recommends a network instead of a hierarchy as the first step, as illustrated in Figure 17. Network of Small Teams.

为了在团队之间建立联系，著名的敏捷专家和作家Johanna Rothman首先推荐了一种网络而不是等级制度。如图17所表明的“小团队网络”。

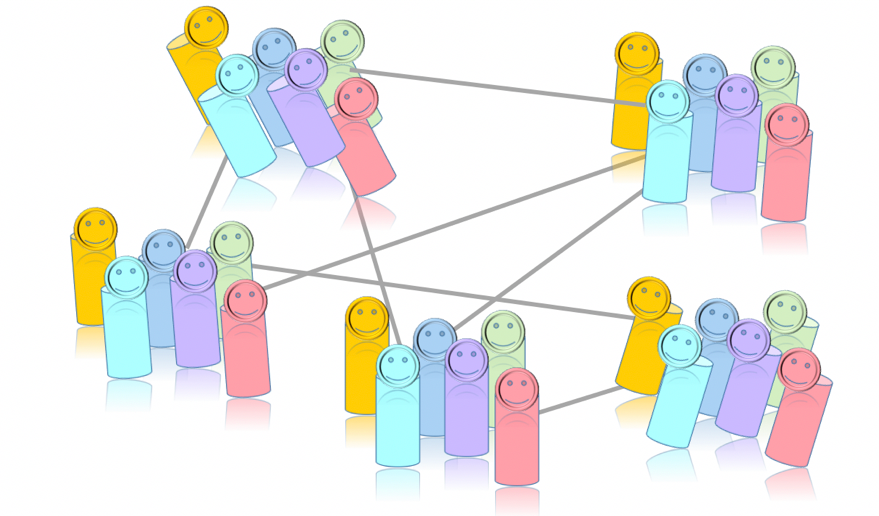


Figure 17. Network of Small Teams

图17.小团队网络

As multiple teams are formed, we want these networks to emerge organically from the teams themselves, and not be dictated top-down. For instance, team members may connect around a shared discipline, as in a working group dedicated to improving test automation across teams. Or, they could connect as kaizen continuous improvement group to implement retrospective action items across teams.

随着多个团队的形成，我们希望这些网络能够在团队中自然的出现，而不是呈现出自上而下的支配。比如，团队成员可以关联在某个共同的行为中，像横跨团队之间的、专注于改进自动化测试的工作组中。或者，他们可以关联在类似于横跨团队的持续改进小组中，以实施项目复盘。