

Break Silos, Build Chapters

Ashok Kumar. S

ashok.jan31@gmail.com

Abstract

As enterprises mature in their agile journey and move towards becoming more distributed cross-functional teams, testing that was once done by a single central team has transformed into a distributed capability where testers are embedded within the teams.

While this has many benefits, some potential challenges surface. When testers disperse across multiple teams they can start to feel isolated and collaboration with peers diminishes.

Duplication of tools and frameworks may emerge, transparency decreases and ways of working and practices become highly siloed and inconsistent.

To solve the lack of connection, inspiration, and knowledge-sharing, one solution is to work with testing chapters to connect testers into a shared context - a community. They provide a platform to share and organize skills and expertise as well as serve as a safe community of like-minded people to share new ideas, test experimentations, and receive feedback from peers. The term chapters in some enterprises is also called Community of Practice (COP) and for clarity of this paper, we will continue with the terminology of Chapters.

I will share our learnings from building these testing chapters, the opportunities they can provide to collaborate across teams, and the value they create. I will also cover challenges we faced, such as:

- The mindset shift needs to happen within the teams as well as with the stakeholders connected to them.
- The false notion is that external chapters are not required or valuable since agile is about autonomous teams.
- How to plan and account for time spent on chapter activities and who should take the lead on organizing and coordinating them.
- How to bring these learnings and best practices back into the teams.

I'll share how we overcame them, used gamification to keep engagement high, and made the chapters fit into an enterprise-wide agile team structure.

Biography

Ashok Kumar is the head of Quality Assurance & Engineering for Transunion Africa. In this position, he leads Testing and Quality Assurance for the Africa region which enables Transunion to roll out innovative credit bureau solutions to our African customers at scale and with high quality. Ashok works with multiple agile teams within the company and coaches them on the quality mindset. Ashok has nearly 2 decades of experience in testing across multiple industries such as financial services, retail, & credit bureau.

Copyright Ashok Kumar 25082023

Introduction

In today's rapidly evolving software development landscape, agile methodologies have emerged as a game-changer, revolutionizing the way teams collaborate and deliver high-quality software. At the core of Agile lies the principle of continuous improvement, emphasizing iterative development, flexibility, and adaptability. Cross-functional agile teams bring value to the business in terms of faster time to market and self-contained teams. However, one crucial aspect that often demands careful attention is the inability of the team members in continuing to collaborate, share and improve professional expertise within their skill groups.

In this paper, we delve into the effectiveness of creating chapters for testing professionals within an Agile Enterprise and explore how they enhance the efficiency of testers embedded within a highly distributed and scalable Agile organization. By examining the benefits, challenges, and best practices associated with testing chapters, we aim to shed light on how this approach can elevate the quality and velocity of software development. Throughout the article, we will explore the primary reasons why testing chapters are gaining traction in agile organizations. We will discuss how these chapters empower teams to build a quality culture that is focused on collaboration and implementing best-in-class quality engineering practices across the agile silos and standardizing these practices within the teams.

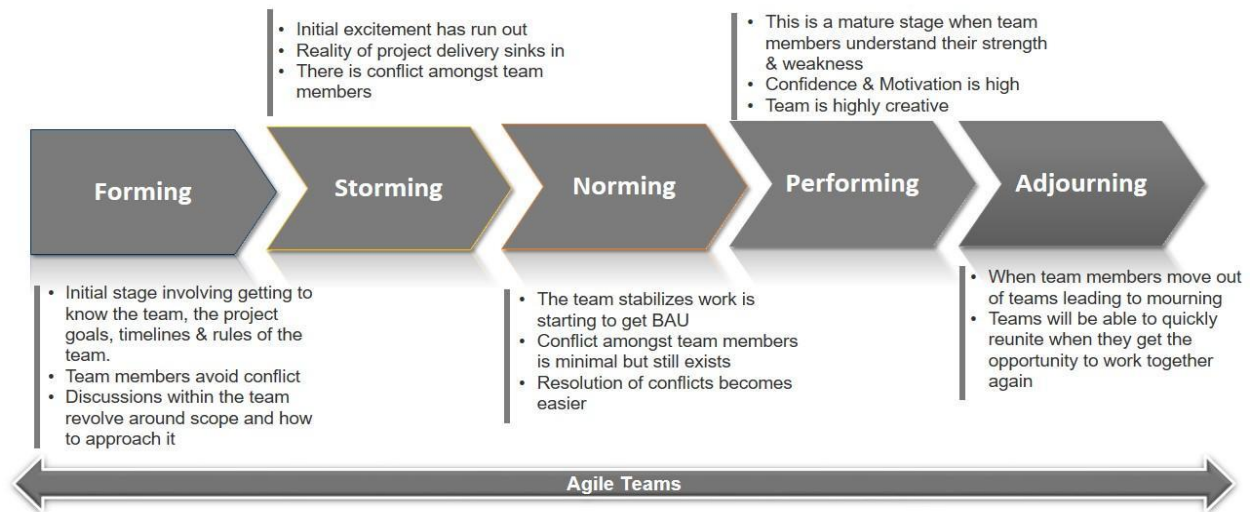
Additionally, we will address the challenges that Agile teams may encounter while implementing testing chapters, such as scaling across large teams, balancing individual responsibilities with collective goals, integrating testing chapters into existing Agile frameworks, and planning for the time required to build, scale and operate these testing chapters.

By providing insights, practical advice, and real-world examples, this paper aims to equip agile practitioners, project managers, and quality assurance professionals with the knowledge needed to leverage testing chapters effectively. Whether you are looking to establish a testing chapter or seeking to optimize an existing one, this exploration will serve as a valuable guide in harnessing the power of testing chapters to drive superior software quality and elevate the overall performance of agile teams.

1 Agile team Structure

An agile team is a cross-functional team that is self-contained with all the skills required to deliver the product's next iteration. Chapters are a team of people with similar skills or job roles or professional interests.

A group of people organized as an agile team or chapter typically goes through different stages of development and transformation. Agile evangelists often draw inspiration from Tuchman's study on "stages of group development".



Forming

The initial stage of team building is when the communication between the teams mostly revolves around scoping, introduction, timelines & challenges of meeting those timelines. Team members avoid conflict at this stage as they are new to the team and most likely will try to fit in.

Storming

This is the second and most troublesome stage of team building where teams generally tend to conflict and compete with each other as individual personalities emerge. Productivity dips as there is a lot of energy spent on conflict resolution. Team members tend to disagree on goals, and subgroups are formed under other strong personalities.

Norming

In this stage, consensus begins to evolve within the team and there is more clarity about who the leader of the team is. There is a sense of cohesion and unity among team members. Team performance increases gradually at this stage as the members learn to cooperate and work as a team. The teams' overall productivity is also on the rise considering they can focus on more productive work rather than resolving conflicts.

Performing

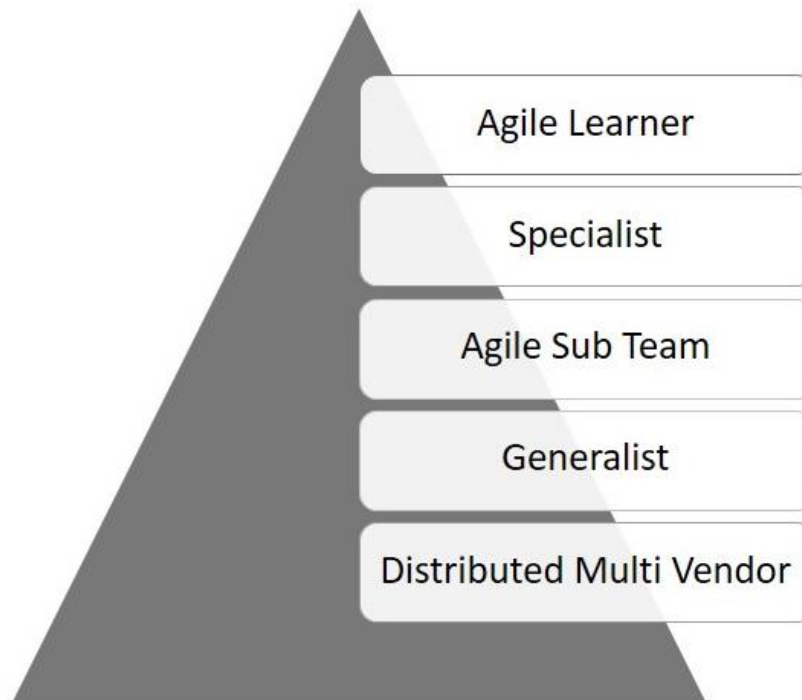
This is a mature stage where the teams reach the highest level of collaboration and cohesion amongst themselves. There is a clear and stable structure and members are aligned to the joint team goal. Conflicts still emerge however they get resolved more quickly and constructively.

Adjourning

In this stage, most of the team's goals have been accomplished. The emphasis is on closing out any final tasks that need to be completed and documenting efforts and results. As the work diminishes, individual team members move to other teams. There may be regret as the teams disband however it is expected that teams take the time to celebrate their achievements – having positive shared experiences will make it easier, if you work with some of the same people again in the future.

1.1 Transformation over time

The structure of agile teams themselves has transformed and evolved over time as a response to the business need to build high-quality software products.



1.1.1 Agile Learner

This is the initial stage of team structure agile teams typically start with. When a team transitions to Agile for the first time they need a lot of support and guidance. This structure of teams would require an agile coach to be dedicated to the teams to help them navigate the learning curve. Teams at this stage typically follow a combination of waterfall and agile practices to enable them to navigate the transition more easily.

1.1.2 Specialist Teams

This is the second stage of agile teams' transformation. They are now a team of specialists where every team member is an expert in his/her specific skills. Although teams following this structure usually deliver high-quality software they often struggle with delivering high-quality work within the framework of the sprint deadline. Also, the other challenge with this structure of teams is that team members often sit idle waiting for their specialist task.

1.1.3 Agile Sub Team

These are team structures that are made up of smaller units of teams that are focused on specific skills. Although this defeats the purpose of having cross-functional teams, some merit in this is each sub-team will have responsibility over a specific area of the application. There is a drawback in terms of too much

time spent in handover work from one sub-team to another, resulting in the overall project timeline being extended.

1.1.4 Generalist

The team consists of members that can pick up any task as they are cross-skilled and work very well together within their team. The members who are part of this structure of teams are usually very passionate and are driven by the opportunity to learn multiple skills. Also, this structure of teams is usually small in terms of size and would have team members working together in this fashion for extended periods resulting in high levels of mutual trust and cohesiveness.

1.1.5 Distributed Multi-Vendor

This is a new structure of agile teams where team members are distributed across multiple geographic locations and even come from different companies. They are aligned on the common sprint goal and work towards delivering it at the end of every iteration. This structure is complex in implementation considering that team members are not only remote but also from different parent companies. There can be an initial challenge in getting these disparate team members to be aligned on the common goal however this model is the most cost-effective way of implementing an agile multi-vendor team model.

The other challenge with such a distributed model is that the more distributed the squads or teams become the more they start to duplicate practices, frameworks & tools. An essential transformation of team structures to overcome this challenge is the formation of chapters that draw people with common professional interests into a group that can foster the standardization of practices across these distributed multi-vendor agile teams.

2 Cross-functional teams boon or bane

Cross-functional teams can be both a boon and a bane, depending on various factors and how it is implemented and managed. Let's explore both perspectives:

Boon

- **Diverse Expertise:** Cross-functional teams bring together individuals with diverse skill sets and backgrounds. This can lead to a broader range of perspectives, ideas, and problem-solving approaches, enhancing creativity and innovation within the team.
- **Self-contained teams:** Since cross-functional teams have diverse skills within the team there is very little dependency on external people which makes the decision-making process much faster.
- **Holistic problem-solving:** With members from different functions or areas, cross-functional teams can address complex problems from multiple angles. This holistic approach increases the likelihood of identifying comprehensive solutions that consider various aspects of the issue.

Bane

- **Siloed:** The more cross-functional teams are, the more they tend to become siloed. This results in duplication of best practices across teams
- **Skill Stagnation:** The drive towards nurturing more generalist talent within agile teams can result in skill stagnation whereby team members within the agile team who were once specialists now strive to move away from their specialization and become more generic professionals.
- **Lack of Standardization:** Cross-Functional teams give their team members a false sense of comfort in terms of the practices and processes they adopt. A direct outcome of the siloed manner in which cross-functional teams operate is that they lack a platform to collaborate, share and learn best practices with other team members within their skill area working across multiple agile silos.

Cross-functional teams are a key aspect of any agile structure and it becomes necessary for us to find solutions to the challenges around teams becoming siloed and the lack of standardization of best practices.

3 Enabling distributed testers through Chapters

When testers disperse across multiple cross-functional teams, they can feel isolated and collaboration with peers diminishes. Duplication of tools and frameworks may emerge, transparency decreases and ways of working and practices become highly siloed and inconsistent. A solution to improve knowledge sharing and standardized testing best practices across multiple agile teams is to form testing chapters within the agile structural framework.

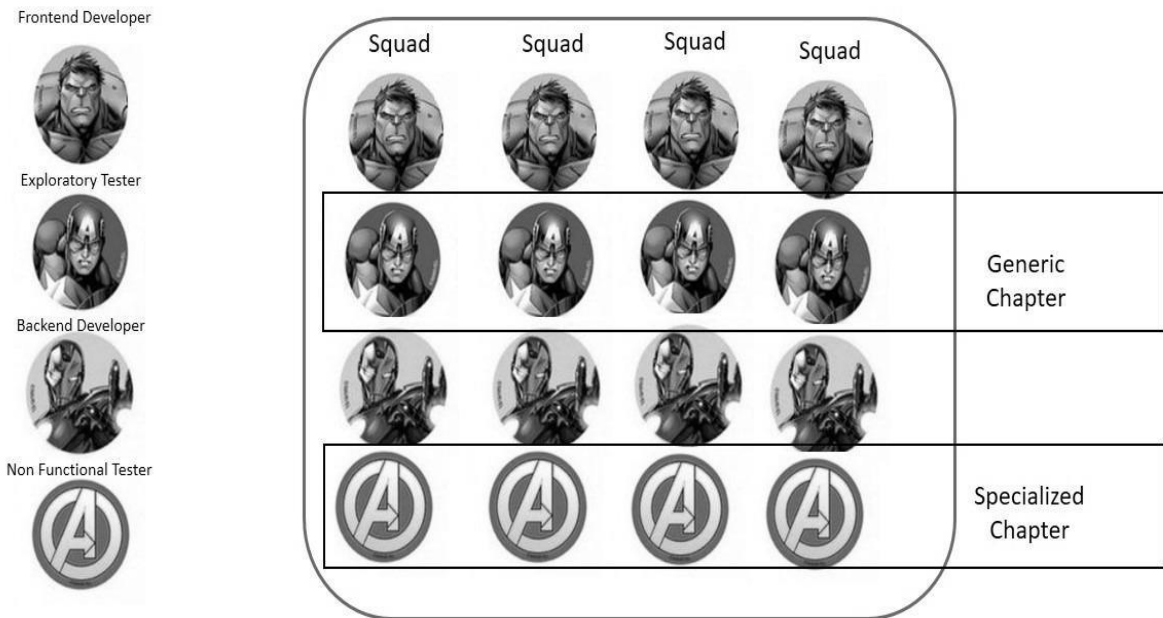
A chapter is an open group of people working for the company (including vendors) who have the same job role or are interested in the same topics supported by the chapter. The chapter can be based on expertise or based on a topic. Some chapters are officially established, and some might appear naturally as more people are interested in the solution to the same problem. Chapters are not limited to an individual project or program but are more of an amalgamation of similar skilled professionals across different projects/programs/agile teams/squads. Chapters can in certain organizations be termed as a community of practice (COP) which is again a community of similar skilled professionals practicing their skill. For the clarity of this technical paper, we will be using the term Chapter and in certain contexts can be similar to how a Community of Practice (COP) operates.

These chapters typically exist outside of the individual cross-functional teams but they are within the agile boundary of the enterprise. Testing chapters are a cohort of testers drawn from individual teams. These chapters serve as a safe sandbox where testers can practice and collaborate on their specialist skills while still continuing to wear the generalist hat within their scrum teams.

Chapters also act as incubation centers where testers can share and experiment with new ideas, and get feedback from other chapter members. Testing Chapters can also be an effective tool to standardize best practices of testing, cascading them across the different agile teams. This ensures that tools, processes, and frameworks used for testing within the teams don't get duplicated. There is also the opportunity to save in terms of cost and effort for the organization by ensuring reuse and optimization.

4 Testing Chapters our perspective

There are many categories of chapters that can be formed, in this section I will discuss on a high level what each of these are and in what situations can they be useful.



4.1 Generic chapter

Chapters can also be formed to be more generic in nature. This can be an overall testing chapter that includes team members performing various skills encompassing automation, performance, and functional testing skills. Such generic chapters can help with the cross-skilling of team members across multiple specializations within Quality Engineering. These generic chapters can be an effective lever for the optimization of skills across testing.

4.2 Specialized Chapter

These are chapters of testing that are focused on a specialized skill within the testing domain. An example of such a chapter is the one dedicated to performance testing where performance testers across the various scrum teams collaborate and share their expertise. The advantage of having such specialized chapters is that it helps in building a knowledge repository and best practices for specialized skills. Specialized chapters can also be a sandbox where specialist skills are groomed and nurtured more effectively considering that within agile teams there is an emphasis on generic skills.

4.3 Internal Chapter

Internal chapters are groups formed with team members belonging to the same company internally. These internal chapters help with more flexible ways of data sharing among the different chapter members as they all belong to the same organization. However, the drawback with this form of internal chapters is that they lack the outside perspective of solutions to problems being dealt with by the various chapter members.

4.4 External chapter

External chapters are groups formed with team members who are external and belong to different organizations. These chapters are effective in implementing unique solutions to challenges as they bring varied perspectives from people inside and outside of the organizations. External chapters also can benefit from the fact that they can draw in experts from the field not limited to the company to which the experts are attached in a professional setting.

5 Chapter Leader

A key role in chapters responsible for forming, organizing, and running chapters is the chapter leader. Some of the key responsibilities of this role are below

- To establish a lively chapter
- To invite people to be part of the chapter
- Share outcomes of the community with management, other chapters, squads, and people,
- To coordinate and moderate meetings,
- To run chapters with transparency,
- To share expertise within the community.

The essential traits of a successful chapter leader are as below

- Exceptional Leadership abilities as the chapter spans multiple agile teams and squads
- Possess deep specialist skills that are connected to the chapter
- People leadership, developing talent across the business and outside the organization
- Championing and influencing a culture of growth mindset
- Craft enablement through coaching, mentoring, facilitation, and teaching
- Technical acumen in capability delivery and thinking critically about the application of technologies to solve customer experience problems
- Effective problem-solving and decision making

6 Challenges in organizing chapters

6.1 Internal challenges

Some of the challenges that exist internally within chapters are

- Chapters are more loosely organized teams and without motivation among the chapter members they can become dysfunctional
- Chapters aren't a one size fits all model. An example is the Spotify model of agile teams might not work for all companies

6.2 External challenges

- Chapters do not necessarily have a direct impact on the sprint goal of a team. This often creates friction in the overall agile enterprise on the need for them
- Organizing chapters can become complex with collaboration and communication becoming a challenge between the various teams.

7 Solution from the trenches

In this section, I will discuss the strategies that have worked for us when building chapters and the solutions that were implemented to make our chapters more effective.

7.1 Management Air cover

The success of any chapter is directly proportional to the extent of management support that they enjoy. Since chapters are external to the agile teams that directly work on business outcomes and goals, often time's chapters and chapter goals get side-lined, because no product owner or someone in a similar role is setting aside time for it. With the right management support and cover, chapters can be effective and focused on achieving their goal. Management support is necessary for chapters to focus on the right problems to solve.

7.2 Gamification of chapters

Gamification is about leveraging the motivational aspects of games and applying them to real-world situations. It draws on the innate human desire for challenge, achievement, and rewards. By incorporating game-like elements such as points, levels, badges, leaderboards, and quests, gamification taps into our natural inclination to participate, compete, and succeed. Gamification in chapters can effectively engage chapter members, motivate them to work on tasks, and bring them to closure.

Leaderboards in gamification are a feature commonly used to promote competition and motivate players or users in a game or gamified system. They display rankings of participants based on various criteria, such as scores, achievements, levels, or progress. Leaderboards typically show the top performers or highest-ranking players, allowing others to see where they stand in comparison. This fosters a sense of challenge, encourages engagement, and drives individuals to strive for improvement. By showcasing the achievements of top players, leaderboards create a competitive environment and spur friendly competition among participants.

Chapter leaders can implement leaderboards and award members badges based on their accumulated points. The points are allocated to individual chapter members on an ongoing basis as they complete specific chapter tasks and initiatives. Points can be collected by chapter members for performing specific tasks. Some of the common tasks that chapter members can get involved in to accumulate points are specified below.

- Complete chapter-specific tasks and activities such as
 - Building an automation framework
 - Design a continuous testing pipeline
 - Formulate meaningful testing metrics to measure within teams.
- Ensure implementation of best practices and frameworks built by chapter within individual agile team/squad
- Champion chapter initiatives within and across squads
- Conduct training programs for chapter members to upskill and reskill

A sample leaderboard for organizing and running chapters.

Chapter Member	Points Accumulated	Badge
Jack	75	 
Ana	60	
Mohammed	52	
John	43	 
Sharon	38	
Jane	36	 
Watson	30	

7.3 Chapter time

This is time that is allocated for team members to work on chapter tasks. In a typical agile organization, this time is often not accounted for, and it quickly turns into a situation where activities specific to the chapter get side-lined or down-prioritized. It is good practice for chapter leaders to work with the Scrum masters of individual agile teams and agree on a specific percentage of the time being set aside for chapter activities. This time that is set aside for chapter work needs to specifically be called out in the PI planning for the teams. A practical allocation of time between sprint deliverables and chapters is 80:20 where 80% of the team's capacity is allocated towards achieving the sprint goal whereas 20% of capacity goes to chapters.

7.4 Chapter Tasks

Chapters can be involved with tasks that horizontally impact all or some of the teams within the overall agile enterprise. These tasks add value to the individual teams and can help them accelerate their velocity and capability to deliver business value for the end customer.

Some of the tasks that can be accomplished within a chapter are as below.

- Building an automation framework that can be implemented within the teams
- Functional and non-functional tool optimization and rationalization
- Designing a test architecture suitable for testing different technologies UI, API, DB, etc.
- Developing coding standards for scripting with the automation tools
- Building continuous testing pipelines
- Designing branching strategies for test code
- Conducting hackathons and technical meetups for testing professionals
- Plan and conduct test process reviews to standardize processes across teams

8 Metrics & KPI

Some of the key metrics that can be tracked to measure the success of chapters are in effect similar to the metrics that apply to an agile team as well.

Lead Time:

Lead time measures the total time from the moment a story enters the system (in the backlog) until it is completed as part of a chapter sprint. It measures the total time for a requirement to be realized and start earning value – the speed of your value chain.

Cycle Time:

Cycle time is a subset of lead time – it measures the time for a task to go from “started” or “in progress” to “done”. If cycle times are longer than a sprint, teams are not completing the work they committed to.

Chapter Happiness Index

Each chapter member is asked to vote on a scale of 1 to 5 on how happy he is in the chapter with 1 being least happy and 5 being most happy. The average of all the chapter members’ happiness index will provide the happiness index of the overall chapter. The chapter happiness index is a measure of how motivated and happy a chapter is functioning. This is an important measurement because a happy chapter can get a lot more accomplished.

9 Extend testing chapters

Our experience in building chapters for testing can also be extended to organizing chapters for other skills within agile teams. The primary tenets of organizing, monitoring, and running testing chapters can be extended to chapters of any skill. The best practices for running these testing chapters can be cascaded to other chapters. Such sharing of best practices among chapters can help chapter leaders accelerate the time required to build and organize them. A ceremony called “chapter of chapters” is similar to the “scrum of scrum” where chapter leaders across multiple chapters participate to share updates on their respective chapters.

10 Benefits

Benefits can be quantitative, such as a reduction in expenses, increase in revenue, increase in market share, or risk reduction, e.g., compliance. In addition, benefits can also be qualitative, such as improved employee morale or a more recognized corporate brand.

10.1 Qualitative Benefits

- Rapidly implement best practices through lift and shift from across the different squads
- Build reusable framework components once and deploy across multiple squads
- Increase collaboration among team members and share knowledge
- Develop agile team members into T-skilled professionals where the vertical line of the “T” indicates the specialization that they gain and nurture through chapters and the horizontal line indicates the more generic skills they develop within agile teams.

10.2 Quantitative Benefits

- Increase in velocity by implementing automation of the existing manual testing process
- Significant decrease in R&D effort within the squads as this effort is now moved to the chapters
- Optimize tool license cost by optimizing the usage of tools within the teams

Conclusion

In conclusion, chapters play a crucial role in agile organizations, contributing to the development of individuals, teams, and the overall success of projects. They provide a structured framework for career growth, knowledge sharing, and collaboration within the organization. By grouping individuals with similar skill sets or interests together, chapters promote learning, skill development, and mentorship opportunities.

Chapters foster a culture of continuous improvement by encouraging regular feedback, promoting transparency, and creating a safe space for open discussions. They enable individuals to explore new areas of expertise, expand their skill sets, and take on challenging projects. Additionally, chapters serve as a valuable mechanism for identifying and addressing skill gaps, ensuring that the organization has the necessary capabilities to deliver high-quality products or services.

Furthermore, chapters enhance communication and collaboration between teams, breaking down silos and encourage cross-functional interactions. Through regular chapter meetings, workshops, and knowledge-sharing sessions, teams can leverage the collective intelligence and experience of their members, resulting in increased innovation, productivity, and problem-solving abilities.

However, it is important to note that the success of chapters relies on effective implementation and management. Organizations must establish clear guidelines, responsibilities, and goals for each chapter, and ensure that there is adequate support and resources available to facilitate their activities.

In summary, chapters are a powerful mechanism for fostering collaboration, continuous learning, and career development within agile organizations. By creating a supportive environment that encourages knowledge sharing, skill development, and cross-functional collaboration, chapters contribute to the growth and success of both individuals and the organization as a whole in an ever-evolving and dynamic business landscape.

References

- <https://rebelsguidetopm.com/5-team-structures-for-agile-teams/>
- https://en.wikipedia.org/wiki/Tuckman%27s_stages_of_group_development
- <https://www.adaptovate.com/agile/what-are-chapters-in-an-agile-operating-model/>