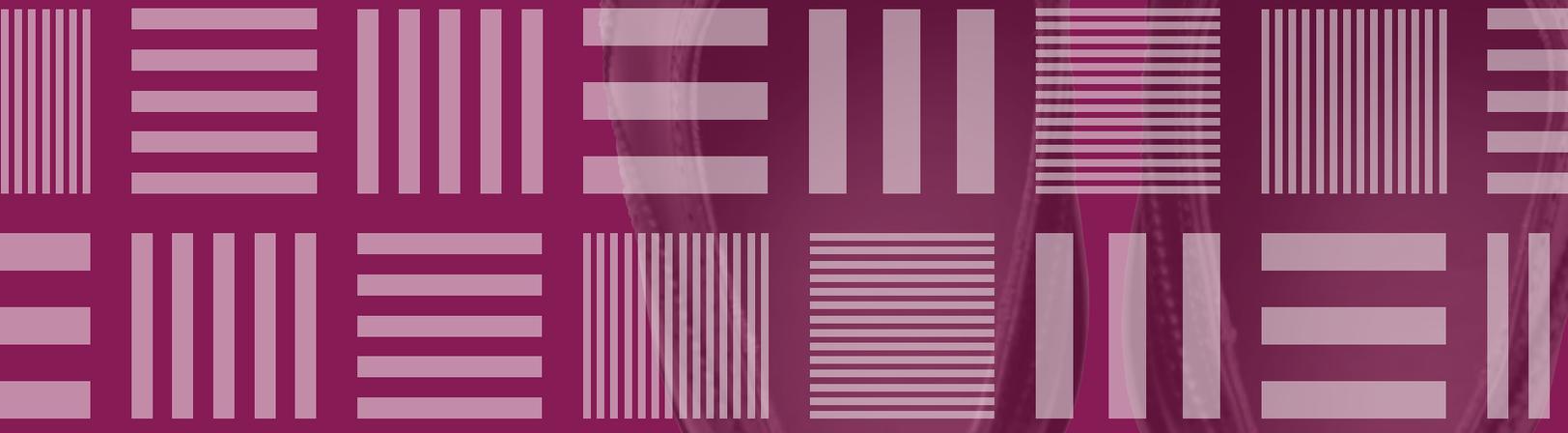


nisum[®]

consulting ≡ digital ||| innovation

Agile Models for Global Teams



Agile Models for Global Teams

Introduction

This white paper is designed to share our knowledge as a provider of distributed Agile software solutions and to help companies get started with distributed Agile, a software development methodology (SDLC) that revolutionizes software release. With traditional “waterfall” methodology a product is released all at once when the product is completed. In contrast, teams using Agile “time box” software releases, deliver the most important features first in short, regular intervals.

In 2001, at a summit of seventeen thought leaders of several programming methodologies, a consensus was reached around four main values captured in the Agile Manifesto. These values serve as the vision for the Agile SDLC. The first value states, “Individuals and interactions over processes and tools”. In the earlier years of Agile

adoption, there was a common belief that this required teams to reside in physical proximity of each other. This belief hindered many companies from adopting Agile, finding it not feasible with employees distributed in more than one location and time zone. In this paper, we will describe some ways in which Agile methodology has been successfully used to improve the productivity of teams with globally distributed members. In our preliminary research, we reviewed existing literature and conducted an online survey to identify ways in which teams (especially teams that are distributed) have implemented Agile. We asked: Can it be done? What obstacles have teams faced? How have they overcome these obstacles? In compiling the answers we identified some of the challenges and best practices teams use to implement this “distributed Agile” and the most commonly used models.

Evaluating the results of our research, we found that distributing team members could likely bring participating companies significant advantages—benefits that were not foreseen when the original Agile Manifesto was written. These include the ability to recruit talent from a larger resource pool, increased flexibility to meet the needs of a changing market, and improved team productivity through taking advantage of global time differences. In some cases, distributed Agile can provide a growing business with the edge that will allow it to be more versatile than its competitors.

When using Agile with a distributed team, it’s crucial to select the model best suited for a company’s size, structure and type. Companies also need to review their growth plan carefully and consider how to mature their distributed Agile model as their business grows and their needs change. In this white paper, we present a selection of some of the most commonly used models of distributed Agile methodology. Our goal is to give you a better understanding of not only how to begin using it, but also how to create a vision of team development that will improve your business competitiveness.

Challenges

The most common challenge faced by teams using distributed Agile is finding an efficient way to communicate. We found that this was often due to differences in both time zones and working hours across the globe, as well as cultural and linguistic barriers.

Team members also complained that they spent an average of 35% more time dealing with collaboration issues than they did focusing on the projects at hand.

Workers also dealt with a lack of infrastructure. Anyone on a dispersed team has at some point experienced the frustration of wasting a good part of a meeting trying to establish a stable connection.

The size of the distributed team was another major concern. As teams grow, it becomes more difficult for all voices to be heard. Inclusion is an important component of the consensus making process, which is imperative to the agile frameworks.

A lack of clarity in measuring innovation and quality was another common challenge. This can reduce managers' confidence that a distributed team will be able to deliver a quality product while following a timeline that will provide a competitive market edge.

Using Agile in a Distributed Model

Nisum has developed a set of distributed Agile engagement models to help you develop your teams and improve your outcomes—we'll review a few of them here. While they do not represent every team engagement structure available to you through Agile, they do provide sketches of some classic structures to show you how distributed Agile takes shape and drives success for a busy organization.

The Agile engagement models described here are three-dimensional:

1. **Human Connection:** refers to the people along with their physical location and roles.
2. **Collaboration Approach:** refers to how people work together to deliver software products.
3. **Apparatus:** refers to what tools people employ to get the work done.

Here we present five Agile engagement models in order of complexity: the most simple to the most complex ways that teams initiate, implement and manage their work. The model's degree of complexity is called its "maturity level," in reference to the way the company has managed its human connection, collaboration approach and apparatus to implement Agile methodology. At its most mature, Agile is fully automated and takes little or no time to deploy and scale.

As we have listed and described the models from simple to complex, the first will be easiest to implement and may be the best choice for a company that is just starting to utilize offshore resources. It might also be useful for a company that already has an offshore team and is thinking about changing its SDLC, moving from a waterfall approach to an Agile approach.

As company teams become more experienced at utilizing and implementing distributed Agile, they can introduce more advanced techniques from the more complex or mature engagement models, such as continuous delivery and resource rotation.

Elements from the models can be combined depending on the team's functional needs and/or the company's business requirements.

Hand-Off Model

This is a great choice for a company that wants to begin using Agile; it can be a stepping-stone to further engagement and to the more complex models described below.

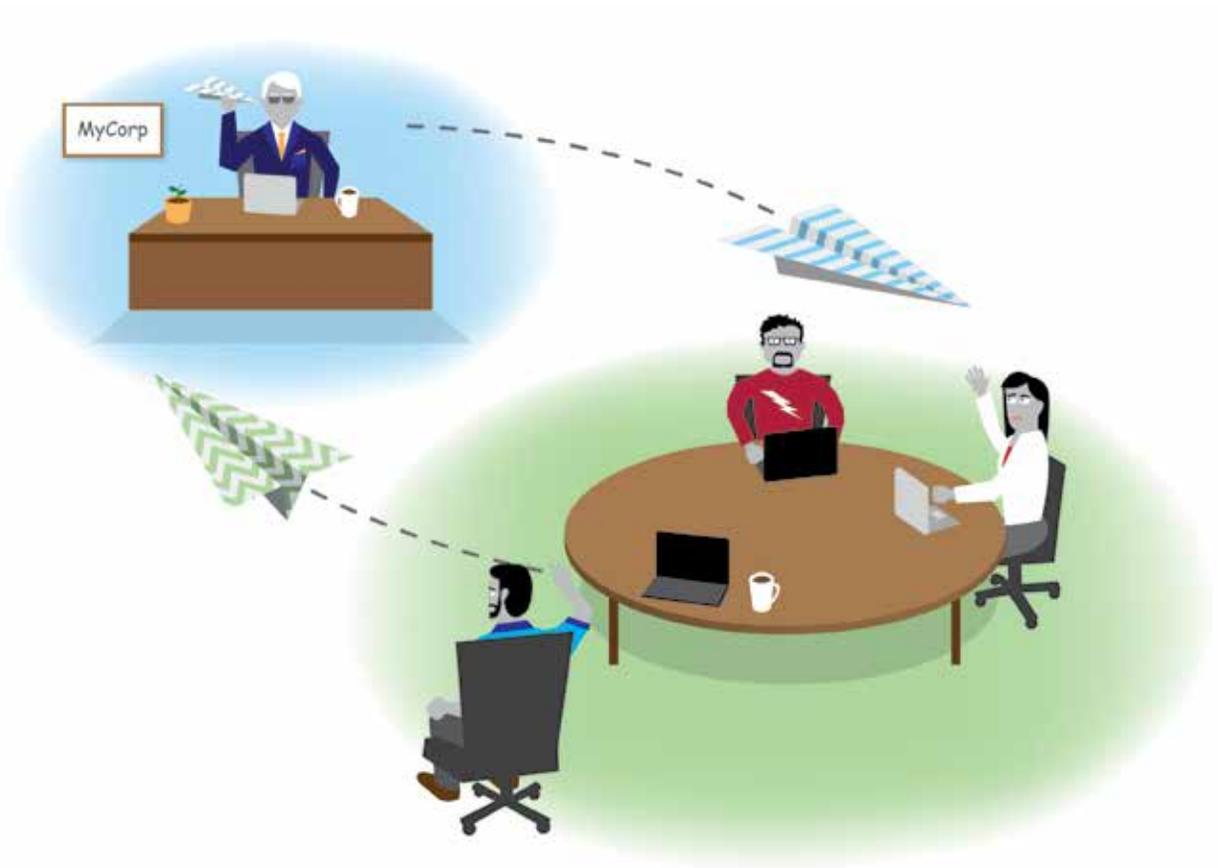
Human Connection

The product owner is onsite, while the rest of the team is located offshore in one remote office: including the Scrum Master, the developer and the quality assurance team.

Collaboration Approach

The product owner works closely with product development staff to create stories and relays this information to the offshore team. Because the product owner plays a key role in this model, it is essential that he or she works closely with the team. Without the product owner's participation in all Agile ceremonies (especially the daily standups), the model can easily revert to a waterfall SDLC.

The Scrum Master is responsible for ensuring that the product owner understands the Agile process, and clarifies the team's expectations regarding story writing and backlog prioritization.



Apparatus

A backlog management tool is important for this model of distributed Agile. Stories need to be clearly written, with acceptance criteria that the team understands. As this model is most suited for situations where the entire development team is co-located, test automation and continuous release cycle tools are less critical.

Advantages

- A quick return on investment for short-term, well documented projects.
- Team co-location reduces the need to work during non-business hours.
- Locating the Scrum Master with the team improves communication and eases the workload as it allows for frequent check-ins.
- Provides an introduction to Agile methods.
- Low organizational impact as no new roles are required.

Challenges

- Maintaining infrastructure for daily communication between the product owner and development team can be a challenge. The offshore team must have access to release management and infrastructure so they do not lose work time in situations such as waiting for a server to be restarted.
- Setting expectations with the product owner if the product owner does not have time allotted to work with the team can be a frustrating experience for the team. A product owner who is too busy with other work demands can become a single point of failure, resulting in a disconnected team and a process that reverts to waterfall methodology.

Best Practices

- Make the product owner feel like a part of the team. We highly recommend flying him or her out to meet the team on a regular basis.
- Establish clear and concise communication between the product owner and the team. The Scrum Master is a technical resource and serves as a liaison between the product owner and the team.
- Develop good stories and clear acceptance criteria. They are essential for the success of this model and must be produced by the product owner. The Scrum Master should work as a coach for the product owner (in developing and splitting stories) and the team (in reviewing and sizing them).
- In order to ensure the success of this model, practice shorter sprints with demonstrations of functionality completed to date.

Follow the Sun

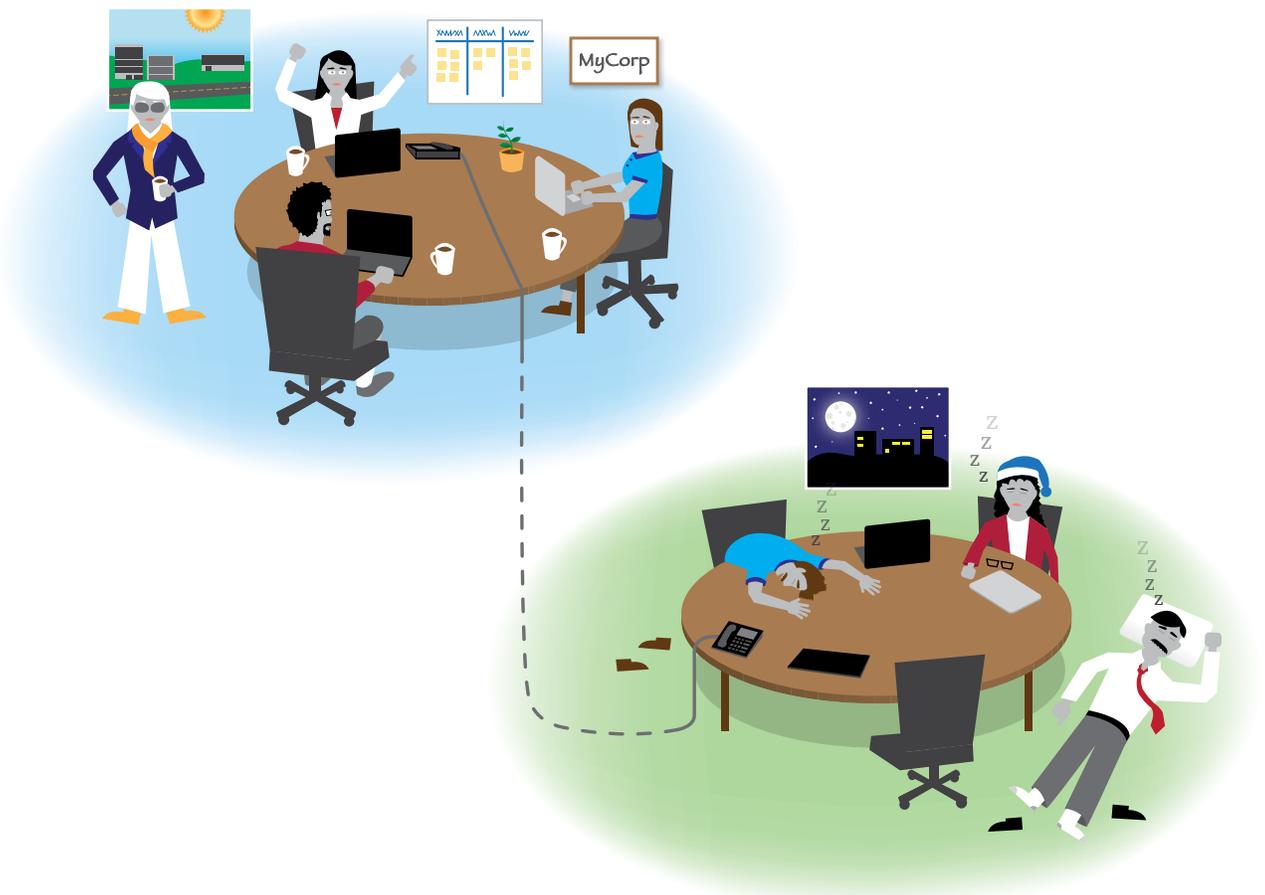
The name “follow the sun” reflects the fact that in this model team members work during daytime hours—whatever those happen to be, based on their particular time zones—and that the model is designed to take advantage of the work time differences of onsite and offshore team members.

Human Connection

The product owner is co-located onsite with the Scrum Master and product development team, while the quality assurance team is located offshore in a different time zone. In this approach it is common that quality assurance is completed during onsite non-business hours. Taking this model to the next level, team members can be distributed in more than two time zones, for instance in North America, India and Europe.

Collaboration Approach

This model takes the company and teams into a more mature relationship between service provider and client. It is based on a waterfall SDLC that was primarily used early in the offshore development trend and was aimed at cutting costs and decreasing time to market. The model allows the development team to leverage the time difference between locations to increase productivity.



For this model to succeed, communication between the offshore team and the onsite developers is key: daily stand-ups and other team meetings must be held during the time overlap window. The core task definition must be kept in mind, while development and quality assurance happen in parallel.

Apparatus

At this stage, introducing testing automation tools is essential as teams expand into more globally dispersed offices. An online tool for backlog management and tracking stories becomes critical along with the ability to share information online, such as in a wiki.

Advantages

- Faster delivery than a hand-off or an onsite model, when well executed.
- Potential for lower cost of delivery, with offshore staffing dispersed in areas with a lower cost of living.
- Easier communication as developers, Scrum Master and product owner are co-located, which facilitates ad-hoc huddles with developers and product owners.

Challenges

- A short, or nonexistent, time overlap window can lead to problems. In the absence of an overlap (such as between the United States and India) team members may have to work late at night or early in the morning, which can make it hard to recruit and retain staff.
- The short overlap window imposes scheduling constraints for daily stand-ups, retrospectives and other team meetings. Any difficulty in communication can leave teams without critical information and result in the loss of a day's work.
- Team collaboration and creating a sense of unity can be more difficult to achieve when developers and testers are not co-located.
- The model may limit the type of testing that can be done. Testers may also find it harder to understand the stories and may feel left out of ad-hoc huddles between developers and product owners.
- The testing environment must be accessible to remote team members; this may require duplication in managing and maintaining physical and virtual environments.

Best Practices

- Bring offshore testers onsite to get to know team members and improve communication and team unity.
- Celebrate success with both onshore and offshore team members: for example team members can go out for a celebratory lunch and share photos of the event. Human Resource managers can work with the Scrum Master to acknowledge hard work and creative solutions.

- Pair developers and testers for check-ins during any overlap time.
- The Scrum Master may hold two stand-ups, one for each team, to ensure that information is completed by one team and ready for the next to pick up.

Functional

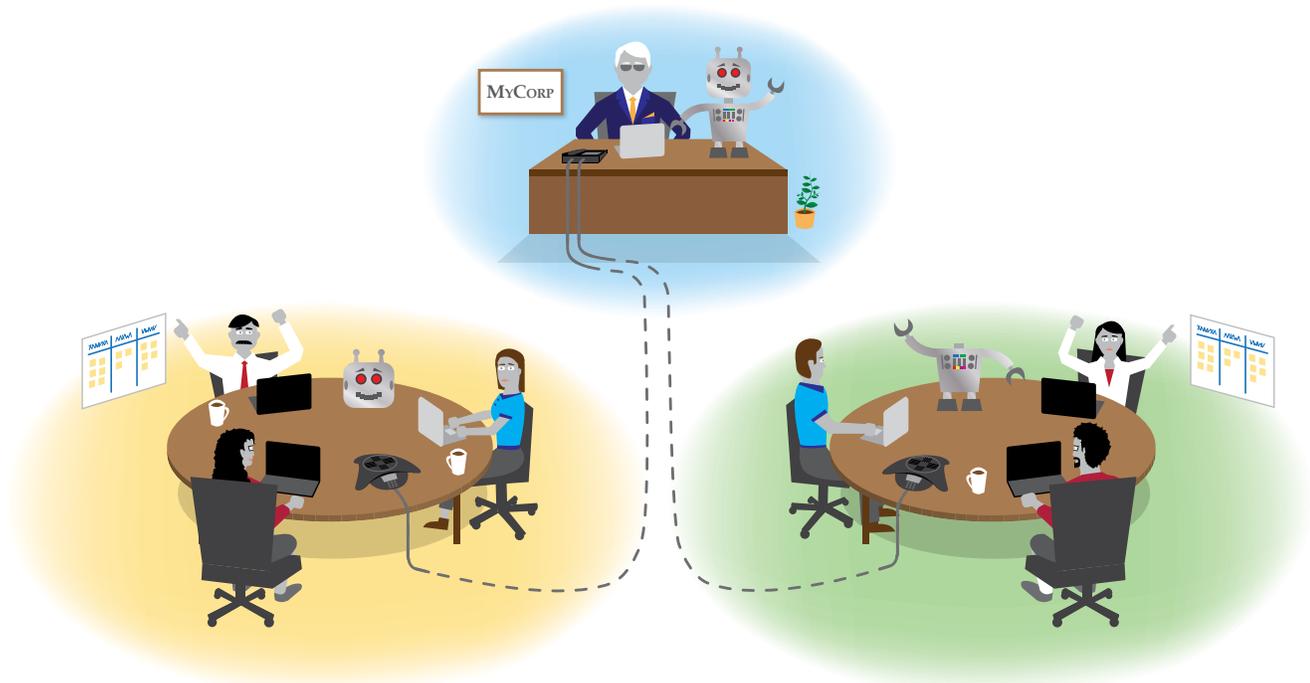
The Functional Model is a step-up in complexity and often incorporates a mix of distributed and onsite Agile teams. Work is distributed between teams according to functionality. This model balances the two underlying drivers of Agile Methodology: the needs of the project (such as profitability) and the needs of the team (such as communication between team members). The offshore team allows a company to expand into new areas quickly and efficiently, reacting rapidly to emerging markets in order to maintain a competitive edge.

Human Connection

This model has the entire Agile team in one location, except for the product owner. An example of this is with a Distributed Agile Support team or a Distributed Agile Innovation team.

Collaboration Approach

This model is best for companies that already have experience working with an offshore team. Service level agreements (SLAs) and communication methods are in place, and the company has experience working with the service provider through distributed methodologies.



One function we see often is the transitioning of code operational management to the offshore service providers' team (the Support function). A long-term team will manage changes, updates and patches to this code base. With a fixed team budget, the company can control costs through outsourcing.

Another option is piloting new technologies. We can utilize a service provider to test new technology, giving a company a competitive advantage. For example, if your software development team lacks experience with mobile development, you can utilize your offshore service provider to create mobile versions of your website (bypassing the need to train internal staff). SLAs should then be put into place with the service provider to update the site to support new devices as they are released. In this example, the Scrum Master may work as a team representative and stay after hours or come in early to ensure the team has everything it needs to function efficiently.

At this point it is important to look deeper into how testing automation is preformed. BDD, behavior driven development revolutionizes the testing processes as the engineering team focuses on customer driven behaviors and functions first. These processes will require a mindshift that goes hand in hand with the organizational transformations that distributed teams will have naturally occur with Agile adoption.

Apparatus

The transition to behavioral driven development can require new tools that empower engineers to think about the design before they code. This is a good point in a company's growth period to evaluate the development and collaboration tools utilized to see what may be available to assist in the success of your Distributed Agile transformation. Incorporate tools that allow engineers to easily document design and development as part of the implementation process.

Advantages

- With the right service provider, new ideas can be quickly piloted and implemented without hiring or training new staff. The service provider can set up client best practices for development and expand and contract teams as needed.
- Freeing up resources from maintaining legacy code can reduce costs, while the service provider can look for creative methods to refactor and optimize this code with rewards built into SLAs.

Challenges

- As with the hand-off model, this can easily revert to a waterfall approach if the product owner is not fully engaged with the offshore team.
- Knowledge transfer sessions can be costly and are often an inefficient way to get a new team ready to change an existing code base.

Best Practices

- We recommend starting off with small projects with your service provider, and establishing distributed Agile meetings to clarify milestones, lengths of sprints, and communication methods. Once you've established functioning teams and developed a "well oiled machine," you'll be ready and able to take on new opportunities as they emerge.
- Utilize technologies within a shared (cloud) infrastructure.
- Prepare clear criteria on what functionality is created onshore and offshore.

Onsite Coordination

The functional model we just described is a strong method for building and testing a company's relationship with a service provider. Once you're sure that the provider can meet SLAs and cost saving goals, you can consider a longer-term model to leverage this relationship and increase communication capabilities between onshore and off-shore team members. The Onsite Coordination Model is ideal for a company that has reached this level of maturity.

Human Connection

In this model, one or more technical coordinators are located onsite with the product owner and the Scrum Master, while the rest of the team is offshore.

Collaboration Approach

The product owner aligns the team with business priorities and values. The coordinators work closely with the product owner and the team to relay the priorities and clarifications the team may have. It is important that the stories are well written and have clearly defined acceptance criteria.

Test-driven development, or TDD, is another core component of this model. It was developed from basic Agile principles, themselves focused on the art of simplicity and self-organizing design. In TDD, a test case is written from the acceptance criteria as soon as the story is picked up to play from the backlog. The first time the team runs the test case it will fail, since code has yet to be written. Still, this process validates the test case clarity and allows the team to review expectations together.

Pairing engineers, a concept introduced by extreme programming principles, can significantly reduce knowledge transfer overhead while increasing team motivation. At this point it makes sense to look at the guiding principles from Agile based frameworks and begin incorporating them to increase the model's success.

It is very important here to ensure that teams feel safe and encourage open communication. In doing so, mistakes are exposed early on and become lessons

promoting innovation and reducing the fear that can be exaggerated by cultural and geographical disparities.

Apparatus

There are many types of tools on the market now that can be incorporated to assist dispersed teams at this stage. Tools can be beneficial for Agile practices such as pair coding and refactoring, release planning, connecting stories to the test and code base and TDD.

Advantages

- Onsite coordinators work directly with the product manager for quick resolution on queries from the team.
- Onsite coordinators can help resolve infrastructure and release issues as they arise.
- Onsite coordinators can represent the offshore team and convey information in story huddles.



Challenges

- The offshore team may feel left out. It's up to the HR manager and the Scrum Master to keep offshore team morale high and to celebrate team success.
- The coordinator's role must be clearly defined, so that a hierarchy is not formed that could inadvertently bypass coaching principles.

Best Practices

- Ensure clear communication and set up processes to avoid creating a hierarchy. For example, coordinators can be offshore team members who visit the client site on rotation.
- Maintain a workstation for visiting team members and remember that they will need housing and other resources.

Agile Next Door[®]

Agile Next Door[®] is a proprietary concept developed by Nisum Technologies for identifying the best practices from Agile models currently in use (and described above), and for creating a system that takes full advantage of each team member's geographical distribution, along with more mature Agile processes.

Agile Next Door[®] is recommended for companies that are using an advanced or mature form of distributed Agile, with teams challenging themselves to reduce sprint cycles to two weeks or less. This increases the team's ability to think up, prioritize and release features to the customer in a timely manner, and creates a competitive edge for the business.

The Human Connection

In this model, all teams and team members can be easily accessed, regardless of their locations. Using technology like video streaming, team members can easily collaborate. They get to know each other personally and professionally, through rotation, and they enjoy the multi-cultural aspect of a global team.

Collaboration Approach

The ability to locate team members in multiple time zones is a key element of Agile Next Door[®]. Engineer pairs can be scattered in around the globe while the team's clear processes blend daytime and nighttime into seamless 24-hour sessions. Rotating between sites and pairs, team members continually learn from each other, becoming more competent for each project as they go.

Apparatus

This model relies the heaviest on technology and connectivity. There are a significant number of products aimed at facilitating distributed team collaboration. A factor for

this model's success is the reliability of these products. It is recommended that the tool set chosen be consistent for all locations. Communication suites (especially those that facilitate screen sharing, VOIP and instant messaging) allow team members to work together as if they were sitting next to each other and sharing a screen, regardless of their physical location.

Advantages

- Distribution of team members through multiple time zones, as described in the Follow the Sun model, can increase a team's productivity while potentially decreasing costs.
- A geographically distributed team allows for increased availability and response times.
- Agile Next Door® allows for the global recruitment of talent, enabling a much larger resource pool.
- Cultural diversity of team members can create motivation and innovation within a team, as they integrate diverse ways of thinking.



Challenges

- Expect a longer ramp-up time for new teams, as cultural and language differences require more time for team members to learn to communicate effectively with each other.
- High speed, reliable connectivity to the virtual team space is a requirement.
- Business hours may need to be adjusted in order to increase the overlap window between team members.

Best Practices

- Online dashboards for collecting metrics are a key factor in implementing Agile Next Door[®]. Neighborhood Metrics, which measure a team's progress at facilitating continuous improvement, are important components of these data sets.
- Repeatable processes and supporting infrastructures allow teams to form good work habits. The stability and consistency (of doing the same thing in the same place at the same time) allows the team to measure its velocity.
- Moving away from the concepts of "day" and "night," Agile Next Door[®] introduces the concept of "Neighborhood Meets and Greets." These are important work sessions that can happen at any time, as determined by the team. A Neighborhood Greet is time-boxed to 15-minute sessions. The first 15 minutes resemble a daily stand up: team members greet each other and exchange news. They tell each other what they've completed since the last Neighborhood Greet, and what they plan on doing until the next one. Any items that require deeper discussion are paired for further discussion immediately afterward in the next 15-minute block, the Neighborhood Meet. These sessions take advantage of limited overlap time to ensure the entire team, or "neighborhood," understands what each other team will focus on until the next session. These sessions are key for the success of this model. The neighborhood metaphor has been a successful tool allowing us to move away from the division of an Agile team into clusters of offshore or onsite members. Instead, all are part of one "local" neighborhood and all team members are just next door.
- In this model, team processes and tools maximize automation. Ensure all team members are comfortable with the selected tools. That will reduce the likelihood of corner cutting during time crunches and will ultimately increase your success.

Distributed Agile Best Practices

Nisum Technologies has helped numerous companies implement the Agile models that best meet their needs. From our experiences, we've put together some recommendations for best practices to help you along your own distributed Agile journey.

Before you start that journey, you and your teams should discuss a number of important agreements to establish a clear understanding of Agile, set expectations, and educate team members on the process. We recommend that an Agile Coach work with you to put these practices in place, together with the service provider and product owner, so as to best reflect the company's needs and overall methodology.

Team Size

Most Agile teams are comprised of six to eight team members—they are similar in size to an average family. A team with more than ten members can become hierarchical and can smother the voices of softer-spoken members. A consistent set of team members enables the team to improve incrementally. Adding or removing a team member always changes the team's overall velocity. Design patterns recommended by the Extreme Programming framework, such as pairing, are best implemented with an even number of team members. Pairing encourages developers to work side-by-side with a counterpart while teaching each other skills; this doubles the team's size (and respective costs), but also increases productivity and improves morale. In our experience, teams organized around two to four pairs showed the most consistent overall trend in product delivery.

Team Composition

The stability of an Agile team directly contributes to their ability to "be agile." Relationships based on trust take time to develop. Agile ceremonies, processes, and interpersonal interactions also take time to develop and are essential for building team cohesion. Companies share members across teams all too often—breaking a distributed Agile team in this way can impair its autonomy, reduce team cohesion, disrupt its rhythm and depress its productivity (as measured by its velocity). It is important that the team is composed of members who feel empowered to make decisions and do not need a "manager" to bless the decisions around technology or estimates.

When building an Agile team, it is important to differentiate between roles and members. One team member can play more than one role on the team. For example, hiring and/or training team members to design and write both functional code and testing automation removes the bottleneck that occurred from the hand-off between developers and QA that many teams suffer from. This flexibility allows people to focus on what needs to be done, to "gather around the work," rather than sheltering behind a role.

In using the Scrum framework, most prevalent in the industry at the moment, a distributed team's Scrum Master is the team's Agile advocate. This role embodies the concepts, motivation and energy of "being agile." Teams members' approach to meeting challenges is vitally important. An effective Scrum Master makes sure the

team has what it needs and is not delayed by waiting for blockers to be resolved in another time zone. The Scrum Master keeps a record of team decisions and Agile ceremonies and ignites the team to continuously improve. He or she meets cultural and communication difficulties with more timely team check-ins and ensures that all voices have been heard.

The Scrum Master does not play a part in Human Resource Management. Both onsite and offshore team members need to have separate human resources managers who communicate regularly with the Scrum Master.

Team Member Rotation

Offshore team members can be regularly rotated to join the onshore team to improve communication and cohesion, and build a more motivated and collaborative environment. Rotation also facilitates a natural “leveling off” of knowledge: a team member coming onsite becomes the spokesperson for his or her entire team while visiting, and in doing so gains business and technical knowledge. Team members can be rotated in pairs or as individuals. In either case, those on rotation are given a specific plan and set of goals to meet. They can be encouraged to begin each work session by describing differences they see at the new site or expectations they have for the visit. Our experience and survey results show that team members brought on location become more engaged; they communicate more effectively, become more dedicated, and capable of making stronger decisions. The success of Agile depends on having teams that are both self-empowered and enabled to make collective decisions, resolving issues as they come up. Teams like these will be more productive, will produce higher quality products, and will contribute to increased satisfaction with the service provider (see survey results).

IT Infrastructure

Setting your team’s expectations and building the right technical infrastructure is critical to their success. A clear method for communication (determined, tested, configured, and established with your service provider) can cut down on frustration and costs, and will increase a team’s productivity and satisfaction.

Agile process recommends “interactions over documentation.” It is to your benefit to put the tools and processes in place for clear communication and collaboration at the beginning of this engagement.

Behavior, Design and Test Driven Development

Behavior, design and test driven development frameworks were developed to support Agile principles. These methods build quality into the development process; turning it upside down from the way testing is performed in waterfall methodologies by writing the test cases prior to the functional code. These methods are extremely helpful for

distributed teams as they increase quality and decrease time required for knowledge transfer. When selecting a service provider make sure they utilize these methodologies to ensure that code is more easily maintained and supported. The effort put into establishing these processes ahead of time can make a big difference to your company as it grows.

Conclusions

We began this white paper by presenting the hand-off model for Agile, which does not require the onboarding of new resources. For this reason, it can be adopted quickly by a smaller company. A company wishing to adopt this model can also start by utilizing a service provider's infrastructure and best development practices. This model is especially useful for companies that do not have their own software development department or would like to pilot a new technology they do not have in-house.

In contrast, the Agile Next Door® engagement model utilizes a more integrated approach to working with a service provider. Team members periodically visit their counterparts at the client's location, making this model best for companies that have long-term, ongoing relations with a service provider.

We recommend that companies interested in using Agile methodology start with a low-level maturity engagement model (such as the hand-off model) to establish and test a working relationship with a service provider. Once those processes are in place, and relationships and trust have been created, then the company is better positioned to integrate the service provider into its overall corporate strategy for software development.

More Information

For additional information or free assessment, please contact pmo@nisum.com.

About Nisum

Nisum enables transformation for industry-leading brands. We know how to build strong emotional bonds between B2C clients and your customers via smart technology solutions. Nisum is a global consulting firm headquartered in Southern California. Founded in 2000 with the customer-centric motto of "*Building Success Together*®," we've grown to over 900 consultants and 8 offices across the United States, India, and Chile. Our philosophy and deep technical expertise result in integrated solutions that deliver real and measurable growth.

Whether you're a hot start-up, or a major global brand, our approach is the same: forge the most powerful connection possible between people, processes, and products in order to achieve unparalleled success. At the intersection of business and technology, Nisum has everything you need to grow your organization. From Strategic IT Planning, Agile Enablement and Business Process Engineering to Application Development, Test Automation and DevOps, Nisum has you covered. We specialize in building Adaptable Back-End systems such as Order Management, Inventory and eCommerce. Thereby, facilitating true omni-channel success for our customers.

Representative Clients

minted.

★ macy's



Walmart



BANANA REPUBLIC
BR

Guthy | Renker.

WILLIAMS-SONOMA

bloomingdale's



Boulder Logic

Canon



TOYOTA



Alliance Partners



ORACLE



Canon

Microsoft

Disclaimer: Nisum Technologies, Inc.'s white paper is made available for educational purposes only, as well as to give you general information and general understanding regarding Agile and Agile Next Door Models. The information herein is not advice and does not create any obligations, relationships or duties on the part of Nisum Technologies, Inc. This white paper is provided "as is" and with no guarantees, representations, undertakings or warranties whatsoever, including any warranty of merchantability, fitness for any particular purpose, or any warranty otherwise arising out of any proposal, specification or sample.

The contents of this white paper, the names Nisum Technologies, Inc. and Agile Next Door, the logos and artwork used herein, are protected by the copyright, trademark and intellectual property laws of the United States and other jurisdictions. No license, express or implied, to any intellectual property rights is granted or intended hereby. You may print a copy of any part of this white paper for your own use and reference, but you may not copy any part of this white paper for any other purpose, and you may not modify any part of this white paper. You may include any part of the content of this white paper in another work, whether printed or electronic, or other form, or include any part hereof in another web site by linking, framing, or otherwise only if you provide all proper credits and references to Nisum Technologies, Inc.

Building Success Together®

≡ CORPORATE HEADQUARTERS

500 S. KRAEMER BLVD
SUITE 301
BREA, CA 92821
714-579-7979

≡ CHILE

AV. APOQUINDO PISO #4700
PISO 14
LAS CONDES, SANTIAGO, CHILE
562-220-796-98

≡ INDIA

PLOT NO. 107
LUMBINI ENCLAVE, EURO SCHOOL ROAD,
GACHIBOWLI, HYDERABAD-500 032, INDIA
+91-40-65446522

≡ PAKISTAN

SUITE #3 GROUND FLOOR, PLOT NO. 25-C,
4TH SUNSET COMMERCIAL STREET,
PHASE IV D.H.A KARACHI, PAKISTAN
+92-21-35389700

