OpenAgile Team Member Manual

Version 1.0

This document is constantly evolving.

To see the latest revision, go to

wiki.openagile.org/wiki/Team_Member_Training_Manual_

This page intentionally left blank.

Table of Contents

Team Members are the individuals who are the contact-point for the OpenAgile processes. If OpenAgile does not work for these on-the-ground people, then it does not work, period. Thus, for the Team Member, OpenAgile must be practical. It must provide more benefit than

Introduction

This manual is designed to help guide you and your teammates in developing the basic capabilities—qualities, skills, habits and attitudes—of an OpenAgile Team Member. Towards this objective, we will examine more closely the practical applications and implications of most of the basic concepts outlined in the OpenAgile Primer.

The OpenAgile Team Member is the second level of OpenAgile Capacity-Building. The requirements for certification of this capacity, that of a Team Member, are as follows:

- OpenAgile Readiness Certificate (first level of certification)
- Reading this manual
- Passing an online test based on this manual
- Endorsement from 3 people, one of whom is an agile coach or OpenAgile Mentor.
- Signed agreement to code of ethics, applicable for lifetime
- Participation in a 2-day OpenAgile Team Member Training seminar

Further details about the Capacity Building program and the Team Member level are available online at http://www.openagile.com/.

The background for this manual includes the combined experience of many consultants, coaches and trainers working for many years with many teams from various industries and disciplines. Some of the resources that have contributed a great deal to the learning behind OpenAgile and referenced in this manual include "The Wisdom of Teams" by Jon R. Katzenbach and Douglas K. Smith, "An Introduction to General Systems Thinking" by Gerald M. Weinberg, "Beyond the Culture of Contest" by Michael Karlberg, "Agile Software Development" by Alister Cockburn, "Toyota Talent" by Liker & Meier, as well as some of

the curriculum of the Ruhi Institute and FUNDAEC (Foundation para la Aplicación y Enseña de las Ciencias).

This manual is designed to function more as a living document rather than as a definitive technical recipe of best practices. This living document approach allows the learning of those who engage with its contents to interact—at least to the extent possible given obvious constraints—with the learning of those who have developed it. OpenAgile is an evolving conceptual framework for doing work. Those who strive to rigorously implement it as a framework and learn to adopt its principles, concepts and practices into their own work will—through their own insights—invariably generate new knowledge that will contribute to the further evolution of OpenAgile as a methodology.

Lastly, a word about your learning: the OpenAgile Institute assumes that all individuals who engage in learning with us have taken full responsibility for their own learning and are willing to tolerate certain ambiguities inherent in a system of learning. We consciously avoid presenting OpenAgile as having all the solutions to all possible problems. Therefore, patience with ourselves as learners and with the learning process in general will ensure that we are advancing in our understanding and capabilities associated with the core concepts and foundational principles of OpenAgile.

We at the OpenAgile Institute look forward to learning with you!

Part One – Deeper Foundations

Why go deeper? Understanding various aspects of the OpenAgile foundations—Truthfulness, Consultative Decision-Making and Systematic Learning—will empower the individuals and their teams. As a team appreciates these foundations, through discovery and practice, the further it will advance towards delivering work of value to the organization it belongs.

Our normal concept of the foundation metaphor is that of the foundation of a house. This concept is sufficient for basic understanding. A richer metaphor is that of the root system of a tree. This is also a foundation, but it is a foundation for a living system rather than a static edifice. A conception of the foundations of OpenAgile that reflects this living system metaphor helps us understand that Truthfulness, Consultative Decision-Making and Systematic Learning do not have to be perfect and complete before we start to use OpenAgile. Instead, like the roots of a tree, these foundations can grow organically to support a growing use of OpenAgile's processes and practices. The growth of the foundations needs to keep pace with the growth of the other aspects of OpenAgile, but which foundations grow quicker, which grow deeper or broader depends, in part, on the growth of the tree above.

As you review and learn more about each of the foundations in the following chapters. Consider these questions:

1. How can I improve my understanding of these foundations beyond this manual?
2. How can I apply these foundations to my work, both day-to-day and over a longer period of time?
3. As I apply these foundations, what are the broader implications to my team, my organization and beyond?

Chapter One – Truthfulness

> *"A commitment to truth creates a moral imperative that forces you to acknowledge the data and to take the important first step of recognizing reality."*
>
> - M. K. Gandhi

As we learned in the OpenAgile Primer, Truthfulness is one of the essential foundations of the OpenAgile system. Truthfulness has a number of components. In this manual, we will explore Truthfulness to understand it at a deeper level.

We will look at Truthfulness by examining four aspects or components of Truthfulness: visibility and transparency, basic honesty, integrity and self-knowledge. Each component builds on the others. The components are complimentary.

Visibility and Transparency

The easiest aspect of Truthfulness to understand and act upon is visibility. Visibility is the attribute of our work that allows outsiders to see clearly and without barriers what we are doing. This aspect is also sometimes referred to as transparency.

As a Team Member, there are two levels of visibility that you need to keep in mind. The first level is the ability of your Team Members to see your work. The second level is the ability of people outside the team to see the team's work. Let us look briefly at both of these levels.

Visibility of Your Work to the Team

Your Team Members need to be able to see what you have done, what you are currently doing and what you intend to do. This visibility is necessary so that they avoid duplicating effort, learn from your efforts, and avoid doing things that will cause you problems in the future. Of

course, as a Team Member yourself, you need visibility into their work as well. Visibility within the team is reciprocal.

You can use the team's task board to improve the visibility of your work by, for example, marking your initials on tasks you are working on, making task cards even for smaller tasks, and announcing your work verbally when you go up to the task board. With electronic tools, visibility is more difficult. You can share documents, even those in progress, with your team through various sharing systems (e.g. a shared network drive, Dropbox, a content management system, or a version control system). Visibility with electronic tools means that your tools should not restrict Team Members from looking at each other's work.

A less obvious example of visibility is also worth examining. When you are working with a team, sometimes you will have the need to either not work or remove yourself from the team environment. This could be to get a mental break or possibly if you are feeling unwell. When you do this, whenever possible inform your Team Members that you are no longer "present" and able to do work (even if you are still physically in the team environment). Also if possible, give your Team Members a timeframe within which you intend to re-engage with the team.

Visibility of the Team's Work to Stakeholders

The team ensures that its work is visible to its stakeholders. Like at the individual level, this is often done through the effective use of a task board and other similar "information radiators" (see "Visible Cycle Plan" below for more details). Unlike the individual level, visibility between the team and its stakeholders is not necessarily reciprocal or symmetrical – your stakeholders may use different methods to make different things visible to your team.

Basic Honesty

Honesty is the practice of telling the truth as you know it. Honesty is

about how we speak - it is a speech behaviour. It's not just an attitude. We are honest when we respond to the best of our abilities to questions that others ask of us. Telling a deliberate lie is not honesty. But what about answering a question incorrectly? Is it a lie when you make a mistake? Our underlying motivations are an important part of honesty. When we sincerely believe what we are saying to be true and it turns out we are incorrect then it is simply an honest mistake, not a lie.

Imagine you are on a team with five other people. Your manager has given you all a clear goal, and as a team you are all excited about reaching this goal. After six months of working hard towards this goal your team is nearly done. One day, you are talking with a good friend and co-worker in another team. She tells you that the goal you have been working towards is not actually the real goal. She says that it's just been to keep you all busy so that upper management doesn't lay you off. She knows this because her boss confided in her. She wants you to leave the team you are on before the work is done, and go work with her on something “real”. How would you feel about learning this? At the very least you would be disappointed. If you haven't already, you might also lose trust in your management. You might even be inclined to start looking for a job at another organization, hoping for a little more sanity. This lack of honesty has harmed the organization, and it has harmed you – you've wasted six months of your life and are probably more cynical than you were before the phone call.

Everyone in a team, in an organization and in a community has a duty to be honest with each other. Lack of honesty ultimately destroys trust and leads to waste, bad feelings, and failure.

Strategies to use when you don't feel / want to be honest

There are a few strategies you may use that help when being honest is difficult. The first strategy is to be honest without delay, the second is to be honest with yourself, and the third is to encourage a no-blame

environment. These strategies are often effective in ensuring ongoing honesty amongst Team Members.

Being honest without delay makes things easier by enabling the team to deal with difficulties and issues quickly. If you delay being honest, then you run the risk of issues becoming more disruptive or conflicts amongst Team Members. If a mistake is made it is invariably more effective to be honest by telling the team as soon as the error is made or detected, rather than covering things up or be dishonest through not communicating important details. Admitting errors early allows the team greater opportunity to rectify the mistake before team efforts are wasted or additional rework is required.

Another strategy is to be honest with yourself. This is effective as it makes one more open to constructive criticism by acknowledging that one is not perfect. Additionally, poor performance or disruptive behaviour may be self-corrected if individuals are honest with themselves when reflecting on their own performance. It is also preferable to be honest with yourself regarding skills and competencies and ask for assistance where appropriate rather than producing poor quality work that will require team effort to rectify. Even difficulty in coping with stress is important to communicate honestly. Teams are often able to assist individuals with difficulties without negatively affecting overall performance by adjusting resources or schedules as needed. Without the knowledge of your struggle, your team cannot respond.

A no-blame environment is crucial to the promotion of team and individual honesty. Team members will tend to withhold information regarding errors to avoid discipline or punishment. Reinforce often at team gatherings or in team rooms that we all make the best possible decisions we can with the information that we have at the time. As time goes on, decisions may need to be revisited, but the no-blame environment allows the team to recognize the error as an unintended consequence and proceed to resolve it expeditiously rather than waste time determining precisely who is to blame.

Making a team commitment to honesty can be very challenging. The three strategies noted will assist in creating an environment where honesty is encouraged, supported, and demanded. Without an honest environment many teams will struggle to develop the capacity to self-organize effectively. This is one reason why Truthfulness is a foundation of OpenAgile.

Integrity

Integrity is the aspect of Truthfulness that is shown when a person, team or organization makes and keeps commitments. As an individual Team Member, your commitments are to your team as a whole. As a team or an organization, your commitments are to your stakeholders. Since the future is unknown, integrity sometimes requires great sacrifice.

Integrity for individual Team Members is tested every Progress Meeting. You will have told your team what you plan to do in a Work Period, and then in the next Progress Meeting you share what you actually accomplished. When there is a discrepancy, you know this is an opportunity to learn to improve your integrity. You might ask yourself questions such as “why did I not complete what I thought I would complete?” and “how can I ensure that my expectations match my results?”

The team also practices integrity. At the beginning of every cycle, the team commits to deliver valuable results from the value drivers by following a Cycle Plan. Ideally, the team should meet its commitment every Cycle. What if the team does not achieve the Cycle goals consistently? A team doing this lacks integrity. Their words and their deeds do not match up.

Without integrity, the team will quickly loose the trust of its stakeholders. When it comes time to finish the Engagement Meeting, each Team Member must reflect on making an accurate assessment of their own capacity, availability of time or any other factors that may hinder the

delivery.

In many organizations, integrity is one of the most challenging aspects of Truthfulness. Individuals are often asked to over-commit. Sometimes this is done as a management technique to put pressure on people. Sometimes this is part of a culture of unreasonable optimism. Pointing out this pattern of behaviour, this lack of integrity, can deeply challenge the thinking and attitudes of people to the point of causing great discomfort.

Visibility, honesty and integrity are all required to build trust. Building trust between Team Members is critical for effective execution of work and continuous learning. It is also critical to the creation of a true team spirit where the Team Members contribute to something greater than the sum of the parts. We will examine this concept later when we discuss the concepts of the team in Chapter Six.

Self-knowledge

Self-knowledge starts at an early stage of our development as humans. For example, a toddler, who has just learned to take a few steps, leaps out into a run, and falls. She discovers the limits of her strength and balance. Hopefully she will master the capacity of walking before running. This self-knowledge helps her avoid future falls.

Self-knowledge is an essential part of Truthfulness. A team's success in delivering quality work consistently depends on self-knowledge. Without self-knowledge the team will over-commit or attempt to do things beyond even its ability to learn. As with the other aspects of Truthfulness, self-knowledge occurs at both the individual level and the team level.

Measurement

One aspect of self-knowledge is the use of objective measurement to determine the strength or level of some quality important to the team's

functioning. For example, a team might decide to measure the average number of person-hours worked in a Cycle. This measurement could then be used for two purposes by the team. First, to determine the amount of work to do in a Cycle and secondly, to ensure that the OpenAgile meetings are taking a reasonable proportion of the team's time overall. Of course, measurement can become extremely sophisticated and used to understand subtle aspects of a team's capacity. In OpenAgile, the use of measurements or metrics is certainly allowed, but caution must be used: such metrics should not be used to cast blame on individuals nor should they be used to create serious competition among Team Members nor should they become so complex that the act of measurement takes a significant portion of the team's time.

Throughout the life of the team, there may be measurements that are long-term or permanent. These should be directly related to the team's overall goal. Any other measurements should be temporary. For example, an individual who wants to improve their speed at doing a certain activity might ask another Team Member to time them while doing the activity. This is repeated several times until a certain goal is reached. Once the individual has reached this performance goal, there is no longer a need to continue measuring that activity.

Action

Often the best way to gain self-knowledge is through action. Taking action gives you something upon which to reflect. The process of action and reflection is a subset of the Learning Circle which can be quite valuable on its own.

Let us say a young programmer, who is eager to know more, volunteers for a task at the beginning of a Cycle but is unable to deliver the work. This shortfall seems to be a result of a lack of skill. This missed step will reveal her programming deficiency. Painful though it may be, the programmer now has greater (and more visible) self-knowledge. During

reflection the team can praise her for being courageous, and then help her in acquiring the missing skills.

On the other hand, let us say she took the task without knowing the complexities, yet was able to accomplish the task. In doing so she was able to demonstrate unexpected capacities. The team may be surprised by her success and now the team has gained a greater level of self-knowledge.

Bias and Filter Awareness

In the OpenAgile Primer, we briefly recounted the story of the Six Blind Men and the Elephant. In this story, each blind man feels a different part of the elephant and as a result, they disagree about the nature of what they are collectively touching. Because they only have their sense of touch, and are only using it in an very constrained way, they are not able to "see" the whole picture – the elephant.

We all have biases and filters that we use consciously and unconsciously to understand our environment. Our senses, our mental models, even our habits of action all filter what we comprehend about our environment. In order to improve our self-knowledge, we can systematically look at our biases and filters. One method of doing this is to think about categories of people (e.g. "drunks", "foreigners" or "children") and categories of institutions (e.g. "banks", or "aid agencies") and decide if we trust someone in that category or not. Sometimes we will trust by default, sometimes we will dis-trust by default and sometimes, particularly for categories we know little about, we will not be able to say either way. This exercise then gives us a little perspective on our filters. We can ask ourselves why we trust a category? What positive experiences have we had in the past that let us trust? What would make us re-evaluate our trust? Likewise we can ask questions about dis-trust. Automatically trusting or dis-trusting is one of the most basic filters we have when interacting with other Team Members.

Summary

We have examined four aspects of Truthfulness: visibility and transparency, basic honesty, integrity and self-knowledge. Each aspect of Truthfulness applies both to individual Team Members as well as the team as a whole. Each of the four aspects of Truthfulness work together to create a growing foundation for OpenAgile.

Chapter Two – Consultative Decision-Making

"None of us are as smart as all of us."

– Japanese Proverb

One of the key advantages of working in a team is the diversity of viewpoints, skills, and knowledge that comes together. This diversity provides a team of individuals a vastly larger pool of ideas and experiences to draw from than an individual has access to. Consultative Decision-Making is an approach that leverages this diversity in a way that allows the OpenAgile team to arrive at unified action which in turn is required to produce exceptional results.

Consultative Decision-Making is a process for making decisions in a team where the views of all Team Members are actively sought and have an impact on all decisions made and where all Team Members learn to fully support those decisions in unified action. In Consultative Decision-Making no individual member of the team has independent authority to make decisions on behalf of the team unless the whole team has previously delegated that authority. Even then, this delegation must be considered temporary and it must be limited in scope. It would be against the spirit of Consultative Decision-Making for a team to permanently delegate all decisions to a single member of the team.

Consultative decision-making requires each individual to seek the truth in all things, to strive to transcend their respective points of view in order to function as members of a single entity with its own interests and goals. The atmosphere of Consultative Decision-Making should be characterized by both candour and courtesy. Every Team Member should always try to keep in mind that ideas belong to the team as a whole and not to the individual to whom they occur during the discussion. Since the team as a whole owns all of the ideas it can take up, discard, or revise them as seems to best serve the goal pursued. In order for consultation to

be considered successful, each individual Team Member must fully support the decisions arrived at, regardless of the individual opinions that they brought to the discussion in the first place.

In the OpenAgile process, Consultative Decision-Making is used intensively every Cycle during the Engagement Meeting, briefly during every Progress Meeting, and whenever two or more Team Members meet together to discuss a problem or issue. When meeting together, one Team Member may wish to explicitly mention that it is time to "consult" and summarize the concepts and behaviour of Consultative Decision-Making.

Behaviours

OpenAgile requires of the Team Member a certain mode of behaviour. This mode of behaviour is designed to ensure effective Consultative Decision-Making. Most of us already vary our behaviour to suit certain occasions. For example, the outbursts of fans shouting and booing are acceptable during a sporting event, but the same conduct would be out of place at a funeral or wedding. Much of our behaviour is determined by three factors: our personality, the culture and the current environment within which we act. OpenAgile teams also have a particular culture and environment that develops when the Team Members follow these guidelines for behaviour.

When bringing together talented individuals with different degrees of experience, capacity and knowledge, it can be a challenge working as a group when all of these are mixed with the various personal characteristics. Certain actions at a meeting can influence the outcome in both positive and negative ways. During a meeting, by paying attention to actions and reactions, a team can reduce the pain of working together. The following is a list of some good and bad behaviours.

Good Behaviours:

- Prepare mentally to have a positive attitude.

- Act with patience.
- Present ideas carefully to avoid repetition.
- Ask questions to understand the opinions of others and to avoid making incorrect assumptions.
- Offer ways to build on others' ideas.
- Never compromise the other OpenAgile foundations (Truthfulness and Systematic Learning) You are willing to firmly say “no” if you think an idea violates such a fundamental principle.
- Seek the kernel of truth that is in every idea (remember the blind men and the elephant).
- Encourage others by constantly asking them for their contributions.
- Offer your own ideas in a way that is designed to encourage the others on your team.
- Refute incorrect information or logic with great gentleness and courtesy.

Bad Behaviours to Avoid:

- Assert an idea repetitively.
- Demand attention either verbally or through your actions such as vigorous hand-waving.
- Interrupt either while other people are speaking or by changing the topic.
- Influence others to be on “your side” of the consultation through either implicit or explicit bribes or threats or other kinds of emotional manipulation.
- Make snide and sarcastic remarks about the things your Team Members are saying or doing.

- Take offence or fear rejection.
- Name call or other verbal abuse.

One of the best ways to improve your ability with the good behaviours is to practice them in role play.

Tools to Assist Consultative Decision-Making

While using Consultative Decision-Making, there are many tools Team Members and teams can use. Many of these tools come from great thinkers and practitioners who work with teams in many contexts outside of OpenAgile. As a Team Member, being familiar with these tools can help you participate effectively in Consultative Decision-Making.

Brainstorming

Brainstorming is a very basic idea-generation tool. It can be used any time the team feels that the current ideas being considered are insufficient in their diversity. Brainstorming is an act of group creativity. During brainstorming, all members of the team contribute ideas without any filtering. There are several specific techniques for doing brainstorming. Please see http://en.wikipedia.org/wiki/Brainstorming for more information.

Six Thinking Hats

Edward de Bono created a role-based method for helping individuals and groups to think more effectively. It can be used when the team is concerned about their own thinking habits and wishes to develop new thinking habits. It is based on a metaphor of six coloured hats that represent different types of thinking:

1. Red: emotions, gut reaction.
2. Blue: overview, process, thinking about thinking.

3. White: data, facts, information.
4. Green: creativity, provocation, investigation.
5. Yellow: logic applied to identifying benefits, harmony.
6. Black: logic applied to identifying flaws or obstacles.

In a group, everyone metaphorically agrees to put on the same hat for a short time. For example, with a difficult issue facing a team, the team might choose to put on the red hat first. Each hat is put on in sequence. One person, a facilitator, always wears the blue hat.

Edward de Bono is also known for coining the term "lateral thinking" and is the founder of the Cognitive Research Trust (CoRT). Please see http://en.wikipedia.org/wiki/Edward_de_bono for more information.

The Core Protocols

The Core Protocols are a set of techniques and personal commitments that combine to create an effective foundation for interaction in teams. At team might choose to use the Core Protocols as a way to carefully define interactions, particularly in an environment where emotional safety is threatened. Here are two examples of the Core Protocols:

- "Check In" - share your emotional state and let others know you are ready for working with the group.
- "Ask for Help" - one person asks another to do something, provides details and then the other person says "yes", "no" or offers an alternative.

Please see http://www.mccarthyshow.com/the-core-protocols-online/ for more information.

Appreciative Inquiry

This approach to group change helps people focus on individual and group strengths and then building upon those strengths. There is a very

strong avoidance of problem identification and problem solving. Appreciative Inquiry might be used by a team when individuals on the team are very sensitive to negative statements. Appreciative Inquiry is a process with four steps:

1. Discover – identify processes that work well.
2. Dream – envision process that would work well in the future.
3. Design – prioritize and plan future processes.
4. Destiny/Deliver – implementation of the proposed design.

Please see http://en.wikipedia.org/wiki/Appreciative_inquiry for more information.

Non-Consultative Behaviours and Techniques

There are several approaches that groups and teams use to interact that are not compatible with or simply not relevant to Consultative Decision-Making. We will briefly mention four of these approaches.

Discussion

Discussion tends to be informal, and open-ended with no decision-making purpose, nor any sort of united vision or action. An OpenAgile team may engage in discussion of ideas, issues or plans at any time during a work period, but Engagement Meetings and Progress Meetings have focused needs and open-ended discussion is discouraged at these times. If a team finds itself having difficulty with the time-boxes for these meetings, it may be because the team is *discussing* things instead of using Consultative Decision-Making.

Robert's Rules of Order

Robert's Rules of Order is one of the most commonly used set of procedural rules for group decision-making. There are several issues that

arise when Robert's Rules of Order is used in a collaborative team environment. One key thing to remember when looking at using Robert's Rules of Order is to realize that the tools presented are not primarily intended to support collaborative environments as defined by OpenAgile processes. Instead, they are designed for adversarial decision-making environments where there are winners and losers.

Debate

The concept of a debate is also evoked when discussing communication tools used to evaluate courses of action. Debate however, implies an adversarial relationship. It assumes that there are two sides, one in favour and one opposed, and that individuals fall on one side or the other. It is not conducive to generating collaborative solutions, nor does it promote the integration of proposed solutions.

Rhetorical tools are also typically used as debating tactics. Emotional arguments, political arguments, and humorous statements are often used to make one solution sound better than the alternative. The advantage in a debate typically goes to the individual that is more charismatic and personally engaging rather than who may present the best solutions.

Consensus

Consensus is defined as a "general agreement" or "group solidarity in sentiment and belief." Consensus decision-making is a decision-making process that not only seeks the agreement of most participants, but also to resolve or mitigate the objections of the minority to achieve the most agreeable decision. Consensus often results in sub-optimal or compromise solutions where every party to the decision is only partially satisfied.

Summary

The result of Consultative Decision-Making is a decision which all team

members support through unified action. Consultative decision-making is based on the following simple rules:

- Everyone is encouraged to contribute to the discussion; a diversity of views is welcomed.
- No idea is owned by any individual and therefore it is okay to change your own mind.
- Offence should never be given nor taken.
- Personal attacks are absolutely forbidden; aggressive interruption is considered bad behaviour.
- Everyone should be able to fully express themselves in a frank, amicable, and courteous manner.

There are many tools and techniques that individuals and teams may use to support this process. Discussion, Robert's Rules of Order, debate and consensus are not compatible with the Consultative Decision-Making approach.

Chapter Three – Systematic Learning

> *"Don't let your learning lead to knowledge. Let your learning lead to action."*
>
> — Jim Rohn

Systematic Learning encompasses all aspects of learning for the individual, the team and the organization or community. The Learning Circle is the model we use to understand Systematic Learning. In many workplaces, learning is minimal or constrained to very narrow aspects of the environment. Career development focuses on a person's specialization. In some more progressive organizations, staff are supported in learning outside their specialization. The Learning Circle is a model for learning that takes us far beyond this individualistic approach to include teams and organizations or communities and to systematically build capacity. For this manual, we will focus on the individual and the team.

In the Learning Circle model, the individual Team Member and the team as a whole engage in a symbiotic relationship. Each of the steps of the Learning Circle provides opportunities for the individuals to strengthen the team and for the team to strengthen the individual. For example, the Cycle Plan represents a collective commitment to doing some work. In the context of this Cycle Plan, an individual makes decisions about personal tasks and activities. The Cycle Plan, a team outcome, therefore strengthens the individual's ability to take action. The symbiotic relationship of the team and the individuals within the team is also seen in the stages of team development (see below for more details) – if the membership of a team changes, it can re-set the team's progress through these stages back to the beginning.

Systematic Learning as a foundation also underscores the idea that any work performed can benefit from learning. With learning, we can

improve the results we deliver as a team, the processes we use, the individual skills we apply, how we work together, and the tools that we utilize.

The Four Steps In-Depth

Let's look in more depth at the four steps of the Learning Circle, keeping in mind that each requires a specific capacity.

Reflection

During the Reflection step the team is engaged in collecting data, impressions, histories, stories, and any other factual observations about the work that has been done or is being done.

Reflection requires the capacity for Detachment. To understand Detachment, let's examine its opposite. We are attached to places, ideas, food, people and many others things which give us a sense of belonging or comfort. For example, a child becomes emotionally attached to a particular favourite toy. This attachment may become unhealthy for the growth of this child. The child's development is harmed if their attachment is so strong that they become disobedient, refusing, for example, to put down the toy when the mother says that it is time to eat. In a work environment, our attachments are often to ideas or things or lines of action that we have set ourselves upon. If we become overly attached to the things we "own", then we become blind to the value that may exist elsewhere.

When Team Members come together for Reflection, Detachment means letting go of the emotional connection to certain ideas, principles, theories, and practices and be open to something new. For example, imagine someone is attached to a certain course of action due to a hidden personal stake in the results. This person is going to struggle to be open to new ideas on this subject. How good will this person be in a group consultation about alternative courses of action? Only by becoming

detached will this person be useful in the consultation.

Detachment, like any human capacity, grows with practice. In OpenAgile, we practice Detachment in the Reflection portion of the Engagement Meeting and the Progress Meeting. We also practice it during any other Consultative Decision-Making sessions where we do not personally agree with the end decision.

Detachment also applies in our attitudes towards other Team Members. In many teams, when Reflection is being done, the team will start by reviewing the "Retrospective Prime Directive":

> *Regardless of what we discover, we understand and truly believe that everyone did the best job they could, given what they knew at the time, their skills and abilities, the resources available, and the situation at hand.*
>
> (http://www.retrospectives.com/pages/retroPrimeDirective.html)

This statement reminds us that intention, circumstance and results are separate.

Detachment supports Reflection by allowing us to lessen the impact our biases and filters have on our ability to see reality. Effective Reflection results in unbiased sharing of data, truthful sharing of feelings experienced, and openness to seeing "the elephant in the room"... if there is one!

Learning

The Learning step is where we become conscious of our learning. In OpenAgile, Learning follows Reflection in the Engagement Meeting and in the Progress Meeting. Learning is the stage where "discovery" of new knowledge takes place. The form and nature of this discovery is varied. It could be insights into skills, relations, structures, obstacles and failures that can enhance the work flow. Sometimes we learn something from outside of our team's work that applies inside the team or we learn

something inside the team that applies outside the team's work. A team going through the Learning step will look for patterns, principles, insights, and deductions.

In this step, we need the capacity for Search. We search through our reflections, actions and plans to find what we have learned and make it conscious. This search can be systematic, it can be guided or it can be haphazard. There are a number of tools for systematic search that are valuable:

1. Five Whys. This is a return to the curiosity of a child: ask yourself "why?" five times in a row about something you have observed, each time digging deeper to get to the root cause. See http://en.wikipedia.org/wiki/Five_whys for more information.
2. Identify Themes. As a team, use one or two word phrases to describe high-level concepts and ideas that can be found in the work of the team. For example, if the team has collaborated particularly well in a Cycle, then someone might identify the "teamwork" theme.
3. Highlights-Learning. As a team, collect examples of the highlights of a Cycle or Work Period, then ask everyone the question "what did you learn?"

Some of the supporting techniques from Consultative Decision-Making (e.g. Brainstorming) can also be used to help with Search.

Search can also be a process of filtering. When looking for patterns or insights, we can start to re-apply our filters and biases. We pay attention to certain things and ignore others. If we are conscious of this process of filtering, we can become adept at creating and using mental filters in creative ways.

Planning

In Planning, the team takes the discoveries of the previous steps and

makes a concrete plan of action. The plan includes new ways of doing things that are directly based on the principles and insight acquired in the Learning step. In fact, it is critical that the team is systematic about making these connections. An important aspect of a team's maturity is reflected in this ability to connect Learning to Planning.

When a team is first starting out, this connection-making is mostly done by re-examining the Value Drivers and Tasks and making changes to the current Cycle Plan in light of the learning. For example, a team working to create a new product might have learned that a stakeholder has a large, valuable opportunity to be addressed and so a new Value Driver is created to address this opportunity and it is put high on the list of Value Drivers so that it will be worked on in short order. This adjustment produces better results, and is a very basic way of connecting Learning to Planning.

As teams mature, and patterns of work emerge, the team will systematically use learning to improve the way work is performed, diversify the number of people on a team who can perform a given task, and improve the tools used within the team. For example, a repetitive task is identified, and over time, the team improves the way this task is performed. First, it might be improvements in the manual processes used for the task. Eventually, the team might invest time to automate the task using some sort of tool. These types of adjustment generally increase the capacity of the team to produce results and improve the quality of the results delivered. Again, this is a way that the team connects Learning to Planning.

Advanced teams will measure their improvements and may even become innovative in how they apply the OpenAgile framework.

In the Planning step Love is the prerequisite capacity. This is a vast subject. In the context of a single Team Member it has two important aspects: the passion that we have for the work and the willingness to make sacrifices for the success of the team. It is through this passion and

willingness to sacrifice that a Team Member is able to make plans that are connected with learning, and produce great results.

During the initial formation of the team, Team Members are often selected based primarily on skill sets. This selection process often ignores or discounts a person's Love of their work. Why is Love important? If we examine life's great artists, scientists, craftspeople and leaders, we notice there one is one common element in all of them: great love for their ideas and their work.

We can also understand love in the context of its opposite: hate (or perhaps apathy). If an individual hates their work, then the work results are going to be poor, that person will be resistant to learning and improvement, and in particular their receptivity to Guidance is greatly diminished.

This love for the work and love for the team is contagious. If you are new to a team where the other Team Members exhibit this Love, then you will be more likely to feel it yourself. If this Love does not exist within the team, then it is a great struggle for the one individual who does feel this way. Again, the relationship between the team as a whole and the individual Team Member is critical.

If you do not exhibit this Love as you move into Planning, then there are a couple simple things you can do that may help to develop this capacity for Love.

1. What do you like? Focus on Planning the Value Drivers and Tasks that you like to do personally. During Planning, spend more time working with Team Members that you like working with.
2. Serving your stakeholders. Many of us are motivated by the idea that our work has value to someone else. Think about the true needs of your stakeholders and how you can use your own talents to serve them. As you go through a Planning step, emphasize the things that you understand serve the stakeholders.

3. Find ways to create beauty. We are attracted to beautiful things, ideas, environments, processes. In our Planning work, think about how we can make a beautiful plan and create beautiful results. (We discuss beauty in more depth later on in this manual.)

Action

Systematic Learning requires us to take Action. Action allows us to remain grounded in reality. Action also allows us to change reality – to apply our Learning – and to see how reality responds!

In environments where learning is an afterthought (if it happens at all), taking action is easy. Action in a non-learning environment quickly becomes habit, "the way things are done". There is very little personal risk when all things are known.

In OpenAgile, where Systematic Learning is so large a part of the way of working, taking Action can be more difficult. In fact, it requires Courage. Every time we take Action in OpenAgile, we are moving into the unknown. We are experimenting.

We must balance two considerations when we take Action. The first is that we are Acting with some destination in mind – our Goal. The second is that in order to advance our learning, we must also move into the unknown.

The capacity of Courage is perhaps the easiest to understand of all the capacities of the Learning Circle. Courage is something that we see easily in others, and often we desire to emulate that Courage.

The Learning Circle and Other Models

The following is a brief comparison between the Learning Circle and three other related models.

Boyd's Observe-Orient-Decide-Act Cycle.

The Observe-Orient-Decide-Act (OODA) loop was originally developed as a model for the way a jet fighter pilot works in a combat situation. Observe the situation, orient on the critical aspects of the situation, decide on approach, then act to execute the decision... and repeat. All other things being equal, a fighter pilot who goes through this cycle faster in combat than an adversary will prevail. By going through the cycle faster, the pilot will influence the decisions of the adversary. The pilot generates activities that give the impression of confusion and disorder because the adversary cannot orient quickly enough. The adversary may then under- or over-react to the situation. The concept has been adopted beyond the military into corporate environments.

OpenAgile's reflection-learning-planning-action steps map closely to observe-orient-decide-act. The primary difference in the steps is in "learning" versus "orient". The learning step includes, but goes beyond just recognizing the salient features of a situation (orientation). The Learning Circle model also includes a number of features explicitly that may only be implicit in the OODA model. For example, Guidance in the Learning Circle is similar to the direction a fighter pilot may be receiving from command or surveillance. To some degree, a team going through the Learning Circle model faster will help them to be more successful. However, the OODA model puts a much greater emphasis on this speed, and an OpenAgile team depends on other things besides just speed.

Of course, in OpenAgile, we aren't always thinking in terms "winners and losers" in a battle field (although certainly competition can be part of the environment for an OpenAgile team). Other factors may be much more important such as delivering value, time to market or customer satisfaction... or even just growth for its own sake.

Inspect and Adapt

The Inspect and Adapt methodology, developed by Ken Schwaber, is

well known among agile software practitioners. Simply, the objective of inspection is find any defects in the software before it is delivered and the adapt part is to make changes in software development practices in the hope improving quality.

When comparing to OpenAgile, inspect corresponds to reflection and adapt to action. However, OpenAgile has the added benefits of learning and planning steps. Both give the team the advantage of using and implementing the discoveries of insight, skills, and relationships during reflection immediately while planning of the next cycle.

The Scientific Method

The scientific method is a well known technique that consists of measuring and learning fields of observation, such as physics, chemistry and space and deriving certain knowledge. The first step is to gather data, which must be observable, empirical and measurable, on a subject. The procedures may vary from one field of science to another, but they all follow the basic principle of reasoning. The collected data is then formulated into hypotheses that explain the target field of observation or phenomena.

The end result is to acquire new knowledge, correct existing knowledge, or integrate with previous knowledge. Another essential component of this method is that the hypotheses must be testable. This is a nutshell description of the method which can be translated into three distinct steps: Do, measure and learn.

In mapping the scientific method to The Learning Circle, the steps of the former, do, measure and learn, are similar to action, reflection and learning of the latter.

The duration of each step in these two methods differs vastly. In OpenAgile's learning circle, every step is timeboxed. Reflection, learning, and planning take between one quarter to one half of the engagement meeting, which is proportional to the cycle duration. A cycle

is typically from one week to one month long. This is a rule of OpenAgile.

In steps other than doing work, this restriction is imposed to avoid endless discussion with little or no value to the team. However, in the scientific method there is no strict guidance on how long each step must last. One single step such as data collection could last months and years depending on the field of observation.

Another difference between the two is the demonstrable result. In OpenAgile the team delivers work of value at the end of each cycle. Also, there is a possibility that during the course of a project the team decides not to do certain work because it has no value to the organization. Science projects are usually funded by universities, government and companies and before an initiative is approved, it must meet certain criteria of justification.

However, it is also generally understood that the gathered data may not lead to any understanding of a target. Only one cycle is done per project. Unlike OpenAgile methodology, there is no built-in mechanism for frequent assessment during the project and abort it if no value is perceived.

Summary

Systematic Learning is embedded in OpenAgile with the Learning Circle model. This model is applied formally in the Engagement Meeting and the Progress Meeting. It is applied informally throughout a team's work. Systematic Learning is done both at the individual Team Member level as well as collectively at the team level.

Part Two – Practicing OpenAgile

Team Members are the individuals who are the contact-point for the OpenAgile processes. If OpenAgile does not work for these on-the-ground people, then it does not work, period. Thus, for the Team Member, OpenAgile must be practical. It must provide more benefit than the cost of using it. In this second part of our Team Member Manual, we will examine many practical aspects of making OpenAgile not just practical, but dramatically valuable. Of course, doing this rests on the foundations we just covered. We will see the connections between the foundations and the practical aspects of OpenAgile throughout this second part of this book.

As you are reading this part of the book, please keep in mind these questions:

1. How do the foundations of OpenAgile support what I am learning about and how does what I am learning about improve my ability to keep to the foundations of OpenAgile?
2. What will I do with what I am reading when I return to my day-to-day work?
3. How will these practical aspects of being a Team Member make my life and the life of my team easier or provide enhanced results?

Chapter Five – Processes, Tools and Practices for Teams

Team members, especially at the early stages of adopting OpenAgile, face many challenges, which are understandable, given that this is true with any methodology. This section gives some helpful ideas on how to make the transition easier. One of the greatest hurdles facing newcomers is that the team take precedence over personal preferences, a concept that may take time to understand and embrace wholeheartedly. As humans, not as robots, we do experience mental fatigues, require quiet time for personal reflection or have our creativity numbed by routine work. The following processes, tools and practices are ways to help Team Members' work easier.

Visible Cycle Plan

A practice that results in realizing some key efficiencies through OpenAgile is to use visual tools to develop and present the team's Cycle Plan. This allows all Team Members to become almost instantly aware of:

- The completion of tasks;
- Required changes to the cycle plan in process;
- Whether or not activities are progressing as estimated; and
- What they should be working on.

Additional benefits accrue when other project stakeholders have access to the visible Cycle Plan. Truly effective Cycle Plan display boards will inform everyone who views them significant details regarding the plan without having to ask anyone or require documentation to explain. The purpose of the Visible Cycle Plan is not only to document and plan activities and tasks, but to quickly communicate it to those that see it.

In many cases it may not be possible to post the Visible Cycle Plan in an

area where all stakeholders can easily or regularly come into contact with it. In some cases it may become desirable to use a digital or electronic version of the Visible Cycle Plan that all Team Members have access to. In OpenAgile the vastly preferred approach is to use an actual physically-posted Cycle Plan whenever and wherever possible, with the less optimal but sometimes necessary electronic version used when necessary.

Wall of Post-Its

One tool that is used by many types of OpenAgile teams is the wall of sticky notes. A key part of the visible plan is to make it simple, easy to maintain, easy to understand, and easy to modify. There are few tools that meet all of these requirements as effectively as a number of coloured sticky notes. There are a number of methods of using these notes in effective ways to promote the communication of the cycle plan in its current status.

One tool that has been developed is the concept of the OpenAgile Task Wall, which is a design for a wall of sticky notes that presents an easy to understand method of communicating a cycle plan, its value drivers, the types of activities, the specific activities, and reports progress.

TODO: PHOTO AND DIAGRAM OF TASK BOARD[i]

Big Visible Charts

The "Information Radiators" sometimes referred to as Big Visible Charts is used by many agile teams to communicate information about a project. It is a large display of information that the team is tracking. In case of OpenAgile, it could be cycle plan, its value drivers, types of activities, the specific activities, and reports progress. One important aspect of this technique that any member is free to update the information radiator and it is located in a spot where the team can see it constantly.

Whiteboards, flip charts, poster boards and electronic screens have been used as display media for information radiators. A team is free to choose

an appropriate medium, including coloured sticky notes or index cards and push pins. Regardless of the choice of medium, critical to its effectiveness is how well it provides critical information to the team and stakeholders. Equally important is the ease of tracking and updating categories of information important to the team such as value drivers, cycle task progress, remaining obstacles, repetitive activities and metrics.

Calendar Events

Calendar events is a category of tasks that are scheduled based on date or time. Some examples of these items are meeting with clients, a trade show, phone call follow-ups, employee birthdays, a company picnic or team lunch. However this list does not include OpenAgile meetings, such as the Engagement Meeting. The team must have a venue for adding and updating such events that happen during the course of weeks and months. A large calendar posted on a wall, clearly visible to the members, and adjusted with ease as required would suffice. During the planning period of the Engagement Meeting, this calendar is checked for any items that need to be scheduled during the current cycle. Visibility and ease of use would be the criteria in choosing a tool to keep track of calendar events without missing anyone of them.

Repetitive Activities – Team Standards

Repetitive Activities as the name suggests are routine tasks to the point of becoming brainless work. Once started, these day-to-day operations do provide opportunities for improvements by creating new standards.

Identifying basic repetitive activities

There are two kinds of repetitive activities: time base and event base. Time base category items occur on a regular basis, such as daily sales summary report, weekly database backup, bi-weekly cheque printing, monthly office cleanups and yearly preparation of government

documents. Some of these activities are required to be done by the team and others, like office cleanups, are usually taken care of by external crews but the team needs to know in case it is required to do a minimal amount of work. The event base activities happen as a result of a calendar event. For example, a trade show would be classified as a schedule activity which can generate follow up work, such as phone calls and potential client visit. An employee birthday celebration is considered a calendar event but the actions, such as finding a venue, buying a gifts and making an invitee list, taken to make that event happen belong to the repetitive activity category.

The team finds a way of how to manage these activities and incorporate them during a cycle planning. Again visibility and ease of management would be the criteria in choosing a method.

Taking a learned skill or process and making it visible as a Repetitive Activity

The VP of sales in a wireless phone company needed a daily summary report delivered to his office every morning for a ten o'clock meeting. To fulfill this request, the IT maintenance department split this request into 7 tasks among three Team Members accord to required skills. Daily the tasks were executed in the following ways.

DBA: 1. Check the storage capacity of the data warehouse and top it up if needed. 2. Load into the database the previous day sales information and transform it into company standard format. 3. Check for any failure, if none found, inform the developer.

Developer: 4. Run the computer job to generate the daily sales snapshot report. 5. Deliver the report to a BA.

BA. 6. Check the report for integrity and correctness. 7. Deliver the report to the VP's office.

In a traditional environment where roles and tasks are highly correlated

changing these processes would not be easy. In OpenAgile and especially if all three workers were in one team, this structure would beg for a review. The first five items can easily be done by one person, possibly some training would be needed. Therefore, examination of processes and skills is an ongoing activity.

Habits and Repetitive Activities

The activities that we do which have developed into habits are special. They are often things we do very efficiently but also "mindlessly" - our minds can be occupied with other things rather than concentrating on the activity at hand. As a result, habits can develop such that we have blind spots. As an example, we (hopefully) all brush our teeth daily. And when we brush our teeth we probably do it pretty much the same way every time. If we have developed a brushing habit that misses a spot, we won't be aware of it. We probably won't know until we go to the dentist and discover that there is a cavity developing in that spot. The habit is efficient, but it is imperfect. By making the habit visible as a Repetitive Activity (going to the dentist), we can evaluate how we are doing the habit, and possibly correct it.

Work standards, the definition of "done"

Work standards are a definition of how to do a piece of work. They are set out by the Team Members and should not come from an external source. It does not need to be perfect initially. For example, in writing a piece of software would just mean 1. Writing the code. 2. Checking it into a repository. These two steps define what this developer considers "done". Once created, it must be examined regularly for improvements. After a few cycles, it may be decided by the team to expand the definition of "done" by adding one more step such as unit testing. This process of analysis and improvement continues until the end of the project.

Team Standards

Fundamental to the OpenAgile framework is that a team exists for the purpose of delivering work of value. During the life of a team, as it moves cycle by cycle towards the final goal, learning how to improve this delivery is at the heart of all activities. The team defines and refines processes, skills, modes of behaviour, and knowledge. As result a culture emerges as soon as the start button is pushed. They become the standards by which future work is measured. How to document and make them visible so none get ignored? For Post-It lovers, one method would be to write them down and post them on a wall for clear visibility.

Guidelines for Completing Tasks

Sometimes, as we are working, we become "stuck". Our brains disengage; we are sleepy after a heavy lunch; we aren't motivated by the tasks in the Cycle Plan. When this happens there are a few things that we can do to overcome this mini-obstacle. The following are some simple tactics you can use to help you get back to a productive state.

Self-organizing and volunteering

Teams are self-organizing in the sense that they may change their own rules and methods and Team Members may choose their own roles and tasks. The Team Members volunteer for roles and tasks based on experience, desire to build capacity, or a sense of connection to specific tasks that the team has prioritized. As the team learns, the Team Members increase their capacity to accomplish tasks and to make appropriate commitments. Self organization also allows for adjustments in the OpenAgile model; however, it is desirable to first gain experience with the model and adjust only after other approaches are proving more effective for building the capacity of the team and it's members.

Challenges to volunteering and executing

At times Team Members may volunteer for tasks requiring skills or experience they don't currently possess and, as such, may have difficulty executing the tasks. While the tasks are viewed from the perspective of learning, in a work environment, they also represent actions and commitments which must be accomplished. Team members are encouraged to take on only those activities which they feel they can achieve during a cycle, but are also encouraged to get help when needed either from other Team Members or any one else who may have the expertise to accomplish the task, being mindful of external limitations (e.g. budgets, schedules and commitments of others). Challenges with the executing of specific tasks should be brought to progress reviews.

Asking for help

Volunteering for a task does not mean agreeing to complete it before the end of the cycle. It is simply an attempt. During the course of action, there are many situations that could prevent you from getting the task done. Lack of skill, inadequate time, or mental or emotional readiness could be some of the reasons. Once discovered, asking for help from team mates would be an appropriate action. However, it should not be taken as a failure because in any collaborative engagement helping one another is essential to the well-being and progress of the group. The most appropriate time to bring such an issue to the attention of everyone is during a progress meeting.

Working in pairs

The concept of Guidance is an effective way to build capacities. One strategy to facilitate Guidance is to pair Team Members, which can be part of the planning. How often do we hear the tutors expand their understanding of a subject while sharing their knowledge and experience with a junior. The same happens in the other direction -- the less skilled

person learns the tricks of the trade from a senior member of the team. This is a win-win situation for the team. Also true is when two persons of the same level of skill work together -- learning takes place. A brilliant moment can happen by just explaining your idea to someone.

Taking a physical break

One effect that OpenAgile has is that the expenditure of effort of the team is very efficient. A major benefit of developing the Cycle Plan is that the required activities and their priority have already been determined. This results in much less time spent between tasks determining what to work on next. This reduces transition time between tasks and can have the effect of making people feel rushed or pushed from one task to the next with few natural breaks.

One fundamental concept behind OpenAgile is the steady and sustainable production of value. Sustainability can only be achieved by establishing a pace of work that can be maintained from one cycle to the next. The goal of OpenAgile is not to start and finish work tasks as fast as possible with little or no time for physical breaks. The goal is to produce regular and reliable production of value during each Cycle Plan. Regular physical breaks are very effective at energizing Team Members to enable improved productivity throughout the workday.

Mental and emotional readiness

Statistics reveal that new teams usually experience many crisis during the first few cycles. This challenging period is normal and is one of the basic stages of team development which are discussed in detail in the next section. However, as comforting as this knowledge may be, members will experience stress which can undermine their emotional and mental capacities. The challenge is to recognize crisis as they happen so as to address obstacles before they become serious impediments.

There are two ways to become aware of this situation. It is easier with

self-knowledge, which is one of the cornerstones of truthfulness and crucial to understanding one's own capacity.

Another method is have all Team Members be watchful for emotional and mental stress. One way to recover from this setback is to take a physical break. Another strategy is to shift from one type of work to another, for example, from demanding work to routine activities or from boring work to challenging work.

Attitude of service

The attitude of service can become a mini-obstacle. How we approach work can differ from one person to another and also from one team to another. Since an OpenAgile team's primary aim is to provide quality work of high value to the organization, let's look at some of the incompatible attitudes in such an environment. Comments like "I have to do it", "It's just work," "Have to get it done to satisfy the boss," and "I need a job to live on" will not agree with the principles of OpenAgile. In this methodology, teams are collaborative, oriented towards excellence, consider love of learning as one of the cornerstones of their success and appreciate the honest contributions of all Team Members. New members may require mental shifts towards these attitudes of service that the team strives for.

Another aspect to keep in mind is the symbiotic relationship that exists at the levels of team, organization and community. A team serves the need of capacity growth of its members and the members serve the team towards fulfilling its goals; the team serves the organization by delivering work of value and the organization serves the team through financial and other supports; and the organization provides goods and services that the community needs.

Effective Reflection in the Engagement Meeting

If The Learning Circle is the heart of OpenAgile, then reflections are its

arteries. The results of the analysis of the previous cycle have a profound effect on subsequent steps in the current cycle and beyond. As seen in the previous section, the team develops a culture of learning and problem solving which emerges gradually. Reflections start with a demonstration of work that was just completed. Understanding the full range of shareholder reactions is a way of gauging the value of the work. At this point, the team must have enough questions to fathom the satisfaction of the team's client.

Some of the queries would be regarding the technical integrity of the product, the specification of the product, the quality, communication between the team and stakeholders, and interpersonal skills. But also, there are many non-verbal communications that take place at these meetings, such as looks of satisfaction, the tones of worry and language of mistrust, which should also be noted.

Next, only the Team Members examine the completed tasks of the Cycle Plan of the prior cycle by simply asking what went well and what needs to be improved. If this section is effective, then it will provide the right input to the learning step, which will flow through planning and action. It will provide the right amount of oxygen to the rest of the cycle and keep the pulse regular and healthy. The proof is in the pudding.

What if things are not going well? If the team discovers that it is not functioning as well as it should, then there are a few actions it can take. First and foremost among action items is to review the three foundations, in terms of understanding and applications. Setting aside time for some deepening would be a good idea. Without resorting to blaming and shaming, are the Team Members adequately equipped for capacity self-evaluation? Is self-knowledge happening? Is the team being truthful about their understanding of the value drivers? Is the customer participation adequate? Is there a need for mentoring? When it comes to consultative decision-making, is anyone dominating the meetings? What capacity needs to be built for the team, individually and collectively? These are some of the questions that can be asked to improve to

effectiveness of Reflection in an Engagement Meeting.

There are also a few practical steps which can enhance gathering data, history, stories and other impressions -- these may lead to better insights in the team's work delivery. One such step is mind mapping. Take the ideas, concepts, insights, capacities, and learning of the previous cycle and write them down on a blank sheet of paper. After some reflection, draw lines between them. It tells the story of what happened.

Another method of evaluating work done is to ask each member to write down two things that went well and two items that will need improvement on Post-It notes. After all of them are posted on the wall where everyone can see, ask the team to group them in categories. What will emerge are ideas that can be the basis for more exploration.

Another way is a calendar review. Post all the tasks on the wall, gather the team around them and do an inspection of what has been done, what went well, what went less well, and adapt so that the team do things better next time.

Chapter Six – Teamwork and Teams

Teamwork Skills

Team Members need to be able to work with a team. The skills to do so are referred to as Teamwork skills. Teamwork skills include a broad range of soft skills as well as a few skills that are more concrete. It is critical that Team Members have and develop teamwork skills, however, just throwing a bunch of people with these skills together does not make an OpenAgile team.

The soft skills include:

Active Listening: Active listening is a skill that allows a person to completely focus on the communication of another person including both verbal and non-verbal aspects. Active listening requires the ability to not think of our own responses until after a person has finished speaking. One simple way of doing this is to echo what a person is saying in our silent internal voice. When someone says "I think we should build a new gimbal on the widget", we are saying exactly the same thing in our own mind. Active listening also requires that we request clarification, often by rephrasing what a person has said and asking if we have understood correctly.

Questioning: Being able to frame and express questions effectively helps us understand and integrate knowledge into our own mental model of the world, or even to modify our mental model. Asking questions is easy. Asking good questions is much harder. We need to use an appropriate set of words and tone of voice so that we do not alienate or offend the recipient of the question. For example, asking "why did you do that?" will often put people on the defensive since they will assume that we mean we disagree with their actions. Instead, saying "I do not understand the reason you did that. Could you please explain it to me?" can be a much more gentle way of getting to the same information.

Logical Argument: When presenting an idea or position, being able to logically support it is important to exploring the truth of it. This includes being able to share our assumptions or axioms, the data we are basing our argument upon, and the logical sequence of reasoning to reach our conclusion. Being able to avoid fallacious logical methods is also important.

Respecting: Showing respect includes acknowledging the fundamental human value of the existence of our teammates, and being able to step back from our own understanding of the world to acknowledge the legitimate nature of the perspective that other people have. This does not mean that you have to let teammates get away with inappropriate behaviour. In fact, respect for our teammates will allow us to support them in behaving in ways that are in alignment with their fundamental nobility as human beings.

Helping: Offering help and actually following through with real assistance are aspects of helping. When we suspect that a Team Member is struggling with something, we offer to help both verbally and with our actions. This can take the form of offering information, offering emotional support, offering to assist with problem-solving, or actually taking action to do an activity together. When we help someone, we share their burden.

Sharing: Sharing our knowledge, time, skills or physical resources are all aspects of sharing. Sharing among Team Members is focused on those things which will help a team reach its goals. This is similar to helping except that it tends to be more of a transaction than an ongoing activity. The transaction is that we give a gift and then the other person uses that gift to meet their needs. Sharing does not require reciprocity. If we share something with another person, we should not expect that that person will return the gift at any time in the future.

Participating: It's probably obvious, but in order to effectively be on a team, we need to participate! Participation itself is mostly obvious: do

work with the other Team Members. However, there are also some less obvious aspects of it. We are not participating when the team is having a discussion, we find it boring, so we check our email. We are not participating when the team makes a decision and we abstain from helping to execute the decision because we disagree. We are not participating in a work team when we are mentally checked out because of a crisis at home.

Eight Team Must Haves

This will be reviewed by your OpenAgile instructor, process facilitator, growth facilitator or your coach or mentor.

The Five Dysfunctions of a Team

From the book "The Five Dysfunctions of a Team: A Leadership Fable by Patrick Lencioni" According to the book, the five dysfunctions are:

- Absence of Trust
- Fear of Conflict
- Lack of Commitment
- Avoidance of Accountability
- Inattention to Results

Source: http://en.wikipedia.org/wiki/The_Five_Dysfunctions_of_a_Team

Team Self-Organization

OpenAgile is most often used with teams. A good team size is anywhere from about five people to twelve people. OpenAgile can of course be used for goals where many fewer or many more people are involved. No matter how many people are Team Members, there is a very important principle in OpenAgile: teams must self-organize.

Self-organizing is not something that is easy to do, and yet we do it all the time. Self-organizing is about making individual choices on how to contribute to the overall goal. Most of the time in our work environments we are doing this. We decide how much to focus. We decide when to take breaks. We decide on many of the details of how we accomplish a task. We decide if we will ask for help from our co-workers. We decide if we will try to do things more efficiently. But what we don't usually do is decide which tasks to take on – often these are assigned or defined by our work role.

Self-organizing in OpenAgile means going that extra little bit to volunteer for tasks instead of having to be told by a boss or manager. There is no boss in OpenAgile. The Cycle Plan is created by the team, and every member of the team is responsible for completing the whole Cycle Plan. Sometimes, as we explained earlier, that means taking a Task that we might not be totally comfortable with.

Even in OpenAgile there are limits to self-organization. The Learning Circle tells us that we need guidance. Since guidance often comes from authorities, we know that there may be cases where an authority knows best and we should follow their guidance. Sometimes, a team will give up some authority to self-organize in order to have a strong Process Facilitator or Growth Facilitator. Of course, when using OpenAgile for an individual project where you are the only person working, then you must be completely self-organizing!

A self-organizing team has some important characteristics. There must be diversity or differences between Team Members. As an example, suppose your organization was merging with another. Then a team to guide this merger would ideally have members from both organizations so that the diverse cultures and skills are represented. As well, a self-organizing team needs a boundary, a way of defining who is part of the team. Usually this is through the members of the team committing to a common goal. To continue with the merger example, this could be the goal of creating a healthy new organization that takes the best of the two

previous organizations.

Finally, a self-organizing team needs to allow all members of the team to have valuable exchanges with any other Team Member. There should not be a single person who acts as a communication gatekeeper. Again, with our example, a Team Member from one of the merging organizations should be able to communicate with a Team Member from the other organization without always having to get permission or "go up the ladder".

Team Development

Four stages of development

Groups of people working together as a team can anticipate going through four stages of development. These stages are usually referred to as Forming, Storming, Norming and Performing. In Forming, a team is coming to grips with a basic understanding of the collective goal of the team, and usually Team Members are very open to learning about their new environment and each other. Forming is a stage that is characterized by feelings of enthusiasm, excitement, optimism, and fun. After Forming, a team will move into the stage of Storming. A team is Storming when personality conflicts arise, when the Team Members become discouraged by the effort needed to reach their goal, or when they realize they don't have the resources they need to easily reach the goal. Feelings of anger, pessimism, low morale, and exhaustion are common when a team is Storming.

Pitfalls of storming

OpenAgile increases the interaction between Team Members through regular engagement meetings, status meetings, and consultative methods. This can impact the storming phase in two ways. Firstly, more direct communication and interaction can highlight personality differences

amongst Team Members and accelerate the onset of the storming phase. This phase is experienced by all teams in some form or other. OpenAgile does not eliminate this phase but instead ensures that it occurs sooner to enable moving to more productive phases more quickly.

Secondly, OpenAgile methods provide the tools for the team to appropriately deal with the challenges presented during the storming phase. Honest communication, consultative decision making, and an environment supporting capacity building, will not only accelerate the onset of the storming phase but ensure it passes quickly and allows the team to move to the norming phase of team development.

Pitfalls of norming

Arriving at the norming stage is indeed a great milestone, which is a cause for celebration. The team has already crossed the dangerous waters of the storming period. The commitments are on target every cycle. The pace, after much learning, is not too low or too high, but just right. By now many habits and standards have become normal practices and most of all the members are enjoying the rewards of delivering work of value to the customers.

However, even at this stage there are many risks of sliding backwards. Complacency is one to carefully watch for as it can sneak in unnoticed. If habits, principles, modes of conduct, focus on quality, consultative process, and honesty have not been solidly internalized to the point that they are team culture, then lack of any one or combination of few can cause the team to regress. The members have to be vigilant in guarding the learning that had taken place thus far and not lose it through bad practices. As mentioned before, team standards must be recorded, be visible, and be reviewed regularly and improved.

Another pitfall is to gear up to high-performance level too soon. Over confidence may tempt the team to over commit consistently over many cycles, which can cause conflicts, stress, and too many defects.

At this stage management may pay more attention to the team more than previous phases. Supervisors participate more often at meetings, which is an asset, except when they interfere with the decisions of the team. The Team Members might interpret it as a clue to take on more responsibilities after it has reached a new level of maturity. They would resist the pressure and revert to storming again.

Self-knowledge, highly encouraged in OpenAgile, can also become a challenge if it is not tempered with humility. Individuals with new-found capacities may be tempted to lead decision-making processes at the expense of collaboration and consultative spirit. Bahaviours become strong and stifles unity.

When membership changes the team automatically reverts back to forming. There are many ways, to automatically throw the team into a forming or storming stage. Therefore, it is reasonable to expect re-norming to take place a few times until the team is ready for the perfoming stage.

Crisis and victory

In traditional waterfall projects, crisis happens at the end. A team at a bank was given the task to process transaction returns caused by non-sufficient funds, closed account or any other failures, among banks electronically. This system interfaced with many existing applications used at the bank. After one year and going through normal phases of analysis, design, construction, testing, the project was rolled out into production. On the first day, it blew up. Return-item processing halted throughout the system costing the bank millions daily and not to mention its reputation among financial institutions and customers. It was such a crisis of such a great proportion that the team had to work eighteen hours a day, management held meetings every morning and afternoon and VP's of many departments demanded progress status frequently until the problem was solved.

It turned out that the account field in one of the in-house systems was expanded and all the in-house programs were changed and tested accordingly except in one of the vendors' software. After this crisis was over there were other minor ones such as GL accounts did not balance and printer outputs did not satisfy customer requirements.

An OpenAgile team is not immune from crisis but the serious one happens at the beginning of its life, usually in the first few cycles. This is rarely the case when the team has already worked in a project together before and the members are well versed in the foundations and practices. For a team new to OpenAgile framework, the challenges are many considering the membership of diverse technical capacities, interpersonal skills and OpenAgile experience. The deficiencies of the team will show up in two practical ways: the team not meeting its commitments from one cycle to another and the delivered product is plagued with too many defects. To address these two common issues in early stages the team must carefully examine many aspects of working in a dynamic environment, such as clear understanding of value drives, honest assessment of capacity, effective reflection, translating learning into planning and action, and so on. To avoid crisis, use The Learning Circle as a tool which can increase the understanding of how to deliver work of value in your organization.

After the initial stages, can the team be free from crisis? No, it is not as we saw in the previous section. Going forward and backward through the development stages is what happens. It is normal to slide back to storming or forming from norming. But if the team is experiencing repeated crisis, it means either learning is not happening or learning is not being effectively translated into planning and action.

High Performance Teams

By crossing into the performing stage, a team has already realized that it is not just a group of people working together. It is a real team with a

small number of people with complementary skills, committed to delivering work of value to the organization with common goals and working approaches. The members are held together in a coherent manner by being faithful to the principles embedded in the three foundations, by being passionate about learning and skill development and by developing mutual respect and accountability. When a team has reached this stage, it is flying with new characteristics like a butterfly after it has been transformed from being a caterpillar.

Six prerequisites

The prerequisites of a high-performing team are as follows:

- The team has a disciplined approach to self-management such that it operates as a unit and not as individuals held together in a group.
- It meets its commitments smoothly and effectively without conflict or need for external supervision.
- Team members are highly interdependent and collaborative in finding the most appropriate solutions.
- The team is motivated and knowledgeable in specific lines of business.
- It has the right mix of technical, decision-making and interpersonal skills without any need for external help.
- It knows how to adapt to changes.

The Goal and Cycle Plans

During planning, a high-performance team starts to coalesce around challenging aspirations and, by this stage, it is wise enough not to slide into a lower stage. What would be some of these goals? The most common is to shoot for quality. One specific goal would be to reduce defects by 50% every cycle until the work is defect-free. Once this goal

is achieved it might take quality to a higher notch by daily reducing number of defects to zero. For a team not to falter, it must have already developed methods of tracking defects, recording and making team standards visible at every Engagement Meeting, finding the right technical solutions to increase product quality and making learning as a guide. Therefore during the performing stage, the team is more concerned about how the members deliver work during a cycle rather than the quantity.

Another characteristic that stands out in a high-performance team is effective communication among members and with the organization. Transparency and the right flow of information build trust in working relationships.

Commitment vs. Estimation

Estimation is a best guess based on a given set of information of how long it will take to complete a task. A commitment is what a member or team will do, for example, during a day or cycle. Hitting a measurable high mark of commitment, say over 90% at every cycle is also what high-performance teams aim for, which increases the meaning of purposefulness for members. Achieving such a goal requires constant examination of processes, techniques, and learning. Invariably, the team discovers better ways. For example, a software team decides to implement test-driven development to improve quality. In an accounting department, perhaps cross-training is needed to mail all the invoices to the customers on time.

Creativity

Creativity is a mental process to discover new ideas or concepts or to make new associations of the existing ideas or concepts, enabled by conscious or unconscious insights. It is the aha-moment that happens in a group or individually. How many times have we had that brilliant

moment while doing routine chores or during sleep? However, creativity so vital to an organization is often stifled and it is found that such companies have built many procedures which considers thinking "outside the box" both risky and impossible. For example, a mining company had a unique problem of controlling a sump pump remotely. There were three main challenges, 1. The pump had to be controlled from anywhere in the world, 2. It could used in any pit and 3. Any failure had to be detected immediately and action taken to prevent rising water level to cause mine to collapse, especially when miners are working underground.

Many engineering firms attempted but did not come up with workable solutions because they were thinking in terms of existing practices. The one that was successful had to think outside the technologies currently used in process control and they used wireless internet and specially-made devices.

Creativity is needed in all the OpenAgile foundations but the most crucial one is in consultative decision-making, which is also considered a method of discovery. It does not mean an individual has to surrender all the creative activities to the team. Rather, finding the most effective solutions works at two levels, individually and collectively, by allowing free-flow of ideas and being completely open to where the solutions may lie. For complex and difficult problems, many sessions are required. But the team must also have the skill to adjourn a consultative session when needed. The break allows individuals to reflect and search for the right answer to a specific problem. This may take several iterations. In OpenAgile, the members with the attitude of humility realize that solutions always belong to the team and not to one or a few individuals.

The Arts

Creativity is popularly associated with the arts. Managers, scientists and engineers have to be as innovative as any artist and writer in coming up with new ideas. In fact the mental process in arriving at cutting-edge

insights is the same for anyone and the satisfaction derived from each is also a universal experience.

Problem solving

Lateral thinking, a term coined by Edward de Bono in his book New Think: The Use of Lateral Thinking, refers to solving problems through an indirect and creative approach. Lateral thinking is about reasoning that is not immediately obvious and about ideas that may not be obtainable by using only traditional step-by-step logic.

We are entering an age of management when creativity is extremely important for the survival of any company, which ties in nicely with The Learning Circle. A high-performance team builds enough capacities to become the company's lead-problem solvers and creativity plays a crucial role in reaching this stage. At this level the OpenAgile team is not saddled with rules, practices and customs, often found in traditional teams, rather it is willing to explore new and innovative ideas.

Logic and intuition

Creativity is also associated with intuition, which is acquiring of knowledge without using reason. It is attributed to divine intervention and cognitive process. Intuition means knowledge coming from somewhere external. Opposite to intuition is logic, an intellectual activity of discovery or analysis through specific steps. Though different in mental processes, when they are combined together, the most advanced knowledge is achieved. A woman with throat cancer, after going through many remedies, both from medical care givers and herbalists, came to the end of her road of finding a cure. One day, intuitively she knew that a combination of herbs will eliminate her health problem, but, at that moment, she did not know how. Being an engineer, she set out on a path of studying many herbs in a logical and step-by-step process. After months of analyzing herbs from many parts of the world and gaining

knowledge of the properties of each, she found a handful that could be the recipe of the cure. She came up with a powder composed of the selected herbs and tried it on herself. She found the cure for cancer.

In OpenAgile intuition and logic are human capacities to be fully used in finding the most appropriate solution.

Attraction to beauty

Humans are attracted to beauty, for example, we are drawn to a panoramic view, gorgeous face, exquisite jewel, and intense colours. Our admiration does not stop at the physical level, we go beyond by admiring poetry, movies, stories, art and many abstract ideas that give us exhilarations. Also, there is a third level: inner beauty, a concept used to describe the positive aspects of something that is not physically observable. They are qualities including love, courage, detachment, kindness, sensitivity, tenderness or compassion, creativity and intelligence have been said to be desirable. Do two persons appreciate beauty in the same way? No. For example, someone who is passionate about poetry may not necessarily enjoy listening to rock'n roll music or vice-versa. Each one of us appreciates beauty through the lenses of one's inner capacities. An OpenAgile team is very interactive; therefore, appreciation of beauty in self and others promotes unity of purpose.

Practical Matters

Team Size, Skills, Evaluation, Organizations

A few practical points about teams are in order to wrap up this manual.

- Team Size: Ideal team size is in the range of five to nine people. Smaller and larger teams are possible. The larger the team beyond nine, the more difficult it is for that team to get into a high-performance mode. Team size in reality is a tension between several factors including budget, time, skill sets, and personalities.

- Skills: Ideally the team is composed so that when you "add up" all the skills of the individual members, you have a team that can accomplish the goal without requiring external help. In practice, there are gaps in the skill set. The team is responsible for finding ways to fill those gaps as they move through the cycles. Clearly, learning new skills is an important component of this. However, it may also be expedient to ask for temporary help from people outside the team.
- Evaluation: Evaluating the performance of a team is relatively simple in concept. The evaluation should be based on how well the team is delivering value towards the goal. However, it is important to understand that evaluation and motivation of Team Members are linked together in sometimes complicated and unexpected ways. Review of appropriate literature (e.g. "Drive" by Daniel Pink) will help in making effective evaluation decisions.
- Organizations: Many organizations use the term "team" in a fashion that is different than as is used in the OpenAgile approach. For example, a group of one hundred people that work on a factory floor might be referred to as a "team" by management. Organizations may have to change the use of their language as they adopt OpenAgile. As a Team Member, you can help with this by being precise and using the word "team" only to refer to an OpenAgile team and using other words like "group", "cohort", or "community" to apply to larger non-team collections of people.

As a Team Member yourself, you may not have direct control over these items, but being aware of them can help you influence the direction, composition and effectiveness of your team.

Chapter Seven – Summary

This manual is designed to help guide you and your teammates in developing the basic capabilities—qualities, skills, habits and attitudes—of an OpenAgile Team Member. Towards this objective, practical applications and implications of most of the basic concepts are outlined in the OpenAgile Primer.

- A deeper understanding of the three foundations empowers the team
- Honesty builds trust among Team Members and between the team and stakeholders
- Self-knowledge is needed to accurately assess team's capacity
- Consultative decision-making requires candor, courtesy, honesty, humility, patience, creativity and collaboration
- A decision must be whole-heartedly followed by the whole team, regardless of how it was reached
- Learning, when translated into planning and action, is the key to team's success in delivering work of value
- Visibility tools, when properly used, are effective in making Cycle Plans
- Team standard, when recorded and improved regularly, enhances commitment and creates a culture of learning
- Team members are encourage to ask for help and take physical or emotional breaks when needed
- The team is collaborative, oriented towards excellence, considers love of learning as one’s of its cornerstones and honest contribution by members crucial to its success
- The team looks for ways to make effective reflection in

Engagement Meetings

- OpenAgile must have eight characteristics
- Team progresses and regresses through stages of forming, storming, norming and performing
- A high-performing team, the ultimate stage, holds its members held together in a coherent manner by being faithful to the principles embedded in the three foundations, by being passionate about learning and skill development and by developing mutual respect and accountability.
- Practical considerations, such as team size, skills, evaluation and organization, are essential in a well-functioning team

Acknowledgements and Further Information

Contributors: Mishkin Berteig, Mitra Gopaul, Paul Heidema, David Parker, David Sabine, Mike Caspar, and many others. Thank you all.

http://www.openagile.com/ The official site for OpenAgile which includes presentations, updates of this e-book, information on the Capacity Building Program, and recognized providers of OpenAgile services.

http://www.openagile.org/ The website for the OpenAgile Community. Future updates to the official version of this document are influenced by the improvements contributed on the wiki.

http://www.agileadvice.com/ A multi-author blog that covers all things related to agile methods.

TODO: AUTHORS FOR BOOKS[ii]

The Wisdom of Teams

Getting Things Done

Textbook of Wisdom

Changes

Version 1.0

1. Title page formatting.
2. Consistent headings and text formatting, and same as the OpenAgile Primer.
3. Throughout, fixed grammatical problems, split run-on or overly wordy sentences, simplified phrasing, capitalized references to the foundations and other components of the OpenAgile system.
4. Link to the OpenAgile Primer in the Introduction.
5. Re-wording requirements for Level 2 in the Introduction.
6. Replaced the word “certification” with the phrase “capacity-building” where appropriate.
7. Changed “Deeper Foundations” to a “Part” and added “Practicing OpenAgile” as “Part Two”.
8. Changed foundations metaphor in “Deeper Foundations” to roots of a tree / living system.
9. LARGE: added section in Truthfulness about “Visibility and Transparency”.
10. Removed sentence “Additionally it is better to identify irritants or disruptive behavior early rather than being ignored until anger and conflict boil over and detract from team performance.”
11. LARGE: added detail and concepts to the section “Integrity”.
12. LARGE: added detail and concepts to the section “Self-Knowledge” including a new sub-sections on “Measurement” and “Bias and Filter Awareness”.
13. Added “Summary” section to Truthfulness chapter.
14. Removed a redundant paragraph in lead-in section for Consultative Decision-Making.
15. Edited the “Behaviours” section for clarity.
16. Edited the “Tools to Support Consultative Decision-Making” to add details about four of the tools and remove the rest from the list.

17. LARGE: removed a great deal of detailed material about non-consultative behaviours and techniques.
18. Added “Summary” section to Consultative Decision-Making chapter.
19. Re-wrote the lead-in section of the “Systematic Learning” chapter.
20. Edited the “Reflection” section and added the “Retrospective Prime Directive”.
21. Edited the “Learning” section and added some techniques for effective Search.
22. Edited the “Planning” section and added some techniques for increasing Love.
23. LARGE: complete re-write of the “Action” section.
24. Introduce OODA by writing it out in full for the section heading and in the first place it occurs in text.
25. Re-write of the “OODA” section for clarity.

i TODO: PHOTO AND DIAGRAM OF TASKBOARD
ii TODO: AUTHORS FOR BOOKS