While Agile Software Development is Evolving…

Why ‘Hybrid Agile’ still makes sense? And Where?

Abstract

Now that the Agile campaign has entered cross organizations and industries,
lot of variations are emerging.
Hybrid Agile is a new milestone/buzzword in this journey, where goal has always been value creation.

ENTERPRISE PRODUCTIVITY

Contents

1. Introduction 3
2. Evolution of Software Development Methodologies 3
   2.1 Waterfall Methodology 3
   2.2 Agile Methodology 4
3. We live in a HYBRID ERA........ 6
   3.1 Hybrid Agile / Water-Scrum-Fall / Scrum But 6
   3.1.1.1 Hybrid vis-à-vis Agile vis-à-vis Waterfall 7
4. Industry Shift........ 7
   4.1 Analyst Outlook 7
   4.2 Competitor Landscape 8
5. Summary........ 11

1. Introduction

The new methodologies have always challenged the conventional models. Agile for instance, was developed as a form of opposition to rigid regulations and rules engrained in Waterfall approach.

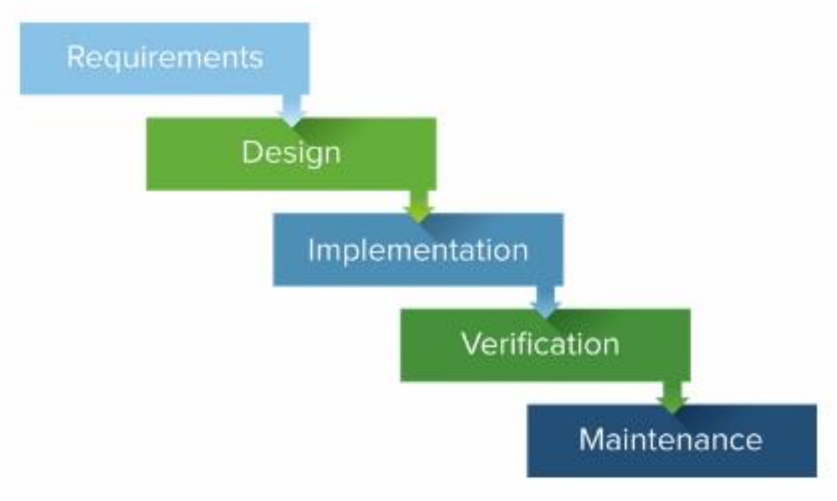
Like every new approach, they have both the supporters and the opponents. But the rise of hybrid methodologies finds a common ground as they bring synergies.

This technical paper is talks about the journey that started with Waterfall, Agile Paradigm shift from there and how Hybrid is becoming the new norm. It also touches on Analyst's outlook and glimpses of key industry players as to how they are adopting Hybrid Agile.

2. Evolution of Software Development Methodologies

2.1 Waterfall Methodology

The waterfall model is a sequential (non-iterative) design process, used in software development processes, in which progress is seen as flowing steadily downwards (like a waterfall) through various phases as depicted below. Originated in the manufacturing and construction industries, in a time when no formal software development methodologies existed, this hardware-oriented model was simply adapted for software development.



ADVANTAGES		DISADVANTAGES	
+	Easy to understand /manage, and discipline is enforced to have rigid timelines	−	Difficult and expensive to change output once in testing
+	Each phase is run through once with no overlap, allows thorough QA at the end of each phase	−	High risks (risk mgmt. done only at planning stage)
+	Works well on small projects with fixed / clear requirements	−	Unsuitable for complex projects or where requirements are evolving
+	Requires a well-documented approach	−	No completed product until the end of SDLC

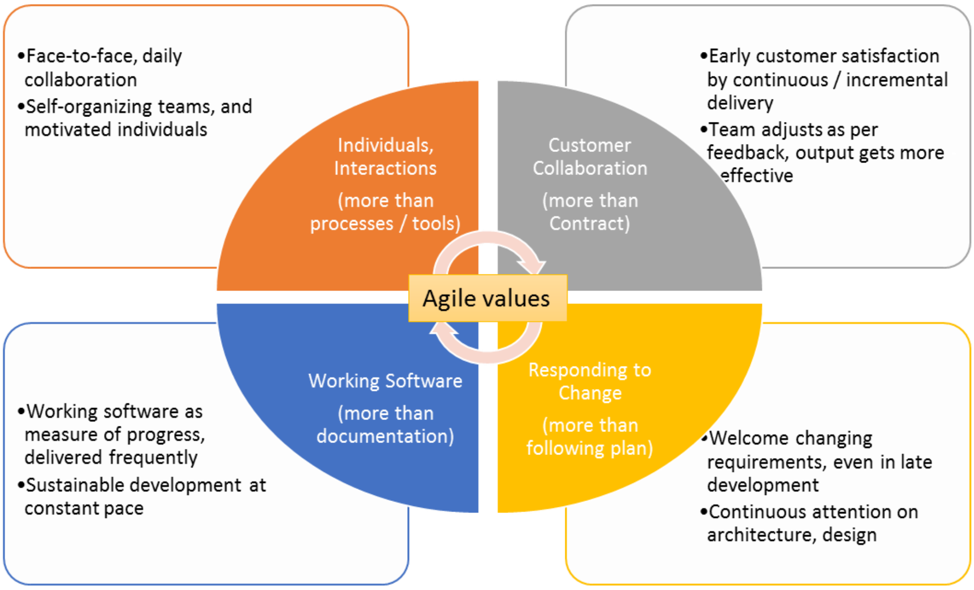
Flavours of Waterfall

Hybrid ("Agile + non-Agile") makes sense!!

Waterfall	Design Team	Design								
	Dev Team				Build					
	Test Team							Test		
Iterative Waterfall (sequential)	Design Team	Design 1			Design 2			Design 3		
	Dev Team		Build 1			Build 2			Build 3	
	Test Team			Test 1			Test 2			Test 3
Iterative Waterfall (overlapping)	Design Team	Design 1	Design 2	Design 3						
	Dev Team		Build 1	Build 2	Build 3					
	Test Team			Test 1	Test 2	Test 3				

2.2 Agile Methodology

The Manifesto for Agile Software Development



- Face-to-face, daily collaboration
- Self-organizing teams, and motivated individuals

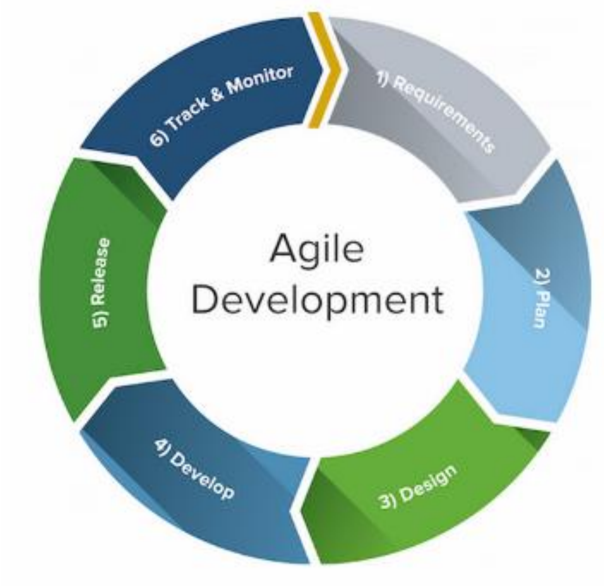
- Early customer satisfaction by continuous / incremental delivery
- Team adjusts as per feedback, output gets more effective

- Working software as measure of progress, delivered frequently
- Sustainable development at constant pace

- Welcome changing requirements, even in late development
- Continuous attention on architecture, design

Individuals, Interactions
(more than processes / tools)
Customer Collaboration
(more than Contract)
Agile values
Working Software
(more than documentation)
Responding to Change
(more than following plan)

Agile software development is based on an incremental, iterative approach. Instead of in-depth planning at the beginning of the project. In Agile methodologies, leadership encourages teamwork, accountability, and face-to-face communication. Business stakeholders and developers must work together to align the product with customer needs and company goals.



ADVANTAGES	DISADVANTAGES
+ Better collaboration-communication + Increased customer's satisfaction + Responsive to market changes / customer's demands + Suited for projects where end-goal is unknown + Faster, high-quality delivery and continuous improvement	− Difficult to plan fixed time projects − Can be intense for developers − Strong requirement management processes needed − Final product can be very different

Flavours of Agile

Agile principles support the definition and continuing evolution of many software development methods, two of the most widely used being Scrum and Kanban.

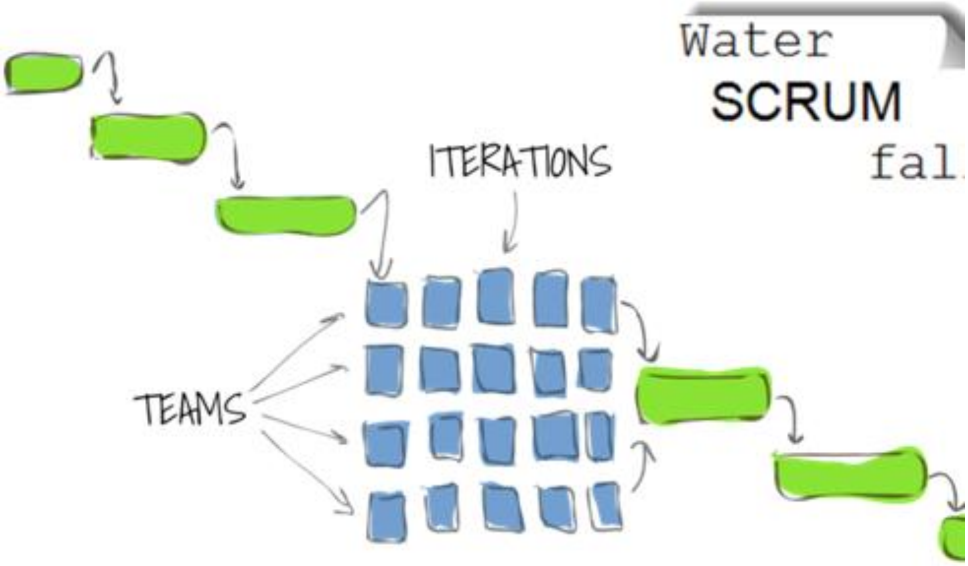
SCRUM	KANBAN
✚ Scrum is an iterative and incremental agile software development framework. ✚ Scrum is designed for teams of three to nine developers who break their work into one-week to maximum four-week cycles, called "sprints", check progress daily in 15-minute stand-up meetings, and deliver workable software at the end of every sprint. ✚ Scrum is for teams that expect some sort of stability during the sprint ✚ Approaches to coordinating the work of multiple scrum teams in larger organizations include Large-Scale Scrum and Scrum of Scrums.	✚ Kanban is a method for visualizing the flow of work, in order to balance demand with available capacity and spot bottlenecks. ✚ Team members pull work as capacity permits, rather than work being pushed into the process when requested. ✚ Kanban suits teams that may need to quickly change their focus based on inputs from outside (customers, users, etc.) ✚ Although originated in lean manufacturing (inspired by the Toyota Production System), it is used in software development and can be applied to any type of service.

3. We live in a HYBRID ERA

As seen in previous section, both the methodologies, Waterfall & Agile, are suited to specific situation. Also, with industries undergoing rapid/ disruptive changes, project management methods are changing as rapidly. Hybrid methods enable smooth transition in such changing times.

3.1 Hybrid Agile / Water-Scrum-Fall / Scrum But

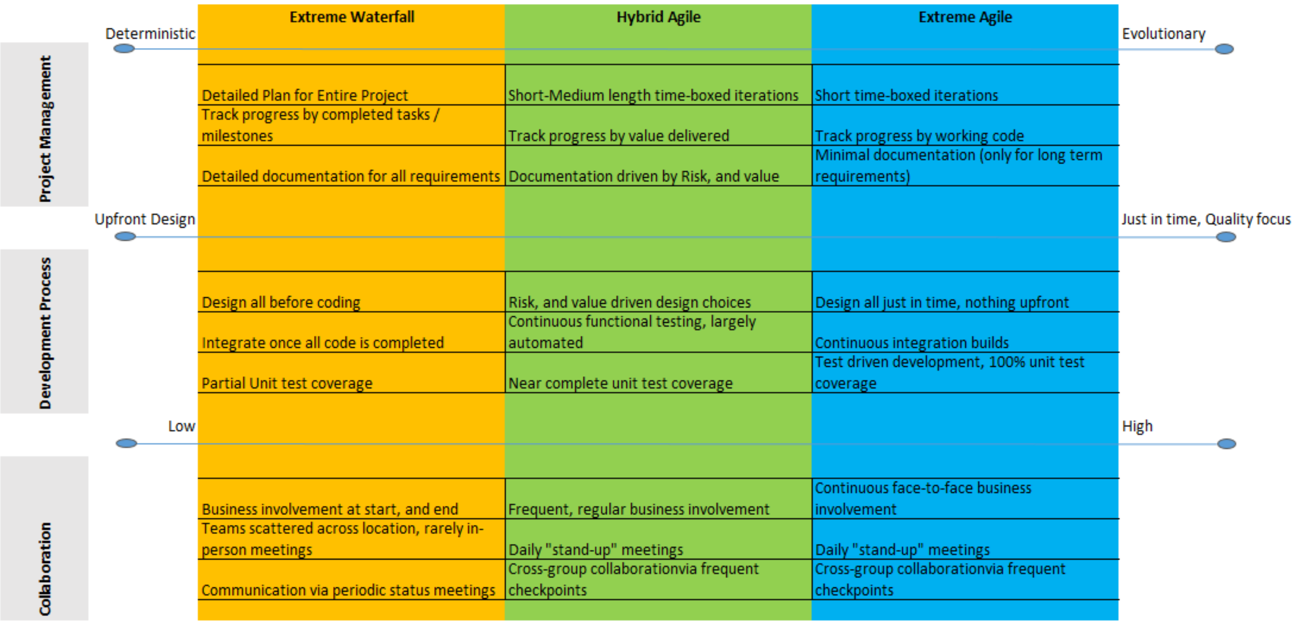
Simply put, a Hybrid Agile approach is one that mixes the Non-Agile (plan-driven) principles / practices with Agile (adaptive) principles / practices in the right proportions to fit a given situation.



Hybrid has come into existence as a result of below scenarios:

- **Hybrid as Fit-For-Purpose**: The agile and waterfall models in isolation have worked impeccably in some cases and have failed miserably in others. Hence, organizations have opted for "pick & choose" approach: For projects that have a lower risk profile, use Plan-Driven approaches to look for lower costs. For higher risk projects, use Iterative techniques to repeat activities until issues are revealed and resolved. For projects needing aggressive delivery, Incremental techniques will deliver something sooner, to ensure customer engagement. Finally, in order to navigate complex environments, Agile techniques may have a higher initial overhead, but it might be worth it for the overall outcomes. Mixing these together in the right way can fit the context better than just narrowly using only one of them.
- **Hybrid as Transition-to-Agile**: Many teams are not able to make the switch to Agile ways of working overnight. The larger the organization, the more moving parts, the longer it will take to shift. For an organization enthused with Plan-Driven world for several years, then Agile methods will look and feel very different. As a result, the initial foray into the Agile world will be an amalgamation of both.

3.1.1.1 Hybrid vis-à-vis Agile vis-à-vis Waterfall



		Extreme Waterfall	Hybrid Agile	Extreme Agile	
Project Management	Deterministic				Evolutionary
		Detailed Plan for Entire Project	Short-Medium length time-boxed iterations	Short time-boxed iterations	
		Track progress by completed tasks / milestones	Track progress by value delivered	Track progress by working code	
		Detailed documentation for all requirements	Documentation driven by Risk, and value	Minimal documentation (only for long term requirements)	
Development Process	Upfront Design				Just in time, Quality focus
		Design all before coding	Risk, and value driven design choices	Design all just in time, nothing upfront	
		Integrate once all code is completed	Continuous functional testing, largely automated	Continuous integration builds	
		Partial Unit test coverage	Near complete unit test coverage	Test driven development, 100% unit test coverage	
Collaboration	Low				High
		Business involvement at start, and end	Frequent, regular business involvement	Continuous face-to-face business involvement	
		Teams scattered across location, rarely in-person meetings	Daily "stand-up" meetings	Daily "stand-up" meetings	
		Communication via periodic status meetings	Cross-group collaborationvia frequent checkpoints	Cross-group collaborationvia frequent checkpoints	

4. Industry Shift

4.1 Analyst Outlook

FORRESTER

"Water Scrum Fall Is The Reality Of Agile For Most Organizations Today"

Organizations are adopting Agile through a combination of bottom-up adoption and top-down change. But the reality of Agile adoption has diverged from the original ideas described in the Agile Manifesto, with many adoptions resembling a water-Scrum-fall. This happens in part because Agile adoption has been practitioner-led, leading teams to focus on domains they can influence, mainly the team itself. Areas outside of their control, such as business analysis and release management, continue to follow more-traditional approaches, meaning that Scrum adoption is limited to the development-team level. Compliance requirements are another factor driving hybrid approaches, as they call for strong governance processes before and after development.

GARTNER

Many clients have begun to take Gartner's advice to explicitly segment their work. "Bimodal IT: How to Be Digitally Agile Without Making a Mess" lays out a framework for understanding this segmentation. This matters a lot when it comes to choosing a method.

- Mode 1 describes the more traditional work that application organizations do. The presumed goal is to provide a clearly defined set of functionality in a stable, secure and predictable way. It's tempting, therefore, to presume that waterfall would work well here, since that's the method that's been traditionally employed. However, reading the data above and based on our conversations with clients, it can be delivered by any recommended method.

- Mode 2 describes work that is more exploratory, is less known and demands quick results. Mode 2 requires constant feedback from the project stakeholders in order to evolve to the best solution to a problem, constrained by the time and money available. Agile development is an obvious choice here.

There are also many hybrid methods, most of which combine agile and iterative practices. They're commonly referred to as "wagile," "scrumfall" or "waterscrum."

4.2 Competitor Landscape

Accenture: MULTI Speed Methodology

Multi Speed Methodology combines best practices from different methodologies and enables the cultural shift to capitalize the outcomes.

- Scoping: Project requirements are analyzed to determine which elements will be channelled towards New/Agile and which towards Traditional/Waterfall.
- Estimation: Effort estimation can be measured in story points for New/Agile and in man-days for Traditional/Waterfall.
- Planning: Execution will be planned in terms of sprints for New/Agile and separate releases for Traditional/Waterfall.
- Scope of Work: Master plan of detailed requirements sorted by channel together with time and cost estimates.

Advancements in technology are pushing Communications organizations to become more adaptive to continuously evolving customer needs. New delivery methodologies and particularly Agile are accordingly gaining more ground at the expense of Waterfall. However, the transition comes at a price: discipline, a cultural shift and strong organizational commitment are prerequisites for its success. By combining both next generation and traditional methodologies and allowing people time to adjust to the cultural change, management can score more rewarding wins.

TCS: SAFe®

TCS has joined the Scaled Agile Partner program as a Gold Partner with Scaled Agile, Inc. Large enterprises' have an ever evolving need to better business and IT alignment, while ensuring improved time to market, overall productivity and quality. TCS supports these enterprises as they embark on an agile journey by providing Agile Advisory Services – often leveraging SAFe®.

With more than 200 TCS SAFe® Certified consultants, TCS' Agile Practice offers end-to-end enablement and transformation with offerings like Readiness Assessment, role-based training at the team, program and portfolio level, and coaching to make the transformation sustainable.

Infosys: Global Agile

Our Agile enterprise services are designed to breakdown cultural barriers, equip colleagues to think and operate differently, and help you rework organizational structures. In practice, we tend to work on these changes incrementally and in parallel, using key projects as vehicles for change.

Our accelerated techniques include:

- Design Thinking workshops to set direction
- Catalyst teams to prototype ideas and take them to market very rapidly
- Design factories to develop design and test concepts
- Agile development coaching and training to equip colleagues to employ alternative approaches
- Change management consulting to help people to change
- Large-scale Design Thinking training to develop creative confidence within the IT organization and to bring together business, IT and operational teams around a new customer-centric agenda
- New working methods and product development approaches

With an objective to achieve a model which derives benefits of the distributed development environment for traditional co-located agile, Infosys Global Agile allows for business value focused rapid application delivery; striving to achieve a perfect balance of cost, effort, risk and time to market. Infosys Global Agile helps you address growing business demands, end to end IT agility, and value delivery through IT applications. Moving away from the traditional, Infosys Global Agile offering enables execution of agile projects from distributed locations and deliver high-quality applications, cost effectively.

Infosys Virtual Scrum - Our award winning distributed agile execution platform, 'Agile tool of the year, 2015' by Unicom, Infosys Virtual Scrum helps organizations address distributed agile challenges to deliver successful agile programs.

Cognizant: Agile – Daikibo™ Delivery

Since its inception, Agile methodologies have predominantly been focused on small software development projects executed at a single collocated environment. Today, as Agile begins to move rapidly into the mainstream conscious of Fortune 500 companies, we find ourselves exploring ways to extend the benefits of Agility to not only large-scale but also to distributed environments. This is not only feasible but even practical today.

Created by Cognizant, Daikibo^{TM} is our in-house Agile execution framework. It is built on the shoulders of several frameworks: Extreme Programming (XP), Scrum and Kanban, as well as the Agile and Lean philosophies of software. It addresses the "how to" questions that arise around large scale and/or distributed engagements based on a producer-consumer model.

The services that we offer are:

- Agile-enabled Delivery
- Distributed Agile and Managed Services
- Agile for Digital

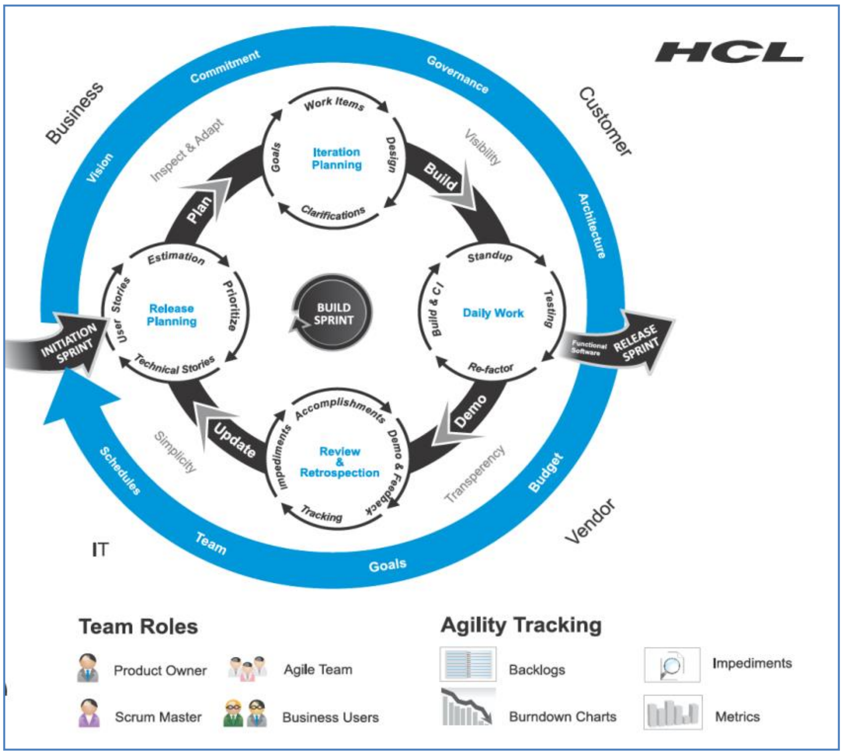
Deloitte: DE FACTS II Hybrid-Agile Framework

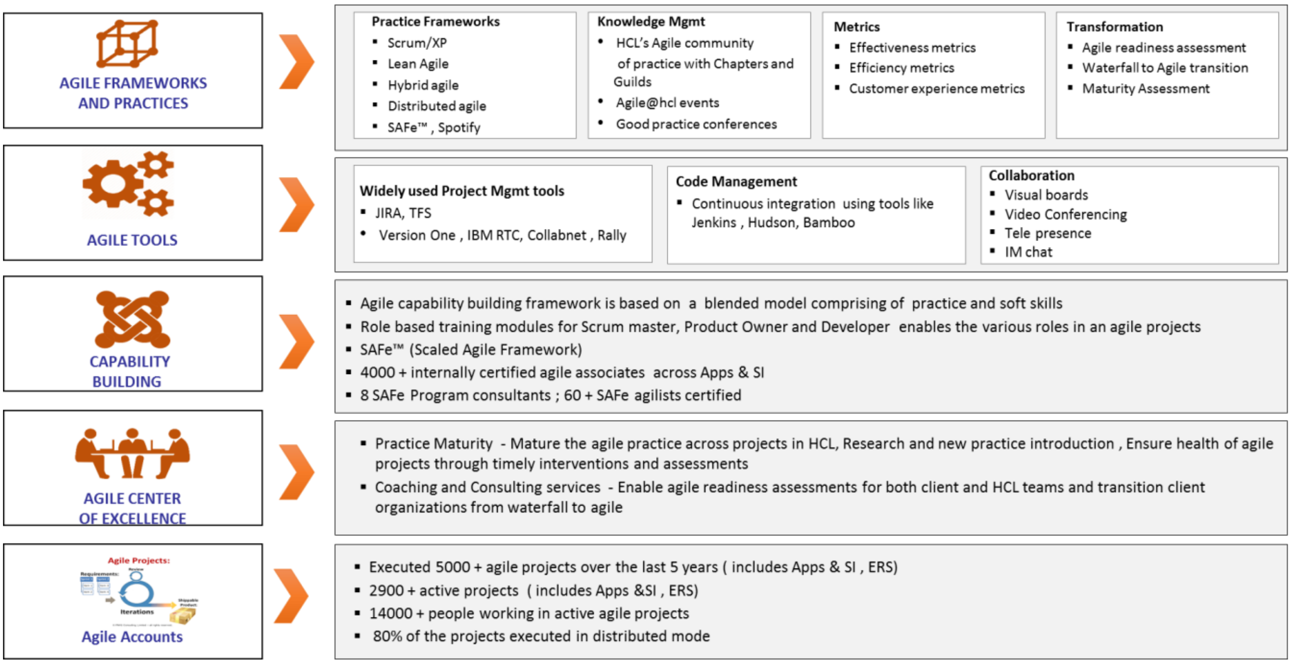
Our Hybrid-Agile process is a modular, flexible approach which applies Agile techniques to iterative development and test cycles.

- Discovery - Planning and Analysis: During the planning and analysis, the foundation for the project is cemented. An overarching master plan is created, themes and epics are mapped on a roadmap and the product backlog is created with initial user stories that are prioritized.
- Design & Build – Sprint: A sprint represents a four week (working days) period in which the team works on User Stories from the Sprint Backlog a subset of Product Backlog with the goal of producing working software.
- Design & Build – Initial Integration Testing: After User Stories are accepted (have met acceptance criteria) at the end of a Sprint, they will migrate to an Integration environment for initial integration testing.
- The Release – Incremental UAT: A release is a grouping of User Stories that represents a system functionality (e.g. Program Involvement). A Release is ready for Incremental UAT once all User Stories in that Release have passed Integration Testing and are considered "Done" .

Agile @ HCL

Agile practice framework depicted in the snapshot below embodies the hybrid model and is complemented by articulation of the various practices, checklists and templates classified into the Initiation Sprint, Build Sprint and the Release Sprint.





AGILE FRAMEWORKS AND PRACTICES

Practice Frameworks
- Scrum/XP
- Lean Agile
- Hybrid agile
- Distributed agile
- SAFe™ , Spotify

Knowledge Mgmt
- HCL's Agile community of practice with Chapters and Guilds
- Agile@hcl events
- Good practice conferences

Metrics
- Effectiveness metrics
- Efficiency metrics
- Customer experience metrics

Transformation
- Agile readiness assessment
- Waterfall to Agile transition
- Maturity Assessment

AGILE TOOLS

Widely used Project Mgmt tools
- JIRA, TFS
- Version One , IBM RTC, Collabnet , Rally

Code Management
- Continuous integration using tools like Jenkins , Hudson, Bamboo

Collaboration
- Visual boards
- Video Conferencing
- Tele presence
- IM chat

CAPABILITY BUILDING

- Agile capability building framework is based on a blended model comprising of practice and soft skills
- Role based training modules for Scrum master, Product Owner and Developer enables the various roles in an agile projects
- SAFe™ (Scaled Agile Framework)
- 4000 + internally certified agile associates across Apps & SI
- 8 SAFe Program consultants ; 60 + SAFe agilists certified

AGILE CENTER OF EXCELLENCE

- Practice Maturity - Mature the agile practice across projects in HCL, Research and new practice introduction , Ensure health of agile projects through timely interventions and assessments
- Coaching and Consulting services - Enable agile readiness assessments for both client and HCL teams and transition client organizations from waterfall to agile

Agile Accounts

- Executed 5000 + agile projects over the last 5 years (includes Apps & SI , ERS)
- 2900 + active projects (includes Apps &SI , ERS)
- 14000 + people working in active agile projects
- 80% of the projects executed in distributed mode

5. Summary

While principles of Agile development are fairly novel, when it comes to practice, certain aspects of them get ignored / left out. Reasons could be many, e.g., Either age/maturity of firm becomes a barrier as having own ways of doing things, OR that the project team are not co-located / close with business team and also business team can't commit itself throughout project lifecycle etc. In the course, it shifts towards Hybrid Agile.

Hybrid Agile allows moving forward incrementally, while organisational hurdles get solved in parallel. Good use-cases for Hybrid Agile may typically be:

- Product Development Teams: Hardware (Waterfall); Software (Agile)
- Software Product Development: Planning / Design / Release (Waterfall); Development & Testing (Agile)
- Software (n-tier) application development: Frontend User Experience (Agile/Scrum); Backend (Waterfall)

In the end, the best methodology for a company or project can only be decided by skilled project leaders that understand both the scope of the project, and the nature of their team. Every project has different needs, every client/ industry is unique… goal is not merely "Doing Agile", but to create value for businesses. So, a better strategy is to take a step back and think about which approach would be best for where we are, and what we want to achieve. In essence, organizations need to endure through this cultural shift while embracing continous change.