OKR

vs.

AGILE/SCRUM

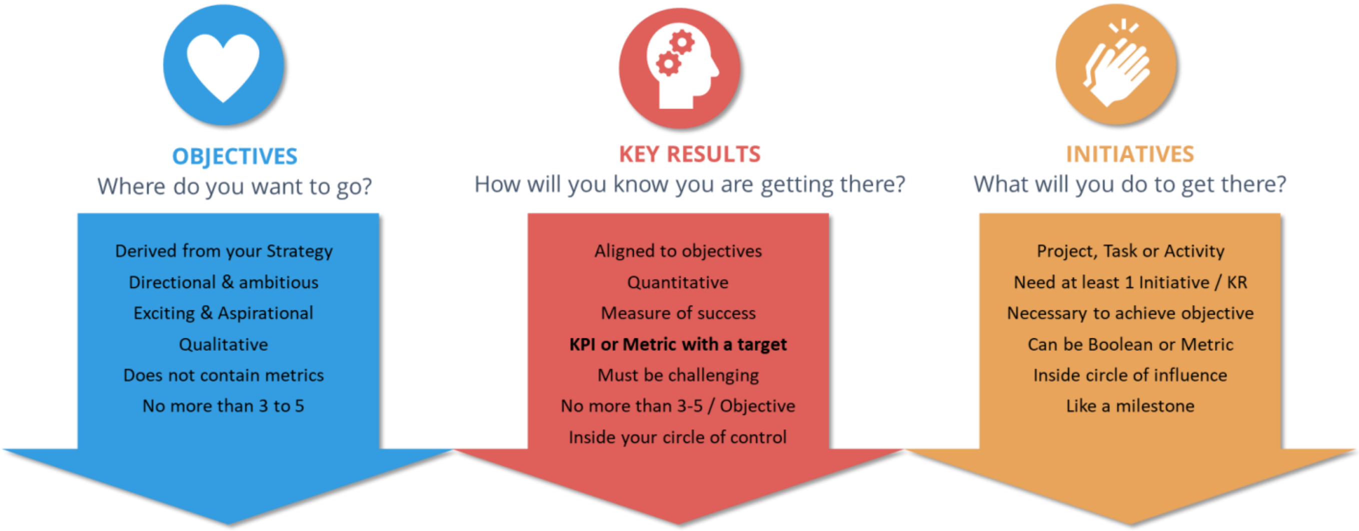
TABLE OF CONTENTS

WHAT IS OKR?

OKR is a goal management and execution framework popularly adopted by companies such as Spotify, Twitter, Airbnb, LinkedIn, Samsung, Microsoft, Amazon, and also used by companies such as Colgate, Walmart, Anheuser-Busch, Target, Gap, GE, Dun and Bradstreet, and ING Bank. Originally developed at Intel, it is most often cited as the go-to tool used by Google.

Objectives & Key Results (OKR) is a goal management & execution system that contains no more than 3-5 Objectives & no more than 3-5 Key Results for each objective. It has 3 major components viz. Objectives, Key Results and Initiatives.

Objectives answer the question, where do you want to go? Key results answer the question, how will you know you are getting there and how will you measure success? Initiatives answer the question, what will you do to get there?



OBJECTIVES
Where do you want to go?

Derived from your Strategy
Directional & ambitious
Exciting & Aspirational
Qualitative
Does not contain metrics
No more than 3 to 5

KEY RESULTS
How will you know you are getting there?

Aligned to objectives
Quantitative
Measure of success
KPI or Metric with a target
Must be challenging
No more than 3-5 / Objective
Inside your circle of control

INITIATIVES
What will you do to get there?

Project, Task or Activity
Need at least 1 Initiative / KR
Necessary to achieve objective
Can be Boolean or Metric
Inside circle of influence
Like a milestone

WHAT IS AGILE?

Agile is a mindset defined by the Agile Manifesto Values, guided by the Agile Manifesto Principles, and enabled by various practices. Agile approaches and methods are an umbrella term consisting of several frameworks and methods - SCRUM is the most popular among them.

Agile itself remains a sub-set of LEAN like Kanban. Like SCRUM, Agile also consists of other popular and often used methods like XP, Crystal, DSDM, FDD, AUP, SCRUMBAN, etc.

The Four Agile Values

1. Individuals and Interactions Over Processes and Tools
2. Working Software Over Comprehensive Documentation
3. Customer Collaboration Over Contract Negotiation
4. Responding to Change Over Following a Plan

The Twelve Agile Manifesto Principles

1. Customer satisfaction through early and continuous software delivery.
2. Accommodate changing requirements throughout the development process.
3. Frequent delivery of working software.
4. Collaboration between the business stakeholders and developers throughout the project.
5. Support, trust, and motivate the people involved.
6. Enable face-to-face interactions.
7. Working software is the primary measure of progress.
8. Agile processes to support a consistent development pace.
9. Attention to technical detail and design enhances agility.
10. Simplicity.
11. Self-organizing teams encourage great architectures, requirements, and designs.
12. Regular reflections on how to become more effective.

WHAT IS SCRUM?

Scrum, a method within Agile, is a lightweight framework that helps people, teams and organizations generate value through adaptive solutions for complex problems.

In a nutshell, Scrum requires a Scrum Master to foster an environment where:

1. A Product Owner orders the work for a complex problem into a Product Backlog.
2. The Scrum Team turns a selection of the work into an Increment of value during a Sprint.
3. The Scrum Team and its stakeholders inspect the results and adjust for the next Sprint.
4. Repeat

Scrum is founded on empiricism and lean thinking. Empiricism asserts that knowledge comes from experience and making decisions based on what is observed. Lean thinking reduces waste and focuses on the essentials.

Scrum combines four formal events for inspection and adaptation within a containing event, the Sprint. These events work because they implement the empirical Scrum pillars of transparency, inspection, and adaptation.

- Transparency
- Inspection
- Adaptation

Scrum Values

Successful use of Scrum depends on people becoming more proficient in living five values:

Commitment, Focus, Openness, Respect, and Courage

Scrum Team

The fundamental unit of Scrum is a small team of people, a Scrum Team. The Scrum Team consists of one Scrum Master, one Product Owner, and Developers. Within a Scrum Team, there are no sub-teams or hierarchies. It is a cohesive unit of professionals (no more than 5 to 7) focused on one objective at a time, the Product Goal.

Scrum Events

The Sprint is a container for all other events. Each event in Scrum is a formal opportunity to inspect and adapt Scrum artifacts. These events are specifically designed to enable the transparency required. Failure to operate any events as prescribed results in lost opportunities to inspect and adapt. Events are used in Scrum to create regularity and to minimize the need for meetings not defined in Scrum.

Sprint Planning

Sprint Planning initiates the Sprint by laying out the work to be performed for the Sprint. This resulting plan is created by the collaborative work of the entire Scrum Team.

Daily Scrum

The purpose of the Daily Scrum is to inspect progress toward the Sprint Goal and adapt the Sprint Backlog as necessary, adjusting the upcoming planned work.

Sprint Review

The purpose of the Sprint Review is to inspect the outcome of the Sprint and determine future adaptations.

Sprint Retrospective

The purpose of the Sprint Retrospective is to plan ways to increase quality and effectiveness.

Scrum Artifacts

Scrum's artifacts represent work or value. They are designed to maximize transparency of key information. Thus, everyone inspecting them has the same basis for adaptation.

Each artifact contains a commitment to ensure it provides information that enhances transparency and focus against which progress can be measured:

- For the Product Backlog it is the Product Goal.
- For the Sprint Backlog it is the Sprint Goal.
- For the Increment it is the Definition of Done.

WHAT'S SIMILAR BETWEEN OKRS & SCRUM?

Fundamentally, OKRs and SCRUM are different. However, they fit well with each other on certain core principles.

1. **Principles of Transparency, Inspection & Adaptation.**
   Both OKRs and SCRUM have this principle in common. In that, they require elements of transparency that enable inspection. To help with inspection, Scrum provides cadence in the form of its five events. OKRs, through transparency provide inputs for teams and members to review progress and pivot based on changes.

2. **Frequent Reviews & Rigor**
   Regular time-based reviews are the cornerstone of both OKRs and SCRUM. Both methods encourage a certain amount of religious rigor in its implementation that leads to dedicated practice, a culture of accountability and cross-functional interdependencies.

3. **Clarity in Measuring Success**
   Being clear of what success looks like is one of the other commonalities between OKRs and SCRUM. While SCRUM has its 'Definition of Done', OKRs use Key Results that contain a metric/KPI with a target (how much and by when), which defines what success looks like.

4. **Orchestrators**
   OKRs and Scrum are similar in that both frameworks need a person dedicated to their implementation, a "Scrum Master" or an "OKR Coach/ OKR Shepherd/OKR Champion". Both have clearly defined roles, and their responsibility is to ensure teams stick to the frameworks.

WHAT'S DIFFERENT BETWEEN OKRS & SCRUM?

OKRs and SCRUM are distinctly different in several ways.

1. **Usage**
   Agile/SCRUM is typically used for software development while OKRs are used to 'call out' focused priorities of the organisation and bring exponential growth through aspirational goals. OKR brings to the forefront the overall picture, while Scrum focuses on micro-management.

2. **Cycles**
   Agile/SCRUM has shorter cycles focused on product development in the form of Sprints. Each sprint needs to last for 1 month. OKRs run in annual and quarter cycles and these too are quite flexible. Within each quarter cycle, cadence reviews may vary from weekly, fortnightly to even monthly in some cases.

3. **Application**
   OKRs can be applied to organisations, teams and individuals alike. There is no pre-requisite on the size and number here. Agile/SCRUM need necessarily have a team and each team cannot be more than 5 to 7 members.

4. **Role of Members**
   SCRUM teams have pre-defined roles like the Product Owner, Developers and SCRUM Master. With OKRs, there is no such requirement and teams are formed freely based on the purpose and focus area.

5. **Deliverables**
   Scrum defines output and is a way to get activities/initiatives completed. OKRs are there to make people focus on the most important things to be done and what value will be generated through 'outcome'. Objectives defines the Why, which is connected to Mission/Values. KRs define what has to be achieved to achieve Objective completion. For every Key Results there are initiatives (projects, tasks or activities). These activities can be defined via Sprint/Scrum. Remember, OKRs always help to see the bigger picture.

CAN OKRS & SCRUM WORK TOGETHER?

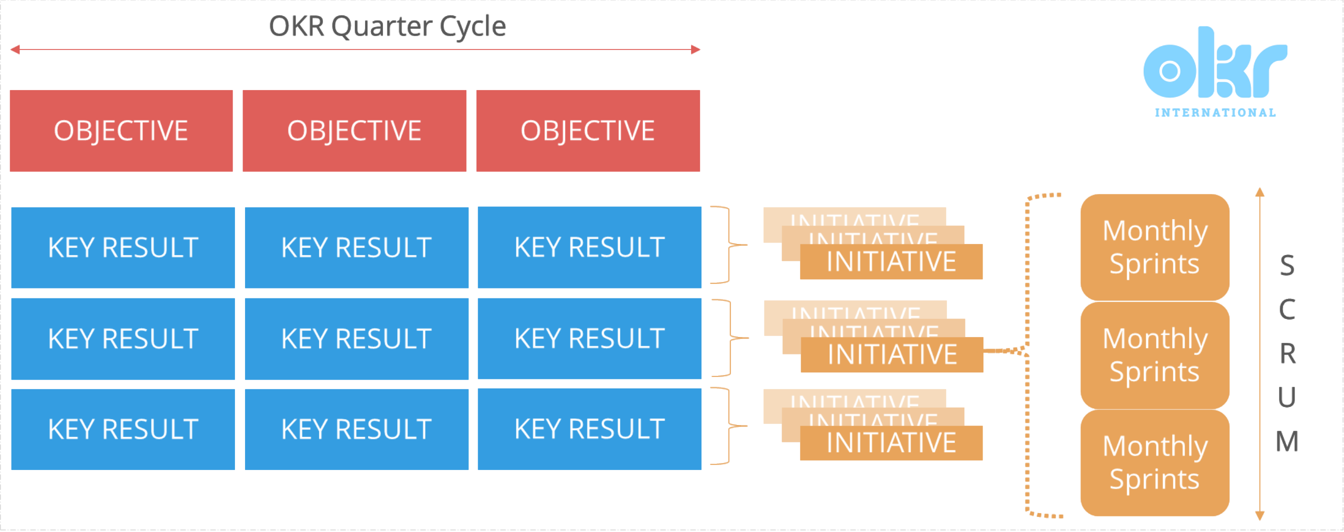
Most definitely, Yes! There are some caveats to consider before you organize yourself to combine OKRs and SCRUM.

Since Agile/SCRUM is used for product development and other such complex projects, combining it with OKRs means using OKRs with the realm of such projects alone. What does that mean? It means if the organisation has OKRs beyond the product development scope, SCRUM may not fit in with other functional or divisional OKRs.

Clarity on the scope of OKRs and SCRUM for all members within the OKR/SCRUM teams is critical. So where do you begin and how do you end?

Within the nuances of OKRs, Objectives lay out the larger goal and its connection to the overall purpose & strategy. Key Results help measure progress and define success parameters. Initiatives are defined for every key result. They are essentially projects, tasks or activities that need to be undertaken in order to achieve the Objectives. Sprints are a perfect fit within Initiatives. And since sprints work mostly in monthly cycles, the Sprint Goals can coincide well with achieving outputs that deliver the overall OKR outcome at the end of each quarter cycle.

The OKR Champion/Coach may collaborate with the SCRUM Master at the start of each cycle to help them understand the larger picture and the 'Why' behind the OKRs. Defining OKRs for a cycle will always come first, therefore. Once the outcomes expected are clear, the SCRUM Master can then work their teams to define Sprint goals and the product backlog that integrate and help achieve the OKRs. This is what we call forest for the trees.


OKR Quarter Cycle
OBJECTIVE
OBJECTIVE
OBJECTIVE
KEY RESULT
KEY RESULT
KEY RESULT
KEY RESULT
KEY RESULT
KEY RESULT
KEY RESULT
KEY RESULT
KEY RESULT
INITIATIVE
INITIATIVE
INITIATIVE
INITIATIVE
INITIATIVE
INITIATIVE
INITIATIVE
INITIATIVE
INITIATIVE
Monthly Sprints
Monthly Sprints
Monthly Sprints
SCRUM
okr INTERNATIONAL