



Scrum.org™
The Home of Scrum

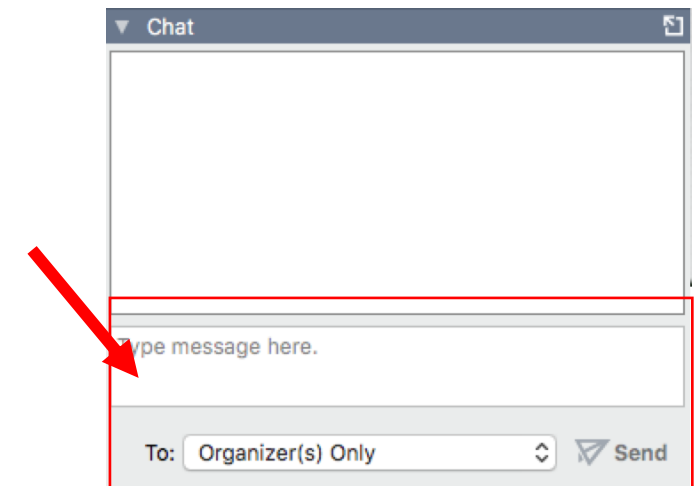
A CYCLE TIME JOURNEY

164 TO ~~86~~ DAYS IN ~~69~~ MONTHS

Adrian Galarza & Ravi Verma

Quick Guidelines

- Your microphones will be muted throughout
- Please ask questions!
- Type questions into the webinar questions box:
- Type comments into the webinar comments box:



Who is Scrum.org



Ken Schwaber
Scrum.org Founder,
Chairman and
Co-creator of Scrum



Consistent

Global

Community



ADRIAN GALARZA

- Scrum master, change agent, former consultant.
- Experienced in process improvement, benchmarking, operational efficiency.
- Committed to exceeding goals and driving cost reduction through data analysis and team development.
- Track record of building team culture committed to collaboration, accountability, data-driven decision making and delivering results for customers.
- Diverse experience...
 - Held various leadership roles in fulfillment centers for the world's largest e-commerce retailer.
 - Leading teams of security specialists in a developing country with high threat profile
 - Teams of government employees and contractors in US diplomatic mission abroad
 - Teams of technical employees in a production lab





RAVI VERMA


- Founder & Org Whisperer at [SmoothApps](https://smoothapps.com)
- PST, EBM Consultant, Class Clown @ Scrum.org
- 20+ Years in I.T. Delivery:
 - Networking • Telecom • Security • Banking • Real Estate •
 - Healthcare • Travel • Sports • Media • Agrotech • Personals •
 - 10 – 10K employees • 1 to 350 Agile Teams •
- Co-founder [Agile For Patriots](#), [Agile DevOps DFW](#)
- Creator of [Sabotagile Manifesto](#) and [Sabotagile Principles](#)
- Co-creator of [Software Code of Ethics](#), [Scrum Pulse Webinars](#) & [Scrum Tapas Videos](#)
- Story-teller: [Agilato Video Series](#), SmoothApps [Agile Blogs](#)



FLOW

1. INTENT
2. RISKS
3. APPROACH
4. ACHIEVEMENTS
5. WISDOM
6. RECOMMENDATION
7. REFERENCES
8. Q&A

FLOW

1. INTENT ← 
2. RISKS
3. APPROACH
4. ACHIEVEMENTS
5. WISDOM
6. RECOMMENDATION
7. REFERENCES
8. Q&A

INTENT

- **AS A:** Webinar attendee
- **I WANT TO:** Get at least one actionable insight on reducing cycle time
- **SO THAT:** I can apply what I learned in my context
To create a measurable improvement in cycle time
Within 30 days or less

CYCLE TIME

of calendar days
A Scrum Team needs

To take a *“Sprint Ready”* PBI
To go from Active to *Deployed*
& *Useable* in Production

SO THAT...


They reduce
the # of calendar days needed
to observe market response to a PBI,
learn and *adjust future investment*

BECAUSE WHY...

*“It is not the most intellectual
of the species that survives;
it is not the strongest that survives;
but the species that survives
is the one that is able best
to **adapt and adjust**
to the changing environment
in which it finds itself.”*

~~Charles Darwin~~ Leon C. Megginson

FLOW

1. INTENT
2. RISKS 
3. APPROACH
4. ACHIEVEMENTS
5. WISDOM
6. RECOMMENDATION
7. REFERENCES
8. Q&A

RISKS

The temptation to
robotically copy-paste practices
from this webinar
to your context

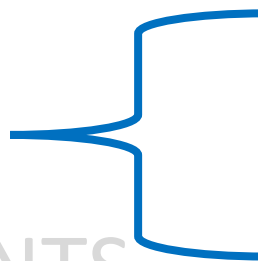
BECAUSE WHY...

It might cause
needless suffering
in the name of
Agility & Scrum

OK, SO WHAT...

1. Listen for Intent
2. Understand the Context
3. Look beneath the surface
4. Adapt to your context

APPROACH

1. INTENT
 2. RISKS
 3. **APPROACH**
 4. ACHIEVEMENTS
 5. WISDOM
 6. RECOMMENDATION
 7. REFERENCES
 8. Q&A
- 
- What happened before and around
 - What happened within

APPROACH

1. INTENT

2. RISKS

3. **APPROACH**

4. ACHIEVEMENTS

5. WISDOM

6. RECOMMENDATION

7. REFERENCES

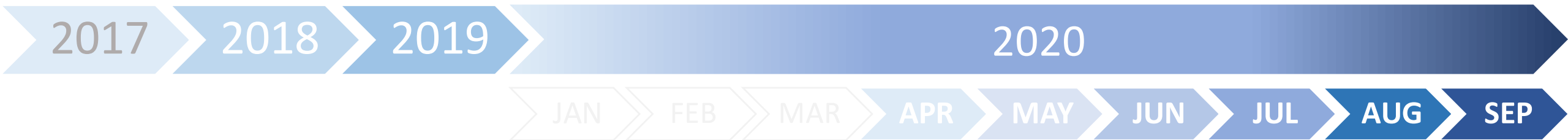
8. Q&A



























• What happened before and around



• What happened within

4 YEAR JOURNEY



	2017	2018	2019	2020
COMMUNITIES	AGILE COP • ARCHITECTURE COP • SCRUM MASTER COP • SERVANT LEADER COP			
EVENTS	SOS • BUSINESS ROADMAP ALIGNMENT • LUNCH & LEARN • PATH TO AGILITY • EXECUTIVE REVIEW			
PLAYBOOKS	• SCRUM EVENTS & ACTIVITIES • SCRUM TEAM QUICK-LAUNCH/ REBOOT • • PRODUCT OWNERSHIP • RELEASE PLANNING • EBM			
VALIDATION	 	  	   	   
TRAINING				
		 	 	 
		 		
FUNDAMENTALS	ETHICS • AGILE MANIFESTO & PRINCIPLES • SCRUM • NEXUS • EBM			


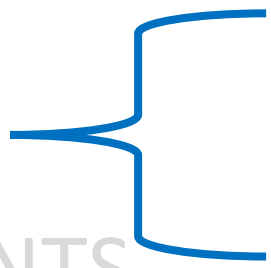
BEFORE WE BEGAN

- Grass-roots Scrum Adoption
- No full-time Scrum Master
- High WIP and cycle times
- Unfamiliar with self-organization

WHERE WE BEGAN

- Added a dedicated PO
- Added a full-time Scrum Master
- Increased focus on refinement

APPROACH

1. INTENT
 2. RISKS
 3. **APPROACH**
 4. ACHIEVEMENTS
 5. WISDOM
 6. RECOMMENDATION
 7. REFERENCES
 8. Q&A
- What happened before and around
 - **What happened within** 
- 

WHAT WE TRIED

- 4 sets of adjustments over 6 months...
 1. Stakeholders Collaboration
 2. Scrum Team Collaboration
 3. Technical Excellence
 4. Sprint Planning

WHAT WE TRIED

- 4 key sets of adjustments over 6 months...

1. **Stakeholders Collaboration**
2. Scrum Team Collaboration
3. Technical Excellence
4. Sprint Planning

What is the single biggest challenge in this area that increases cycle time?

STAKEHOLDERS COLLABORATION

WHAT WE FACED

1. High amount of rework and bugs delayed value creation. Additional AC were identified only after PBIs done.
2. Testing bottlenecks and inefficiencies delayed prod deployment. Untested changes in code base delayed multiple items.

WHAT WE TRIED

1. Increase business unit engagement by inviting additional relevant SMEs to refinement.
2. Increase alignment and agree on expectations with Business unit and its leaders around testing & feedback.

WHAT HAPPENED NEXT

1. A.C. were identified before starting on PBIs. Reduced rework and built the right thing, the first time around.
2. Testers and SMEs could align their schedules to be available to test. Resulted in shorter turn around times.

WHAT WE TRIED

- 4 key sets of adjustments over 6 months...

1. Stakeholders Collaboration
2. **Scrum Team Collaboration**
3. Technical Excellence
4. Sprint Planning

What is the single biggest challenge in this area that increases cycle time?

SCRUM TEAM COLLABORATION

WHAT WE FACED

1. Team required assistance from overburdened external teams, causing days or weeks of delays.
2. Refinement sessions often ran long and became less efficient. We were not always holding ourselves accountable.

WHAT WE TRIED

1. Cross-train team members in Java development and CI/CD pipelines.
2. Agree to stay within timebox in meetings; agree to stay engaged in meetings; turn on cameras; and update PBIs as A.C. evolved.

WHAT HAPPENED NEXT

1. Team reduced dependency on Java team and Architects and developers increased their skills.
2. Completed refinement on time. Fully engaged developers identified more blind spots and increased understanding of PBIs. PO and testers had up-to-date A.C. when testing.

WHAT WE TRIED

- 4 key sets of adjustments over 6 months...

1. Stakeholders Collaboration
2. Scrum Team Collaboration
3. **Technical Excellence**
4. Sprint Planning

What is the single biggest challenge in this area that increases cycle time?

TECHNICAL EXCELLENCE

WHAT WE FACED

1. High amount of rework and bugs
2. Testing bottlenecks and inefficiencies

WHAT WE TRIED

1. Use automation testing to shift from defect detection & fixing towards defect prevention.
2. Prioritize technical excellence, pay off tech debt, adjust forecast accordingly.

WHAT HAPPENED NEXT

1. Reduced huge amount of manual testing. Prevented several customer facing bugs.
2. Quality reviews reduced back and forth with business testers. Forecasts became much more reliable.

WHAT WE TRIED

- 4 key sets of adjustments over 6 months...

1. Stakeholders Collaboration
2. Scrum Team Collaboration
3. Technical Excellence
4. **Sprint Planning**

What is the single biggest challenge in this area that increases cycle time?

SPRINT PLANNING

WHAT WE FACED

1. Wrapping up majority of sprint PBIs in last 1-2 days and putting huge strain on testers
2. Complex, risky items were sometimes difficult to complete before the end of the sprint

WHAT WE TRIED

1. More frequent, small deployments within Sprint
2. Front-load risky PBI's during Sprint Planning and vertical slicing

WHAT HAPPENED NEXT

1. Produced a more even flow of work for testers throughout sprint
2. Saved enough time to identify and correct blockers before the sprint ended

RECAP - WHAT WE TRIED

STAKEHOLDERS COLLABORATION

1. Increase business unit engagement by inviting additional relevant SMEs to refinement.
2. Increase alignment and agree on expectations with Business unit and its leaders around testing & feedback.

SCRUM TEAM COLLABORATION

1. Cross-train team members in Java development and CI/CD pipelines.
2. Agree to stay within timebox in meetings; agree to stay engaged in meetings; turn on cameras; and update PBIs as A.C. evolved.


TECHNICAL EXCELLENCE

1. Use automation testing to shift from defect detection & fixing towards defect prevention.
2. Prioritize technical excellence, pay off tech debt, adjust forecast accordingly.

SPRINT PLANNING

1. More frequent, small deployments within Sprint
2. Front-load risky PBI's during Sprint Planning and vertical slicing

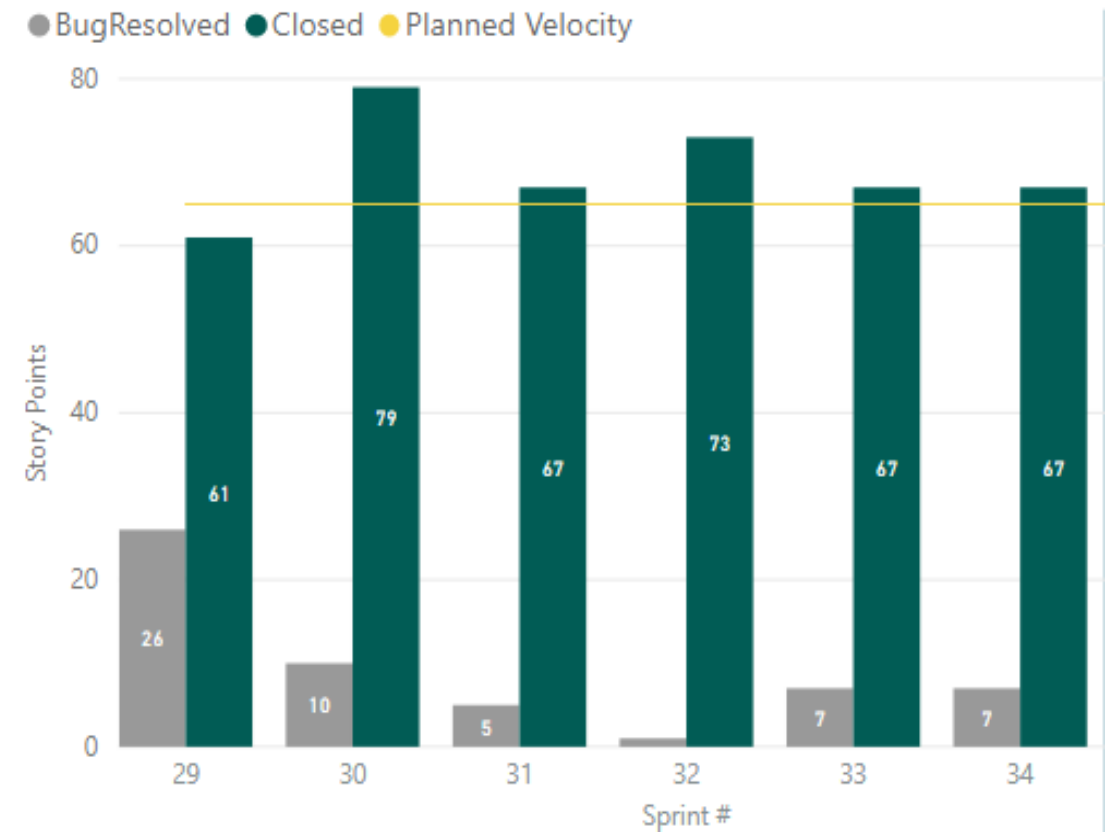
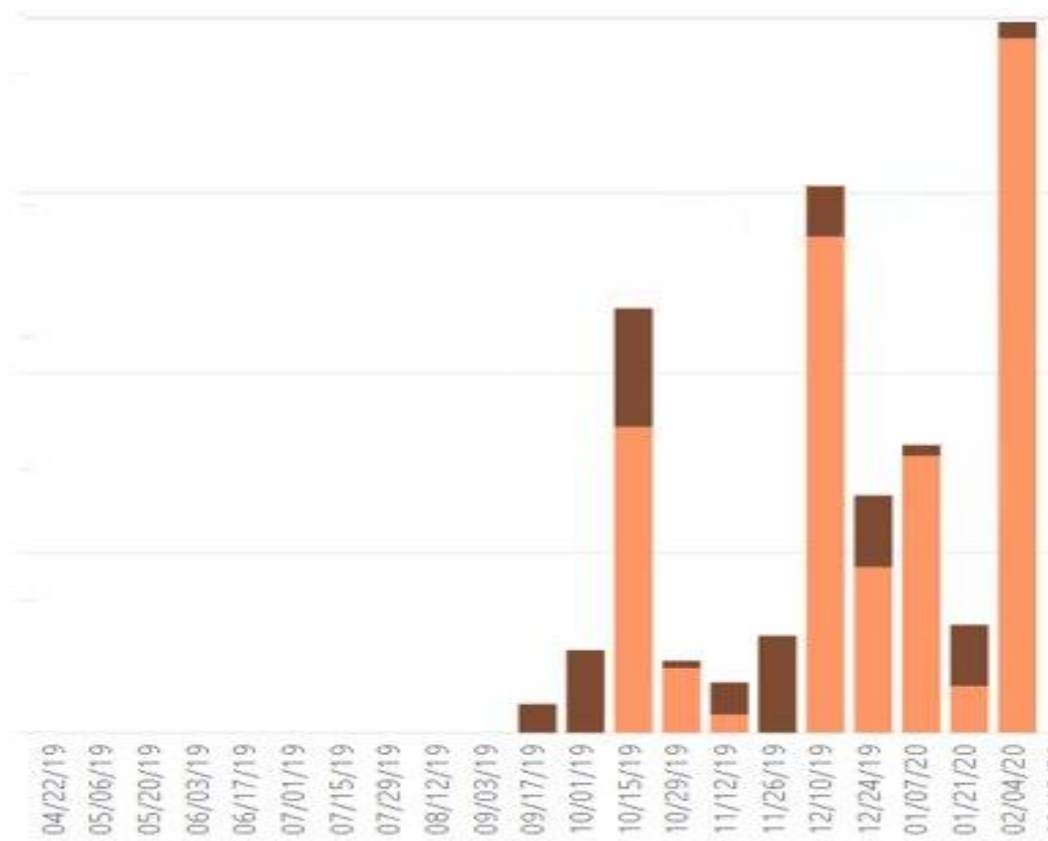
APPROACH

1. INTENT
2. RISKS
3. APPROACH
4. **ACHIEVEMENTS** 
5. WISDOM
6. RECOMMENDATION
7. REFERENCES
8. Q&A


WHAT WE ACHIEVED

1. Quality
2. Reliability
3. Responsiveness
4. Value
5. Delight
6. Trust
7. Morale

BEFORE... AND AFTER



APPROACH


1. INTENT
2. RISKS
3. APPROACH
4. ACHIEVEMENTS
5. **WISDOM** 
6. RECOMMENDATION
7. REFERENCES
8. Q&A

WHO SUPPORTED US

7 COMMUNITIES

1. PO, Business Leadership & Business Community
2. Dev Team – Humble, Hungry & Smart
3. Architects
4. Interdependent Teams
5. Servant Leader, Dir of Agile Enablement
6. CIO, C-Level Business Execs, CEO
7. Scrum Master


APPROACH

1. INTENT
2. RISKS
3. APPROACH
4. ACHIEVEMENTS
5. WISDOM
6. **RECOMMENDATION** 
7. REFERENCES
8. Q&A

WHAT WE RECOMMEND

1. Align leaders
2. Establish safety to ~~fail~~ *learn*, encourage learning
3. Retros and continuous improvement
4. Enable hungry team members
5. Start with the why, not with the how
6. Enable culture, practices will emerge

APPROACH

1. INTENT
2. RISKS
3. APPROACH
4. ACHIEVEMENTS
5. WISDOM
6. RECOMMENDATION
- 7. REFERENCES** 
8. Q&A

REFERENCES

Evidence-Based Management Guide

★★★★★ 4.6 from 10 ratings

Over the past two decades, most organizations have significantly increased the value gained from software through the adoption of the Scrum framework and Agile principles. The introduction of evidence-based management practices promises to deliver even further gain.

Software has moved far beyond being a business expense. It has evolved into a major investment to obtain specific goals in mind, goals that are often unobtainable in other ways. This change means that the need to manage software better has become an imperative. Evidence-Based Management for Software Organizations is the first useful method for transforming software from an expense into a profitable asset.

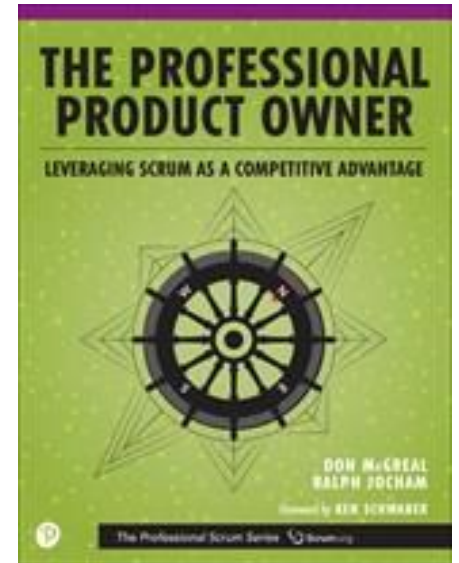
DOWNLOAD



The Coaching Habit Say Less, Ask More & Change the Way You Lead Forever

Michael Bungay Stanier

Wall Street Journal Bestseller
Over half a million copies sold



Evidence-Based Management (EBM) for Audacious Goals!

Agile-DevOps: DFW Chapter
Webinar, hosted by:
Ravi Verma
CEO, SmoothApps
Thursday, May 7, 2020
5:30pm CDT

Mark Noneman
Enterprise Agilist
SMN Consulting
Professional Scrum Trainer
Certified LeSS Practitioner
Accredited Kanban Trainer
SAFe Program Consultant

mark.noneman@smn.consulting
smn.consulting

Professional Scrum Trainer
Certified LeSS Practitioner
Accredited Kanban Trainer
SAFe Program Consultant

Evidence-Based Management (EBM) for Audacious Goals!

3 C's of Scrum

Ravi Verma
The Org Whisperer

Entrepreneurship
Agility
Transformation!
Story-Telling

www.smoothapps.com

Understanding the 3 Cs of Scrum B04 E34

Questions

Each role has a clear Learning Path



Product Owner Learning Path

★★★★★ 4.9 from 5 ratings

Unwatch

As described in the Scrum Guide, a Scrum Product Owner is responsible for maximizing the value of the product resulting from the work of the Development Team. How this is done may vary widely across organizations, Scrum Teams, and individuals.

The learning path is organized by a set of Professional Scrum Competencies which each contain a number of focus areas. Resources are laid out in a prescribed order, however it is your journey, so follow in the best way that helps you learn.

Understanding and Applying the Scrum Framework

Empiricism

Scrum Values

Scrum Roles

Scrum Events

Scrum Artifacts

Sprint Goal

Done

Scaling Scrum

Managing Products with Agility

Understanding and Applying the Scrum Framework / Empiricism

The Agile Manifesto

Created by 17 visionaries in 2001, the Agile Manifesto was developed to look at the core principles and values for Agile software development. 10.04.001

★★★★★ 4.8 from 6 ratings

Completed

Empiricism, the Act of Making Decisions Based on What Is

Ken Schriesber writes about commitment and Scrum as an empirical process. 10.03.301

★★★★★ 4.8 from 7 ratings

Completed

The Three Pillars of Empiricism (Scrum)

READ THE SCRUM GUIDE

VIEW THE SCRUM FRAMEWORK

THE SCRUM GLOSSARY

FIND TRAINING

GET CERTIFIED

<https://www.scrum.org/pathway/scrum-master>

<https://www.scrum.org/pathway/product-owner-learning-path>

<https://www.scrum.org/pathway/team-member-learning-path>

<https://www.scrum.org/pathway/agile-leader-learning-path>

Connect with the Scrum.org community



Forums
Scrum.org



Twitter
@scrumdotorg



LinkedIn
LinkedIn.com
/company/Scrum-
org



Facebook
Facebook.com
/Scrum.org



RSS
Scrum.org/RSS



Scrum.org | Blog

Insights from Scrum.org's community of experts

SCRUM PULSE

A free monthly webcast by Scrum.org **Professional Scrum Trainers** addressing common challenges faced by the software profession.



Thank you!