



Scrum & AI: Partners or Rivals?

Alex Ballarin, PST

9 de septiembre de 2026

©1993 – 2026 Scrum.org All Rights Reserved

1

Quick Guidelines

- Your microphones will be muted throughout
- This session is recorded. The recording and slides will be available after the webinar within 24 hours.
- Please ask questions!
 - Submit questions by selecting the Q & A icon:



Who is Scrum.org

Mission:
*Helping People and
Teams Solve
Complex Problems*



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



Alex Ballarin, Professional Scrum Trainer

- Lives in Barcelona, Spain
- PST since 2016
- 22 years of experience helping teams and organizations to deliver better products
- Paying for ChatGPT Plus since the week it came out. 😊



● ALEX BALLARÍN · PROFESSIONAL SCRUM TRAINER

Scrum e IA

¿Aliados o Rivales?

Cómo cambia el trabajo de un Scrum Team cuando la IA entra en el flujo de desarrollo.

● ENCUESTA RÁPIDA

¿Qué describe mejor a tu equipo hoy?

<https://www.itnove.com/webcast-scrum-ia#6>

Antes de esta sesión — ¿dónde está tu equipo con la IA?

☐ A Usamos IA de forma individual, sin un flujo compartido

☐ B La IA está integrada en cómo trabaja el equipo

☐ C Estamos experimentando / evaluando

☐ D Todavía no hemos empezado

● RESULTADOS DE LA ENCUESTA

¿Dónde estamos como grupo?



● 24 votos

● EL DEBATE

"La IA mató a Scrum."

//

Los Sprints de dos semanas se diseñaron para un mundo donde escribir código llevaba tiempo. Ese mundo ya no existe. Con IA, una funcionalidad pasa de idea a código funcionando en minutos. Para eso no hacen falta ceremonias.

IS Ingeniero senior · 4.000 likes en LinkedIn

¿Es esto cierto? Vamos a comprobarlo.

● QUÉ VAMOS A VER

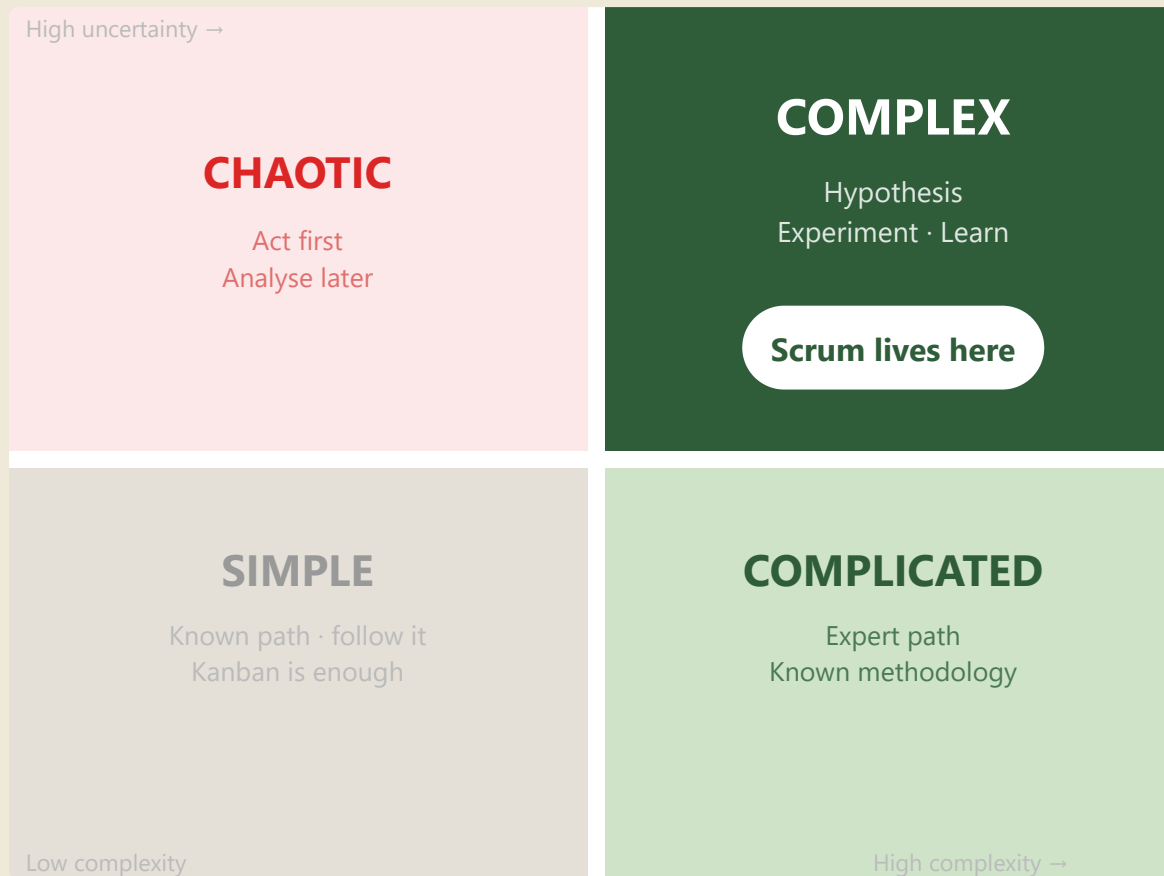
Una pregunta. Cinco respuestas.

01	Para qué sirve Scrum en realidad	8 MIN
02	Qué cambia de verdad con la IA	10 MIN
03	El modelo lógico como brújula	7 MIN
04	Cómo se adapta Scrum	12 MIN
05	¿Aliados o Rivales?	10 MIN + PREGUNTAS

Pregunta guía: *¿Qué cambia de verdad cuando la IA se une al equipo?*

● BLOQUE 1 • DÓNDE ENCAJA SCRUM

No todos los problemas necesitan Scrum.



Scrum se diseñó para problemas **complejos** — donde ni los requisitos ni el camino hacia la solución están claros de entrada.

● AQUÍ VIVE SCRUM

- **Simple:** Camino conocido → Con Kanban basta.
- **Complicado:** Camino experto → Metodología conocida.
- **Caótico:** Sin camino todavía → Actuar primero, analizar después.
- **Complejo:** Camino desconocido → *Descubres* la respuesta, no la planificas.

● SCRUM COMO RESOLUTOR DE PROBLEMAS COMPLEJOS

No es un proceso de entrega. Es una máquina de aprendizaje.

Entrega mecánica

Los Sprints son ciclos de dos semanas para producir output. Entran historias, sale código, la Demo es un checkpoint.

La **velocity** es la estrella polar.

"Done" significa que el código está mergeado.
No se está respondiendo a ninguna pregunta.

Resultado: output rápido, dirección desconocida.

Brújula de resultados

Cada Sprint arranca con un **Sprint Goal** — una hipótesis. La Review es una sesión de feedback, no una demo.

El **impacto** es la estrella polar.

"Done" significa que se ha aprendido algo.
Cada Sprint responde a una pregunta concreta.

Resultado: más lento en features, más rápido en valor.

● LA IDEA CLAVE

La IA no elimina la incertidumbre.

Las razones por las que nació Scrum siguen vigentes: los mercados cambian, los usuarios nos sorprenden, la tecnología se comporta de forma inesperada.

//

El mayor peligro para Scrum no es Waterfall, ni la IA — es Scrum reducido a una cadencia de reuniones sin empirismo.

Si tus Sprints responden preguntas reales, la IA hace Scrum más potente. Si son solo ciclos de output, la IA hace el problema más grande.

● BLOQUE 2 • QUÉ CAMBIA CON LA IA

Tres modelos de desarrollo asistido por IA.

● VIBE CODING

Prompts → código funcionando en horas

Ideal para exploración y prototipado rápido. El desarrollador se mantiene en flujo creativo — sin boilerplate, sin cambios de contexto.

Trade-off: alto potencial de deuda técnica y comportamientos indefinidos en los bordes.

● SPEC-DRIVEN DEVELOPMENT

Specs estructuradas → generación precisa

El desarrollador escribe especificaciones detalladas; la IA genera código de calidad de producción. Menos ambigüedad, menos deuda.

Trade-off: exige pensamiento disciplinado por adelantado — más difícil de empezar, mucho más fácil de mantener.

● AGENTIC AI

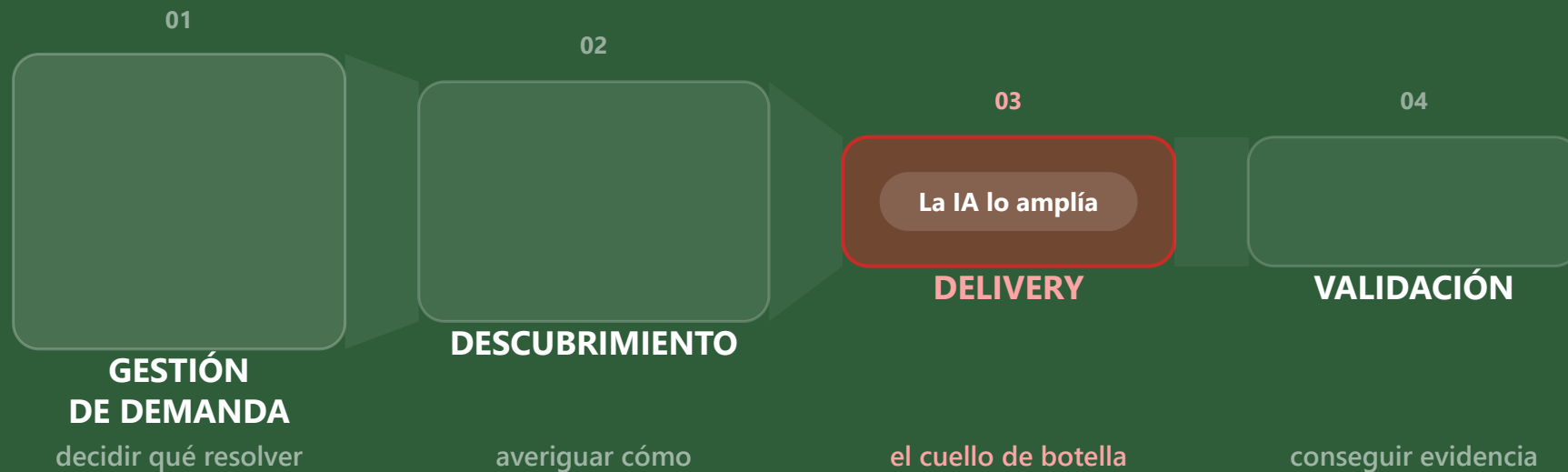
Flujos autónomos → el equipo supervisa

El agente de IA ejecuta tareas de varios pasos de principio a fin: escribe, testea, corrige, hace commit. El equipo revisa los resultados y aprueba las decisiones.

Trade-off: el mayor apalancamiento, pero exige prácticas de validación maduras y límites claros de autonomía.

● EL CAMBIO

El cuello de botella se ha movido.



La IA ha ampliado drásticamente el throughput de **Delivery** — lo que antes llevaba semanas ahora lleva horas.

La restricción ya no es escribir código. Es **decidir qué merece la pena construir** — filtrar la demanda, enmarcar el problema, elegir qué *no* hacer.

● EL RIESGO REAL

Feature factory — con esteroides.

×17

Código generado por desarrollador

×1,3

Releases publicados

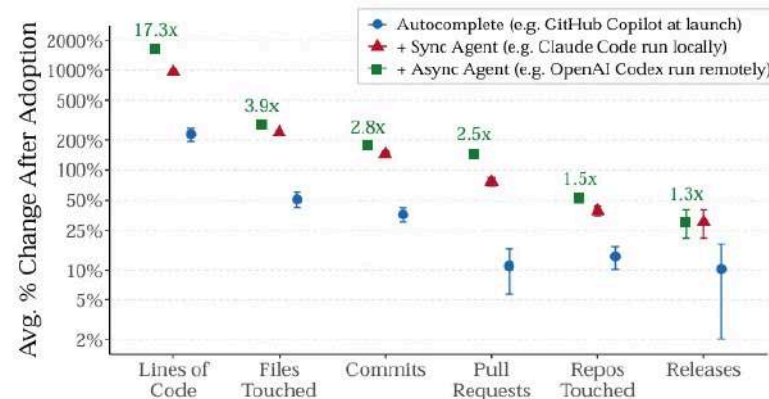
?

Resultados validados

La velocidad no es el problema. La **desconexión estratégica** sí lo es — y la IA la amplifica.

- Los stakeholders piden más. Los roadmaps se expanden.
- Los equipos ejecutan sin cuestionar el valor.
- Los resultados no escalan al ritmo de la velocity.

Figure 1: Overview of Productivity Effects of AI Coding Tools



Demirer, Musolff & Yang · MIT Sloan, 2026 · SSRN 6843118

• BLOQUE 3 • EL MODELO LÓGICO COMO BRÚJULA

La IA acelera un eslabón. La cadena tiene cinco.



El **modelo lógico** (Seiden) muestra qué conecta el esfuerzo con el valor. La IA acelera drásticamente los **Outputs**. El problema: los equipos celebran la velocidad de output como si fuera velocidad de outcome.

Sin Sprint Goals y Product Goals claros, más output solo significa más ruido.

● EL PROBLEMA DEL DESFASE

Cuando la velocity sube, la brecha de medición se ensancha.



A medida que aumenta el ritmo de entrega, el desfase entre lo que publicamos y lo que medimos —cambio de comportamiento de usuario, retención, conversión— crece de forma proporcional.

Volar a ×17 de velocidad sin señales de outcome es navegar a ciegas. Cuanto más rápido vas, más necesitas una brújula.

• EL EJEMPLO GUÍA

GymTonic — un caso, de principio a fin.

• OBJETIVO DE NEGOCIO

Retener a los socios nuevos

Demasiadas altas se dan de baja antes del mes 3. GymTonic quiere **reducir el churn temprano** y aumentar el lifetime value de cada socio.

• PRODUCT GOAL • LA APP

Construir el hábito pronto

Hipótesis: si los socios nuevos llegan a **8 visitas en 30 días**, la retención a 90 días sube. La app existe para conseguirlo.

• SPRINT GOALS • EJEMPLOS

Pequeñas apuestas hacia ello

- Onboarding que reserva la **primera semana**
- Recordatorios antes de cada clase
- Rachas que premian la constancia

● SCRUM ORIENTADO A OUTCOMES

Product Goal y Sprint Goal como filtros de decisión.

Product Goal

El resultado estratégico que persigue el equipo en esta fase del producto.

No es una lista de features. No es el alcance de un proyecto.

Una **hipótesis** sobre impacto: *"Si los nuevos socios de GymTonic llegan a 8 visitas en sus primeros 30 días, reducimos el churn a 90 días un 20%."*

Cada ítem del backlog se evalúa contra este filtro antes incluso de discutirlo.

Sprint Goal

La pregunta concreta que este Sprint va a responder.

No es un Sprint backlog. No es un compromiso de features.

Un **objetivo de aprendizaje**: *"Creemos que reservar las 3 primeras sesiones durante el onboarding aumentará la asistencia en la semana 1."*

El Daily Scrum pregunta: ¿seguimos en camino de responder esta pregunta?

● BLOQUE 4 • CÓMO SE ADAPTA SCRUM

El ciclo empírico se mantiene. Cambian la frecuencia y el foco.

Lo que NO cambia

Los tres pilares: **Transparencia, Inspección, Adaptación.**

La necesidad de un Product Goal claro.

El valor de un equipo multifuncional y autogestionado.

Ciclos cortos con feedback real.

El criterio humano en los puntos de decisión.

Lo que SÍ cambia

La **duración del Sprint** puede acortarse — generar código ya no es la restricción.

El **foco del Daily** pasa de reportar progreso a validar output.

El **Refinement** se convierte en trabajo de especificación — el input para la generación con IA.

La **Review** pasa de demo de features a conversación sobre resultados.

La Definition of Done tiene que subir al ritmo de la velocidad de output.

- DURACIÓN DEL SPRINT Y DEFINITION OF DONE

Cuando el código se genera en horas, dos semanas es demasiado para esperar feedback.

Acortar los Sprints no va de ir más rápido. Va de aprender más rápido — reducir el tiempo entre una hipótesis y su primera señal real.

- **Sprint Goal** → un resultado concreto para el usuario o la empresa.
- **Product Goal** → un resultado más estratégico —también cuantitativo— del producto.
- **La Definition of Done sube** → añadir un sistema de control (harness). El listón no puede bajar porque entregar salga más barato.

Si tu DoD es la misma que antes de la IA, estás acumulando riesgo invisible.

● EVENTOS ADAPTADOS

Los eventos evolucionan. Su propósito se mantiene.

Daily Scrum

Pasa de **sincronizar progreso** a **validar output y revisar decisiones de la IA**.

Las preguntas que importan ahora:

- ¿El output de ayer cumple de verdad el Sprint Goal?
- ¿Qué decisiones del agente tiene que revisar o revertir el equipo?

Sprint Review

Pasa de **demo de features** a **conversación sobre resultados**.

Las preguntas que importan ahora:

- ¿Qué hemos aprendido sobre el comportamiento real de los usuarios?
- ¿Las señales apoyan o cuestionan el Product Goal?
- ¿Qué deberíamos dejar de construir?

Retrospective

Pasa de **fricción de proceso** a **calidad de la colaboración humano-IA**.

Las preguntas que importan ahora:

- ¿Qué specs producen el mejor output de IA?
- ¿Dónde estamos supervisando de más o de menos al agente?
- ¿Qué pasos de validación nos estamos saltando?

● EL SCRUM TEAM HÍBRIDO

Los roles evolucionan. La accountability se mantiene.

Product Owner

Se convierte en **validador de hipótesis**, no en aprobador de features.

El backlog es un conjunto de apuestas, no una lista de tareas. El trabajo del PO es maximizar la relación señal-ruido: ¿qué apuestas merece la pena correr este Sprint?

La habilidad clave: traducir resultados de negocio en Sprint Goals verificables.

Scrum Master

Se convierte en **facilitador del aprendizaje humano-IA**, no en gestor de reuniones.

Ayuda al equipo a construir la disciplina de especificar bien, validar con rigor y mejorar el bucle de colaboración. Elimina impedimentos para supervisar con eficacia el output de la IA.

Developers

Se convierten en **arquitectos y supervisores del output de la IA**, no en operadores de prompts.

Diseñan el sistema, escriben las especificaciones que producen código de calidad, revisan lo que genera la IA y son dueños de la Definition of Done. La IA ejecuta; los developers juzgan.

● EL BACKLOG EVOLUCIONA

De historias de usuario a hipótesis de valor.

Antes

"Como socio de GymTonic quiero recordatorios de clase para no perderme mis sesiones."

- Captura una **petición** — asume que la feature crea valor.
- No dice nada sobre qué vamos a medir, ni cuándo parar.
- El output es una **feature** → éxito = *publicada*.

Ahora

"Creemos que recordar a los socios 2 h antes de una clase reservada sube la asistencia $\times 1,5$. Lo probaremos con el 20% de los socios de GymTonic en el Sprint 8."

- Captura una **hipótesis** — define el resultado y la señal.
- Incluye una **condición de parada** si falla.
- El output es **aprendizaje** → éxito = *validado*.

El Spec-Driven Development cambia el *formato* de la especificación. El cambio de arriba cambia su *propósito*.

● EL NUEVO CUELLO DE BOTELLA

Cuando construir cuesta casi nada, la decisión es el capital escaso.

Más ideas que nunca. Más peticiones de stakeholders. Más posibilidades técnicas. La capacidad de filtrar, priorizar y decir que no es ahora la habilidad más valiosa —y más escasa— del equipo.

//

El Product Goal no es un artefacto de planificación. Es el filtro que protege al equipo de ir muy rápido en la dirección equivocada.

Los equipos que dominen la gestión de la demanda en la era de la IA construirán diez veces más valor con la misma gente. Los que no, construirán diez veces más desperdicio.

● ¿ALIADOS O RIVALES? – LA RESPUESTA

Se diseñó para aprender rápido.

Scrum no se diseñó para producir rápido — de producir se encarga la IA ahora. Lo que Scrum aporta es un marco para **aprender rápido**: hipótesis claras, ciclos cortos, resultados validados.

El equipo que gana en la era de la IA no es el que más publica. Es el que responde más rápido a las preguntas correctas.

● TRES COSAS QUE HACER ESTA SEMANA

Empieza por aquí.

01	Define un Product Goal claro	ANTES DE USAR IA PARA CONSTRUIR NADA
02	Audita tu Daily Scrum	¿VALIDA VALOR – O REPORTA ACTIVIDAD?
03	Mide resultados	NO SOLO VELOCITY

Si haces una sola cosa: escribe un Product Goal antes de tu próxima Sprint Planning — el resultado que persigues, por qué importa ahora, y cómo sabrás que lo has conseguido.

//

La velocidad no es el problema. El problema es ir muy rápido en la dirección equivocada.

● GRACIAS · PREGUNTAS

Hablemos.

Alex Ballarín · Professional Scrum Trainer · ITNOVE

- **Scrum.org** · scrum.org/alexballarin
- **Web** · itnove.com
- **Email** · alex@itnove.com

Questions

Connect with the Scrum.org community



Forums
Scrum.org



X
@scrumdotorg



LinkedIn
[LinkedIn.com/
company/
Scrum-org](https://www.linkedin.com/company/Scrum-org)



Facebook
[Facebook.com
/Scrum.org](https://www.facebook.com/Scrum.org)

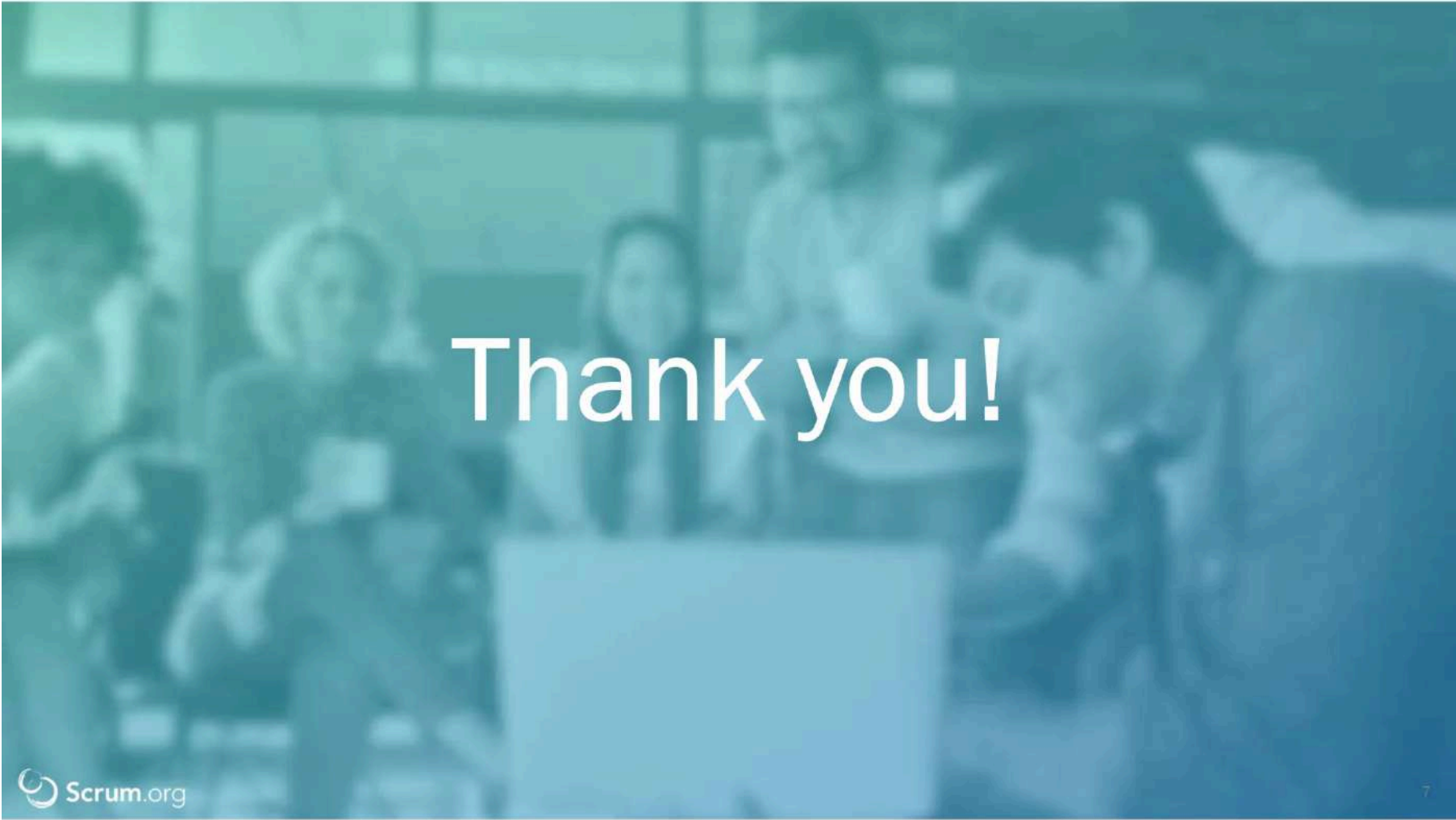


RSS
Scrum.org/RSS



YouTube
[@ScrumOrg](https://www.youtube.com/ScrumOrg)





Thank you!