SHOCK THERAPY
SELF ORGANIZATION IN SCRUM

Jeff Sutherland, Co-Creator of Scrum
Scott Downey, Agile Coach, MySpace
Björn Granvik, CTO, JayWay

Jeff Sutherland, Ph.D.
Co-Creator of Scrum
+1 617 606 3652
jeff.sutherland@scruminc.com

Scrum Training Institute

Jeff Sutherland, Ph.D.

- **Chairman, Scrum Training Institute**
- **CEO Scrum, Inc. and Senior Advisor, OpenView Venture Partners**
  - **Scrum coach, mentor, and trainer to venture group**
  - **CTO/VP Engineering for 9 software companies**
  - **Created first Scrum at Easel Corp. in 1993.**
  - **Rolled out Scrum in next 5 companies**
  - **Achieved hyperproductive state in all companiesSignatory of Agile Manifesto and founder of Agile Alliance**

  - *http://jeffsutherland.com/scrum*
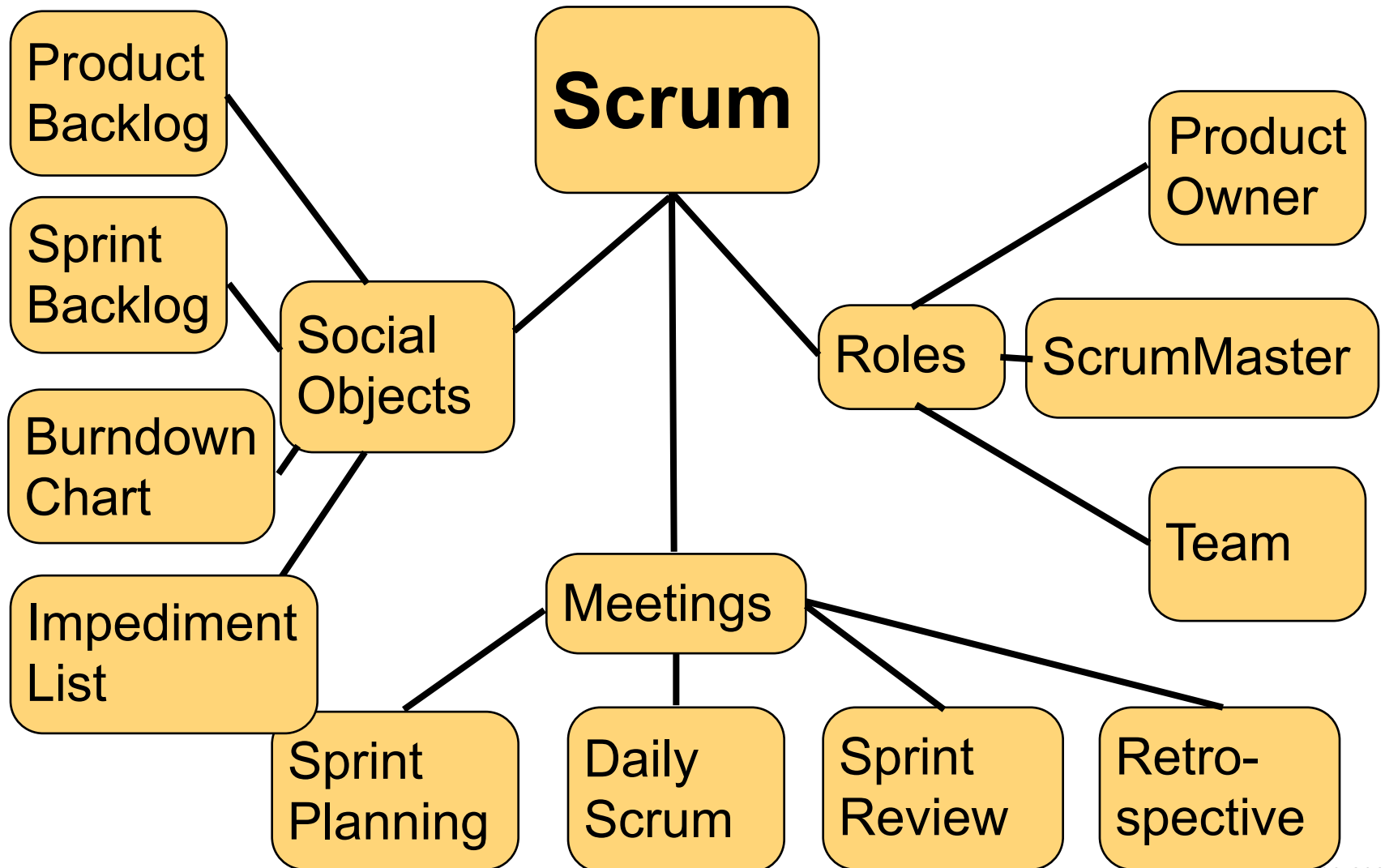 - *jeff.sutherland@scruminc.com*

What is Scrum?

- Not a methodology
- Not a defined process
- Not a set of procedures

- Scrum is an open development framework with a simple set of rules.
- The rules are constraints on behavior that cause a complex adaptive system to self-organize into an intelligent state.
- It enables an average team to self-organize into a super-intelligent team that goes 10 times as fast as normal teams.
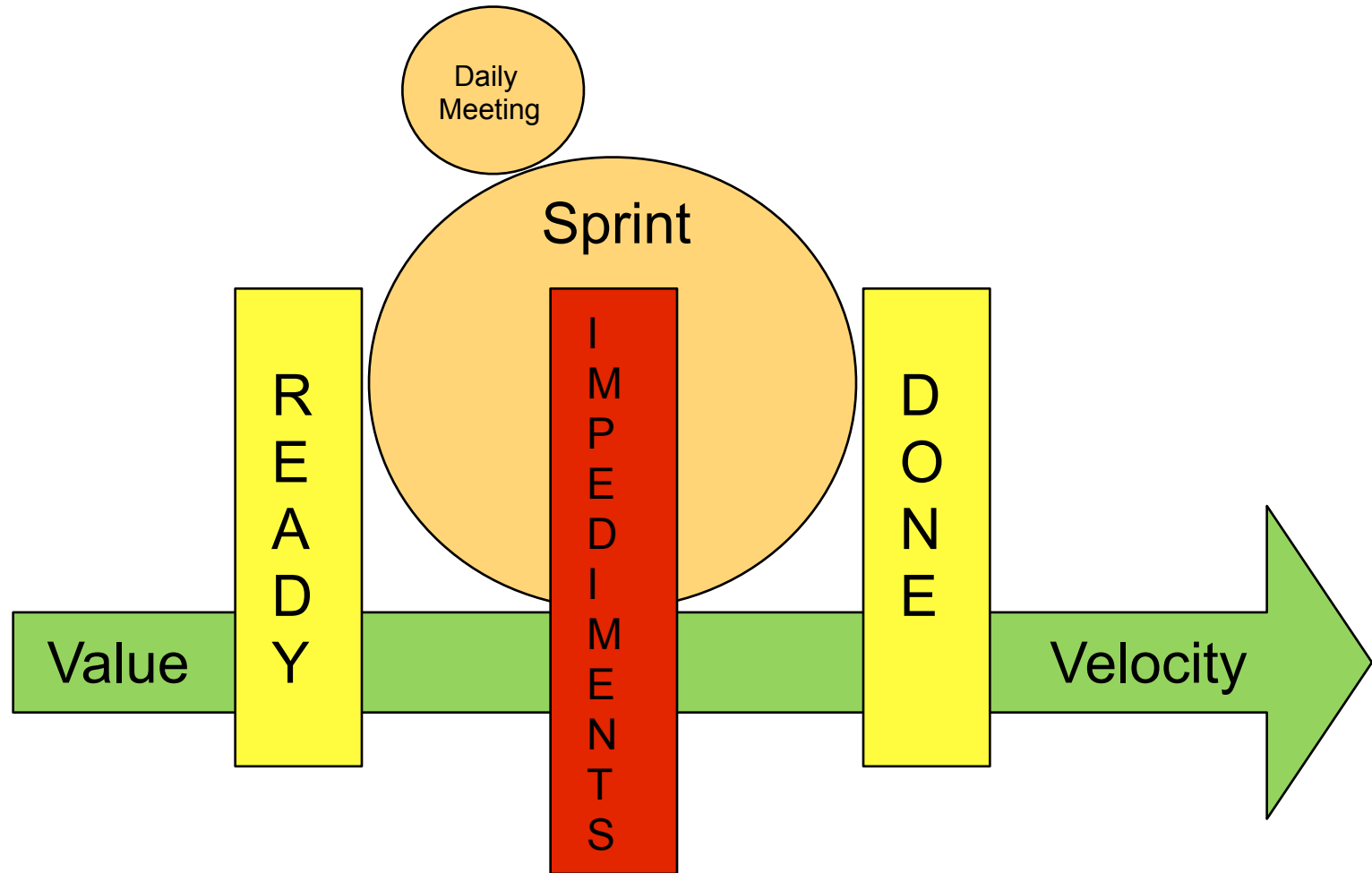
3

Abstract

- Scrum was designed for hyper-performing teams that operate at 5-10 times the velocity and quality of waterfall teams. It is linearly scalable across geographies to any size.

- High performance depends on the self-organizing capability of teams. Understanding how this works and how to avoid destroying self-organization is a challenge.

- Organizations in the U.S. and Europe are seeing a common strategy for bootstrapping hyperproductive teams. They are also seeing management systematically destroy most hyperproductive teams.

- The good news is that hyperproductivity appears to be a learned team competency like riding a bicycle.

Scrum is a Simple Framework

Product Backlog

Sprint Backlog

Burndown Chart

Impediment List

Social Objects

Scrum

Roles

Product Owner

ScrumMaster

Team

Meetings

Sprint Planning

Daily Scrum

Sprint Review

Retro-spective

© Jeff Sutherland 1993-2008

Scrum Dynamic Model

© Jeff Sutherland 1993-2008

Challenges for new Agile developers

- They don't know how to self-organize.
- They can spend three weeks arguing about the format of their Scrum board and get nothing done
- They don't get the highest priority story done first. Everyone works on their own story and the sprint fails.
- They can't get DONE at the end of a Sprint.
- They allow things not READY to enter the Sprint.
- They can take years to figure this out. Some never do.

Getting to hyperproductive state

- Hyperproductivity is as at least Toyota level of performance - 4 times industry average
- It used to take two years for a company to achieve 300% improvement
- We now see 300% improvement in 3 two-week Sprints in some portfolio companies
- The challenge is to consistently bring teams to a high performing state quickly

8

Velocity in Function Points/Dev month

	Scrum[1]	Waterfall[1]	SirsiDynix[2]
Person Months	54	540	827
Lines of Java	51000	58000	671688
Function Points	959	900	12673
Function Points per Dev/Mon	17.8	2.0	15.3

1. M. Cohn, User Stories Applied for Agile Development. Addison-Wesley, 2004
2. J. Sutherland, A. Viktorov, J. Blount, and N. Puntikov, "Distributed Scrum: Agile Project Management with Outsourced Development Teams," in HICSS'40, Hawaii International Conference on Software Systems, Big Island, Hawaii,

9

Dutch Velocity vs. Russian Velocity

	SirsiDynix[2]	**Xebia[3]**
Person Months	827	125
Lines of Java	671688	**100000**
Function Points	12673	**1887**
Function Points per Dev/Mon	15.3	**15.1**

1. M. Cohn, User Stories Applied for Agile Development. Addison-Wesley, 2004
2. J. Sutherland, A. Viktorov, J. Blount, and N. Puntikov, "Distributed Scrum: Agile Project Management with Outsourced Development Teams," in HICSS'40, Hawaii International Conference on Software Systems, Big Island, Hawaii,
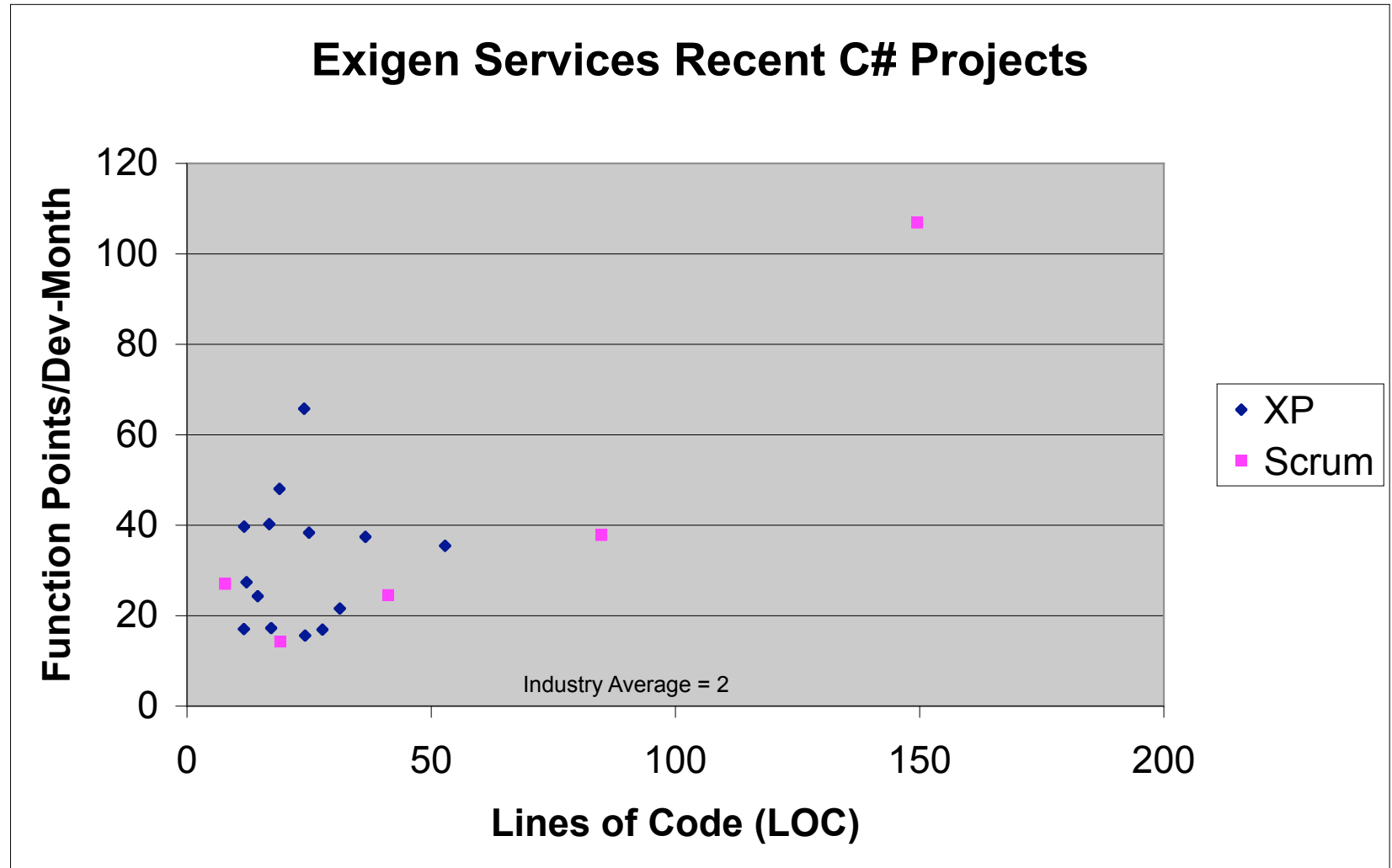3. J. Sutherland, G. Schoonheim, E. Rustenburg, M. Rijk. Fully Distributed Scrum: The Secret Sauce for Hyperproductive Outsourced Development Teams. Agile 2008, Toronto, Aug 4-8 (submission, preliminary data)

Russian projects velocity data suggests high velocity is not an accident

Exigen Services Recent C# Projects

Function Points/Dev-Month vs Lines of Code (LOC). Legend: XP (blue diamonds), Scrum (pink squares). Industry Average = 2

11

How can an Agile coach achieve hyperproductivity in a typical company?

- Typical company is waterfall or ScrumButt
- Management does not understand Agile and is not fully committed
- Instance gratification is needed so why not try:

SHOCK THERAPY

- MySpace in Beverly Hills
- JayWay in Sweden
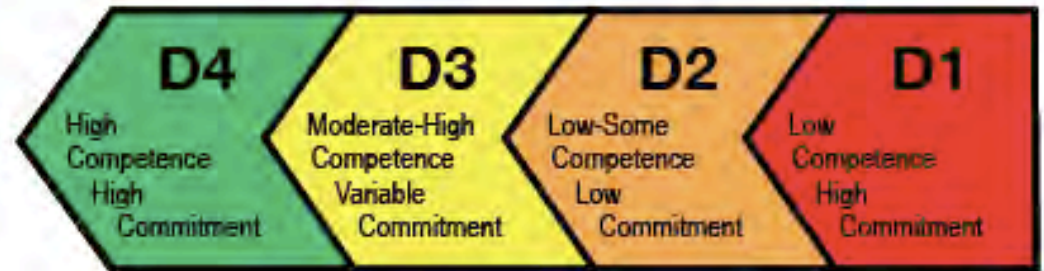- Pivotal Labs in San Francisco

© Jeff Sutherland 1993-2008

The Shock Therapy

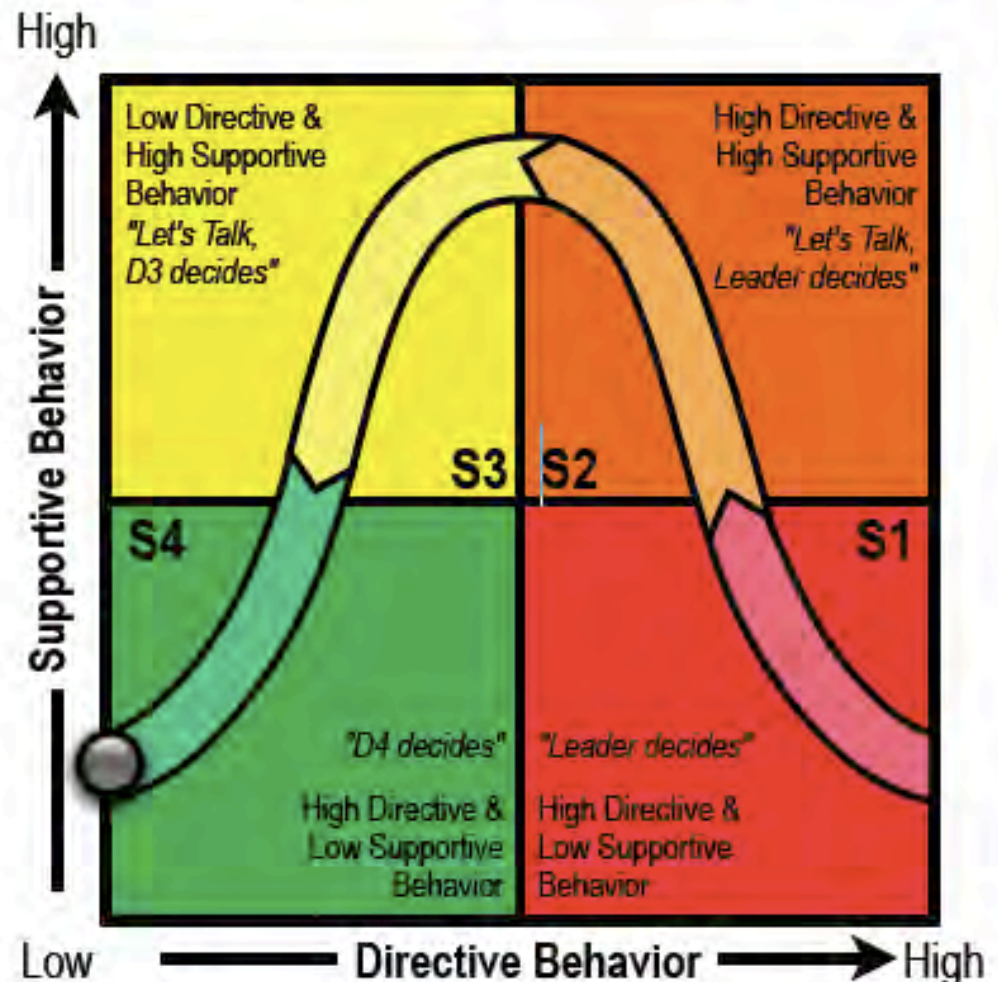- "A set of good practices but no choice"

Task or Goal!

D4	D3	D2	D1
High Competence High Commitment	Moderate-High Competence Variable Commitment	Low-Some Competence Low Commitment	Low Competence High Commitment

Situational Leadership

- **Directing**
- **Coaching**
- **Supporting**
- **Delegating**

See changingminds.org

High

Supportive Behavior

| S3 — Low Directive & High Supportive Behavior "Let's Talk, D3 decides" | S2 — High Directive & High Supportive Behavior "Let's Talk, Leader decides" |
| S4 — "D4 decides" High Directive & Low Supportive Behavior | S1 — "Leader decides" High Directive & Low Supportive Behavior |

Low ——— **Directive Behavior** ——→ **High**

© Jeff Sutherland 1993-2008

14

Patterning

- Shock therapy is similar to martial arts training

- When you walk into the dojo, the Sensei owns the mat

- You do exactly what he does, over and over again, until it is part of muscle memory

- Only when you have demonstrated mastery of the basic practices are you allowed to improvise

- Before you have gained discipline, centering and flexibility, you are a hazard to yourself and others

© Jeff Sutherland 1993-2008

Case Study: MySpace

- MySpace has several hundred developers
  - about 1/3 waterfall
  - about 1/3 ScrumButt with Project Managers
  - about 1/3 pure Scrum
- Scott Downey, MySpace Agile coach repeatedly takes teams to high productivity state in a few weeks
  - Average time to 240% of the velocity of a waterfall team is 2.9 days per team member where the team includes the ScrumMaster and the Product Owner

© Jeff Sutherland 1993-2008

16

Starting up a Scrum Team

- Scrum, as a framework, permits teams a ton of options to customize it to their own environment. In Scott's experience, most teams just starting out are so overwhelmed with choices that they can't find a constructive way to start.

- It occurred to him one day that Scrum Teams are the customers of the ScrumMaster.

  - We all know that customers of our enterprise don't really know what they want until they have seen it. So why do we expect Scrum Teams to know how to play Scrum if they haven't seen a prototype?

- So when Scott joins a team as their ScrumMaster, he issues a few non-negotiable rules (gently if possible, firmly if necessary).

17

Starting up a Scrum Team

Scott describes it:

- My rules remain in effect until the team has met three criteria:

    – They are starting to go Hyper-Productive (>240% higher targeted value contribution)

    – They have completed three successful Sprints consecutively

    – They have identified a good business reason to change the rule

- The rules are roughly these:

    – <u>Everyone</u> on the team will attend a Scrum Training session.

        • I conduct an extremely condensed "Scrum at MySpace" course in about four hours, and the entire team comes together in one session. Until everyone has been trained, we won't begin our first Sprint.

    – Sprints will be one week long.

        • I justify this by pointing out that there is a reason geneticists study mutations in Fruit Flies instead of Elephants – they want to see the mutations quickly and adapt their studies accordingly. So do I.

18

Teams hate one week Sprints (at first)

- I have been able to coax every team into giving me at least 4 one-week Sprints as a trial.
  - Engineer: "But I can't do anything in a week!"
  - Scott: "Then simple math suggests that you can only do four nothings in a month."
- Interestingly, by the time the teams have met the three criteria for changing this rule (Sprint length), only one team so far has ever elected to change it

19

They will use my definition of "Done."

- This is often one of the thorniest issues to iron out with a team, so I take it off the table until they have some shared success as a foundation.

- My initial definition of "Done" is this:
    -       1. Feature Complete
    -       2. Code Complete
    -       3. No known defects
    -       4. Approved by the Product Owner
    -       5. Production Ready

Monday, August 24, 2009

All estimates are in Story Points

- It's usually at about week three when I can intentionally spark a debate over whether a card is a 3 or a 5, and then have the pleasure of watching the passion with which they debate these recently meaningless values.

- I also make a point of shouting "BAA!" whenever they all vote the same value for a given card. My intent is to show them how often they actually agree in their vote.

- As the mood on the team lightens up, some teams begin scanning the other votes and "baa"ing like sheep when that happens.

- Only one has returned my "Baa!" with a "Humbug!" In any event, they all start having fun with it and that's important.

21

We use a physical information radiator

- Not only do I insist on a physical Information Radiator, but I have a basic template that I use initially for all teams.

- I choose the location of the Scrum Board unilaterally and use it as the focus of the Daily Stand-Up Meeting.

- When the team is first formed, I let them focus on the interaction with their teammates (the three Daily Scrum questions) and I move their cards across the Radiator's surface myself.

- Within a couple of weeks, they start moving cards themselves without being asked. This is usually my first indication that I can begin slowly stepping back and relaxing my demands.

22

The "Sprint Meeting" is four hours once a week

- The first complaint of most Engineers is that they perceive Scrum imposing a highly disruptive schedule on them, with more meetings than they somehow think they have ever had before.

- To minimize this common concern, I consolidate everything but the Daily Stand-Up meetings into a single four-hour meeting (covering Sprint Review, Retrospective & Sprint Planning).

- Within a few weeks, the teams usually only need a couple of hours. And by the end of about eight Sprints together, the meeting is becoming ninety minutes or less in duration for a one week Sprint.

23

My initial "Sprint Meeting" agenda is:

1. Demonstration: where the Team shows the Product Owner the working software they completed and "earns" Story Points if everyone concurs.

2. The Retrospective comes next, aided by whole-team metrics, never individual metrics.

3. Product Backlog Presentation follows.

4. The Team is free to question motives, suggest alternatives and add requirements at this point. I reject any improperly formed User Stories on behalf of the team.

24

Sprint Planning:

4. Estimation and Negotiation signals that the meeting is nearly complete.

       4. The Product Owner participates in an *advisory-only* role during this phase of the meeting.

5. Sprint Backlog Commitment is the final act of the Sprint Meeting. Once the team commits to the work, the meeting adjourns.

25

Multi-Tasking is Forbidden

- Work must be done in Priority Order.

  - Some Engineers understand this right away.

  - I insist and enforce that they work on cards without multi-tasking and in priority order.

- I also have standard layouts that I use for their initial Sprint Planning Boards, User Stories, Story Cards, Burndown Charts and Velocity tracking.

  - I take full responsibility for entry and management of the (horrible) Scrum tool we have.

Friendly smile and a "please"

- As I said in the beginning, I do try to get all of this done with a friendly smile and a "please" but generally have to insist -- sometimes quite forcefully

- Although the beginning is rocky, we usually start laughing and having fun within a week.

- And as they become more comfortable and competent with Scrum, I relax my grip on some of the rules so long as they respect the principles of Scrum.

27

Three key reasons for success ...

1. I find the biggest, nastiest problem that the team has and solve it within a day or two of the first Sprint Meeting.

    - Some teams quickly volunteer this problem to me in their first Retrospective while other problems require observation, careful listening and behind-the-scenes reconnaissance to tease out.

    - Especially for those teams who haven't worked directly with me yet, having that very large problem go away underscores that they are important to me, that I take them seriously and that I am working hard to make their world a better place.

28

Key reason #2

2. Since I am the Master Scrum Master/Scrum Evangelist for the entire company, I am almost never their permanent Scrum Master.

- This gives me the freedom to create a bit (but only a very small amount) of "Us vs. Him" atmosphere at first.

- It causes the team to bond in an entirely new way than they have before, and also sets up their permanent Scrum Master to be the "good cop" down the road.

- This also allows me to be more firm about, for example, standing during Stand-Up, keeping SP estimates private until the laydown during estimation, etc.

29

Hyperproductivity is the target

- I generally have to bow out and move on to another team after 6-12 weeks, by which point they are functioning very well and are (on average) around the 500% mark.

- Overall, most teams tolerate this approach very well and learn good habits more quickly.

Monday, August 24, 2009

Key reason #3 - facilitating self-organization

3. I like Socrates' approach. Sometimes I stop the meeting and ask the team, "Team, do any of you see something going wrong with our meeting right now?"

Ironically, it is almost always the most skeptical person who is the first to correct the insolently perched teammate. Soon, they start calling one another on leaning or sitting long before I stop the meeting and ask what's going wrong.

31

Hyperproductive in four weeks

- I have pulled teams into hyper-productivity in as few as four weeks.

- [Note: One of Scott's co-workers calls him "The Scrum Whisperer."]

- I have one team that has achieved 1,650% higher targeted value contribution per week after just four months (16 Sprints) together. We are pretty proud of those numbers.

- I've also noticed that teams using this "quick format" approach tend to hit their Velocity elbow much sooner.

32

Followup on the kick start ...

- It's a fairly large culture shock for most teams and doesn't yield a lot of "let's go to lunch" invitations at first.

- But, per my VP of Engineering, "They only hate you for about 2-3 weeks. Then they're indifferent to you for another few weeks."

- "Then they scream bloody murder if I try to take Scott away from them."

- I do stay in touch with teams that I've kick-started like this and, with one notable exception, they have all continued their trend of improvements in my absence.

- My techniques are always evolving ...

  - Scott Downey

  - http://www.MySpace.com/PracticalScrum

Monday, August 24, 2009

Data Collection:

Establishing initial velocity

- Product Owner presents the entire Product Backlog in order of Priority.
  - Team selects a Sprint Backlog from the highest priorities in the Product Backlog
- Team then reorganizes the Sprint Backlog in order of complexity (no numbers attached, just perception).
  - Scrum Master assigns a Story Point value to the easiest card; Values for other cards are comparatively derived from that assignment.
  - Sum of the estimates provides an estimated Velocity
- Wait for the Sprint to complete to see which items were accepted by the SPO
  - Sum the original estimates of the cards accepted by the SPO. That is initial Velocity.

© Jeff Sutherland 1993-2008

34

Shock Therapy at MySpace

Story Point Velocity for HyperProductive Teams That Became Destroyed Successes

Sprint	Team A Commit	Approved	Work Cap	% Increase	Team B Commit	Approved	Work Cap	% Increase	Team C Commit	Approved	Work Cap	% Increase	Team D Commit	Approved	Work Cap	% Increase	Team E Commit	Approved	Work Cap	% Increase
1	8	6	6	100	7	4	5	100	22	12	22	100	58	27	27	100	24	5	22	100
2	14	11	11	183	21	6	6	150	16	16	25	133	49	49	49	181	26	22	26	440
3	13	10	11	167	20	20	20	500	21	21	44	175	61	27	27	100	11	11	17	220
4	7	7	15	117	21	15	15	375	36	35	54	292	34	34	34	126	24	24	22	480
5	20	18	29	300	18	15	18	375	47	43	50	358	39	23	28	85	26	34	56	680
6	29	17	29	283	11	3	11	75	60	60	80	500	29	29	39	107	29	21	37	420
7	22	11	20	183	17	17	20	425	49	37	47	308	60	60	65	222	38	29	46	580
8	19	19	27	317	16	13	13	325	58	53	66	442	98	98	102	363	41	41	52	820
9	29	23	29	383	16	16	16	400	58	55	60	458	70	70	78	259	42	16	44	320
10	20	15	24	250	22	18	18	450	48	43	50	358	107	107	108	396	21	21	33	420
11	21	21	26	350	17	14	14	350	32	32	43	267	94	89	89	330	37	37	44	740
12					25	21	27	525	80	47	62	392	123	123	123	456				
13					18	18	18	450	49	33	41	275	84	84	84	311				
14					19	16	25	400	43	39	41	325	97	97	97	358				
15					20	20	28	500	50	37	45	308								
16					18	15	21	375	46	41	53	342								
17					18	18	33	450	50	50	58	417								
18					17	17	30	425	53	45	49	375								
19					30	30	41	750	64	64	66	533								
20					29	29	50	725	72	61	70	508								
21					36	36	57	900												
22					40	39	46	975												
23					44	44	70	1100												
24					57	35	62	875												
25					42	42	55	1050												

Color Legend: First Time >240% | Max. Achieved | Period of Instability

Where Are They Now...?

Team A was disbanded when the project was moved to a newly opened, remote office. Team members were scattered among other teams.

Team B had members stolen during Sprints 4-11. They remained stable in membership after that but two members had extended absences during Sprints 16-18. The original team was reunited for 23 and beyond.

Team C got a new Anti-Agile, Command-and-Control Director who removed the original SPO, split the team and put two novice SPOs in place. Their Scrum adoption has not recovered.

Team D lost their SPO after the end of a major initiative and, though they have begged for one, we have been unable to get a new SPO allocated to that team.

Team E lost their SPO and has been unable to replace him.

Shock Therapy at Jayway

The '3 Man' Project						
	Fredrik Källbäck and Adam Skogman, Jayway					
Sprint	Points	% Increase		Team:	3 programmers	
0	2				2 testers	
1	16	100 %			1 lead programmer	
2	8	50 %			1 ScrumMaster	
3	27	169 %			1 Product Owner	
4	28	175 %		Iteration length:	2 weeks	
5	26	163 %				
6	60	375 %		Comments:		
7	21	131 %		Iteration 0 only week long and therefore not part of calculation.		
				Zero known errors on each sprint.		
				Drop in sprint 7 due to management pulling resources and training of staff.		
				Project delivered more than one year ahead of schedule.		

36

Scrum Sensei

When you need me, but do not want me, then I will stay.

When you want me, but do not need me, then I have to go.

Nanny McPhee

©2006 Universal Studios

Scott Sutherland 1993-2008

37

The Scrum Sensei

- Experienced ScrumMaster
- Enforces rules/recipes
- Bad ScrumMaster as in "good cop/bad cop"
- Compassion!
- Leaves when self-organization works

Monday, August 24, 2009

The Senior Agile Programmer

"Unfortunately, no one can be told what the Matrix is. You have to see it for yourself."

Morpheus

© 1999 Warner Bros. Entertainment Inc

© Jeff Sutherland 1993-2008

39

The Senior Agile Programmer

- Translates methodology into basics
- Sets the "good practices" on a bit and byte level
- Coaches by showing

40

Scott's Rules

- Training for everyone
- Backlog must be ready before taking into Sprint
- Pair immediately if only one person can do a task
- No Multitasking
- Physical Scrum Board
- 1 Week Sprints
- Burn down Story points only
- Everything (including support) is prioritized by PO
- Servant leadership – it's not about you

41

Critical Scrum Success Factors

- PO/SM/Team trained properly
- Done means acceptance tests pass (testers on team)
- Product backlog ready (remove if not ready)
  - Clear to developers, sized and prioritized (before Sprint Planning)
  - Estimated in story points
- Scrum Board
  - Burn down in priority order, test completed story immediately
- Burn Down Chart
  - Track Done
  - Execute Emergency Procedure when needed
- Know velocity
- Remove top priority impediments
- Prevent disruption of team
  - No multitasking
  - Minimize team member change

Questions?

Emergent Architecture

43