I have written this text because Mrs Frédérique Sine fills some level of skepticism in my point of view of Scrum, more precisely in the way I see the role of Scrum Master and she is also afraid of that I have never worked in Scrum Master role as defined "in the book". She is right in both point. And here I describe why it is good thing at the end.

(NB: The request to summary my Scrum view was surprising and I committed to myself to close deadline and I did not have time to organize into a really good structure even if the length would need it. As I was writing it I have realized that I could talk about by Scrum/Agile management experience a lot. Several war stories come up in my mind. So sorry for being a little bit unstructured).

I start the latest point. It is true that I have never worked as a pure Scrum Master (and I was never ever seen a pure Scrum Master ever – including my Scrum trainer called Mitch Lacy). In Statlogics I was the director of development, so a manager, in Intrasoft I am a developer. Both role is unacceptable as Scrum Master as defined “by the book”.

In practice in both position I was doing Scrum Masters job *AND many other* kind of jobs required to run the business.

And that is directing to my approach of Scrum. I have participated in many Scrum introduction/implementation and I have direct info from other Scrum implementations in other organizations. There was not a single place where pure Scrum is introduced as it is. There was many reasons why it was not possible to implement. Sometimes it is because of the lack of pre-requirement (e.g.: no Product Owner available) or development team had serious technical difficulties or the organization was not even intended to introduce Scrum, they just used the “title” because it look modern and professional. (a document typically started like “Scrum in <the name of company>)

On the other hand in all case there was a great improvement in development process and (mostly) software quality!

Here are my personal Scrum implementation experiences:

Intrasoft: In a team on 8 developers working in European Parliament premises. There were 1-2 deliver per year and there were lots of (stupid) defects in the delivered software. There were a need on better quality and more rapid delivery.

It is failed in Scrum:

* there was no Product Owner (nobody has accepted the responsibility for the product – too much politics, too less responsibility)
* no Scrum Master. The official Scrum Master has no clue about what to do. We (me and other college with SM certificate) had to take over that responsibility with serious limitations.

And was successful Agile implementation with the following characteristics:

* production delivery in every 2 month (EP infrastructure limitation only)
* kept Scrum ceremonies which improved the communication within (!) the team
* introduced a much more flexible requirement management process.
* drastically improved software quality by introducing test automation. We are always close to production ready state.
* discarded burndown chart because we are forced to use EP tools to tract task and time and that is not supporting any kind of agile practices. And we have rejected the double administration overhead.

Statlogics: More than 20 developer plus a QA team and business analyst team cooperation.

Failed in Scrum:

* no direct Product Owner, only proxy through business analysts. Communication channel is something like the following: The need recognized by end user -> head of his unit -> PM on client side -> business analyst in our side -> Developer. Lots of information is lost.
* no delivery at the end of spring

Success in Agile:

* in average 1 delivery per week to client environment (pre-prod or prod)
* dramatically improved quality by test automation in many level, unit, integration , user interface. Statistical fact: in a certain period of time more than 90% of client issues was related to mistake in requirement or new change request. And only the remaining was related to actual developer mistakes (before it was close to the opposite).
* smooth cooperation of such a big team.
* improved requirement quality by introducing requirement inspection
* we made development fully transparent to management and that built a trust what was missing before (non technical management cannot understand technical terminologies but as the process become transparent the were able to recognize when things went well and what thing went smoothly) they had a full trust in development.
* the work process we used become mandatory to introduce into other unit in the company (because of its success)

How: After introducing Scrum we have slowly adapted practices and evolved to a different kind of method which is fitting to our company and client needs. That method Scrum evolved at the end is quite close EVO by Tob Gilb (my personal preference amongst all Agile methods - it is proving continuously that effective and do not have the weakness of Scrum).

I was not a scrum muster but I was an "agile project manager". I was doing much more than what was needed for Scrum Master.

And here I am reflecting to issue that I am "skeptical" in Scrum. I am skeptical when I see an open position of Scrum master. Because SM and Scrum is a solution to a problem. And by having SC or introducing Scrum is not always solution to problem.

My question: what kind of problem the company have. It is possible that the problem can be resolved differently by introducing some *technique* without talking about a whole development method. (Low quality of software? test automation, manual testing is not enough for long term, Lazy developers? Make development transparent to management. Not delivered what client requested? Requirement inspection process and focus of business value impact; frequent delivery of smaller chunks etc.)

I am a strong believer in Agile methods (including Scrum). Agile is an interval between Waterfall and Chaos. And in that interval Scrum is just a specific point. There are many way of doing software development process well (or at least better) and without any exception is always causing to introduce some kind of agile method and technique.

Scrum is a good starting point but I am almost 100% sure that within 6-12 month of introducing Scrum the resulted process will be different from Scrum but it will be optimized to the company (Scrum terminologies are kept but practices are different). That is the why I say I have seen failed Scrum introduction. But I have seen many agile introduction successfully.

Summary:

I see the power of Scrum. And I see that Scrum is never enough. It always needs some techniques to make it complete. I see the weakness of Scrum and I have the capability and experience to adapt/change Scrum to fit to the environment.