Discussing Project Methods
Guest was Jon Quigley

Sponsored by

lean marketing lab

a sales and marketing learning community dedicated to the user experience.

main   media   lean marketing system   lean engagement team   tools

We are CONNECTING!

Related Podcast: Managing Scrum of Scrums

Business901 *Podcast Transcription*

Implementing Lean Marketing Systems

Transcription of Interview

Joe Dager: Welcome everyone! This is Joe Dager, the host of the Business901 Podcast. With me today is Jon Quigley. Jon is the principal of Value Transformation, which is a product development training and cost improvement organization. Jon has more than 20 years of product development experience ranging from embedded hardware and software through verification and project management. He is the author of a half a dozen or so books with a new one coming out this spring on configuration management. Jon, I have to start out and ask you, when do you have time to do any work because your books are not exactly a quick airplane read, okay?

Jon Quigley: That is true. You'd be surprised how much you can squeak in if you cut a little bit of TV and some other ancillary things off.

Joe: Well tell me just briefly about your books here. Is there a common thread through them all?

Jon Q: Yes, they're all around product development and managing those processes. For example there's the Project Management of Complex and Embedded System that's an

automotive perspective on how to develop an embedded product, from the time you have the idea through the time that the part is manufactured and delivered to a customer. The Testing book drills down and dwells on the testing aspects of embedded products and different perspectives and different approaches you can take. TQM for Project Management, well that's kind of self-explanatory; how you use those statistical quality tools on your PMO. And the other one's a cost book, and there's the Scrum book. So they're all around managing a product development process.

Joe: Where I ran across your work is first the Scrum Project Management because I was looking how to apply Scrum for multiple project management or maybe what we would call the 'scrum's scrums' or something. Can I just ask that question; can you use Scrum to manage Scrum projects?

Jon Q: You can use Scrum to manage more than just embedded projects and software projects as well, or Agile in general. It need not be what it was originally proposed for or provided, with projects that had a level of complexity and uncertainty to them which is, was, or is largely embedded but not necessarily exclusively software development kind of projects.

Joe: I can use Scrum to manage things outside of just software?

Jon Q: Yes, if you're an entrepreneur, you can use it in those kind of -- the premise of developing an iteration, getting it out there, learning something and altering it is still valid.

Joe: When would you recommend that type of -- when you use Scrum versus let's say a

Waterfall, is there a good rule of thumb where one fits? Not everything should be done in Scrum, should it?

Jon Q: I think you're right there. I would say definitely not everything is Scrum, just like definitely not everything is Waterfall. And it kind of irks me sometimes when I read that on blogs or hear people say that Scrum is the bomb or the thing to replace Waterfall. It's not true, no more than Waterfall could replace Agile. There are certain attributes of your organization and your project that tell you whether you should use Agile or conventional. For example, if I'm working an automotive project that has a lot of tie-ins with heavy capital and manufacturing, distributed all over the world and regulations, I probably would consider that I might consider using a Waterfall approach. The logistics around that are kind of high, the regulations on those are very specific, and proof of conformance to those regulations are very specific, and if something goes wrong, you're going to have to have some track record to traceability, and that might be a little difficult to come by in an Agile incarnation. If you have a team that's a bunch of experts, they're very seasoned that's probably a good time to have an Agile project.

Joe: Are you're saying that Waterfall would provide more structure of things for the inexperienced team, can I take that out of that or --?

Jon Q: Yes, that's one of the things you could take out of that and then the other would be the fact that it's so structured gives you in theory a traceability and so at the end of the day, if the government came to call on you because your product may have harmed somebody, you have some track record or some traceability that you may not quite have

that level of detail if you're doing Agile. By definition, you forego detailed documentation.

Joe: *Well you've actually used Agile in verification and interface it with conventional projects, how did you do that?*

Jon Q: Well what we did is in that instance that was a regulatory project. The talent on deck was fairly seasoned. And in this case, we put the specs out, as in we knew what the specs were for the scope of the project, and then assigned -- I actually had the team pick based on their talent area the pieces of the specifications that belonged where. There were a hundred different requirements and documents; it was more than 3,000 different test cases. And for example you have a guy testing on a truck, you have a lady testing on the hardware and the loop rig, those kind of things.

They would divide the specs up according to what was best for what spot and here's the concept of self-directed work team; I did not dictate that. I was more like a scrum master. Okay, here's what's on deck. Here're the most important parts, the highest priority things we need to verify. Let's divide this up and let's start working it. And we actually had a burned-down chart of the number of test cases per each spec. It didn't quite work out like a burned-down chart in that it was not over two or four weeks, it was more like a six-week period based on the amount of time we thought it would take to do the entire system.

And they got divided up, they ran the test cases, we convened every morning and talked about the three things: what did you do yesterday, what are you going to do today, and what's in the way? And I tracked down, they tracked what they did for test cases, and we

plotted that on our burned-down chart, that's like a burned-up chart I think in that the slope was positive and not negative, instead of ours it was number of test cases conducted.

But they were all kind of elements, stolen if you would from Agile. And this part, this part of the project interfaced with a conventional, more staged gate, definitely a Waterfall kind of project. And every day or two, I would give them, the chief project manager the result of the test cases. As in, here's our planned course, our ramped up test case per day executed, here's what we actually did, and here's what we found in terms of problems and that's not necessarily an Agile thing, that's just a test thing.

Joe: Even though we write books on project management and PMO is all very structured, but it sounds like that the project manager should have flexibility in deciding how a project should run. There's not a prescription there to make a project successful every time.

Jon Q: That's right, there is not. There are fundamentals that need to be covered and sometimes the fundamentals you should focus on differs. It depends on your team; it depends on the situation. By definition, Pert projects have a high variability content. You can do something -- say I finished my cluster project for automobile, the next one is going to be different; maybe a different technology. It certainly would be a different team, maybe a different supplier, a different schedule. All these things mean that getting something from a book as a checklist is only going to partially help.

Joe: What would be some of the challenges of scaling up Scrum then? Are they the same that we have in scaling up conventional projects?

Jon Q: That's definitely true. The specific one is the -- what I find in my industry experience has been the desire to do as much as you can with a little as possible when it comes to human resource, or to just keep adding plus one and you have to take great care in that. One of the reasons why Agile works is you put people in the position to be hyper-focused on a set of activities or a set of objectives actually is more like it, a set of objectives and they have wiggle room to manage to get to those objectives. If there's a swamp on the right, then they make a turn to the left; those kind of things. Now if you have somebody who's participating in numerous projects, they have only a chance of running one direction. It's not a lot of think this thing true and try and find alternatives. There's a Harvard Business Review article, 6 Myths of Product Development. It talks about for knowledge management; you should not book your resource beyond a certain capacity, and that is not 100%. In fact, it's closer to 70 or 80%. The theory is, why you would book them, utilize your resource or your talent to 100% is you get the most out of it but the problem is, things do not go according to the easy path that you think they will and that means you're going to need time to navigate the waters of the sea. As you find these traumas, you're going to have to have time to figure a way around them and then execute these alternate plans. If you're booking your resource close to 100% or your talent close to 100%, then you don't have that time. So the same problems you have with conventional projects, you're going to have with Agile if you try and divert the focus of the people doing the work. Other than that, I think that's about the only limitation you're going to have.

Joe: *Well Agile is a pretty big umbrella and Scrum to me is kind of a subset of Agile, would that be correct in my thinking?*

Jon Q: Yes.

Joe: Scrum seems to be more time-based that I'm going to deliver what I can deliver in this 7-days, 14-days sprint.

Jon Q: Yes.

Joe: That means I have this razor focus for that time period, right?

Jon Q: Yes.

Joe: I think of multiple Scrum projects, so that means I only have a razor focus for two hours a day on that project, two hours a day on that project, and where does my slack come in? I mean is it within the two hours, I always should be working an hour of it and 30 minutes of slack? Should that be the way I figure it?

Jon Q: That's a good question. I'm not sure how that would be handled actually. I think most of the time when you have these projects, your estimates are what they are. The content you're going to put into your sprint is what you think you can put in there. But as you learn more, that might not be true. In my experience, it's not been treated as slack; it's been treated more like an estimate and estimates can be wrong. Which means the content of the sprint delivery may not be exactly what you thought it was going to be at the end, which is going to take some time talking with your project sponsor, the project owner. So

I'm not sure how you would handle the concept of slack in this context.

Joe: *The other part I think of is that is there ever a time you work on a project that you're saying, "Gee, we estimated too much…" I mean I never hear anybody talking about that, I have excess time.*

Jon Q: Yes, that does seldom seem to happen, doesn't it? But in theory, I think I've actually seen it maybe happen once, but that's pretty infrequent I would think. In theory, you just add more of what you can to scope in this thing. If you look like you're going to beat your estimates, and you have some sufficient time, then you can start talking to your product owner, what else can we put in here from that backlog that could be interesting? What's your next priority and see if we fit some parts of that on the way?

Joe: *The problem I have sometimes with Sprint, it seems that most people, a lot of teams will work to the level of expectation. They're not let's just say exceeding expectations sometimes. So if I'm given 14 days to do something, it takes 14 days. Or if it's acceptable in my organization for a 14 day to turn into a 16-day project, well then it's delivered in 14 to 16 days. And that is sometimes taboo in speaking that especially in Lean terms because the respect for people that are always trying to do their best in everything. But people seem to work to that level sometimes. Is that a bad way to look at it? Is that a bad way for a manager to look at the system there?*

Jon Q: I don't think that's a bad way, but it does have some truth to it. I think it's called Parkinson's Law or something similar to that that says the work expands to the amount of

time you have to do it. But as extreme as that is, it's neither good to tell somebody who's a professional that it's going to take six hours to do something that they think is going to take two days. So somewhere in the middle, and the way to really understand that I think is to find a measure, a measurement, a metric that helps you predict to that instead of hanging your hat on the estimate as being law. You really don't know; we think that might be, and we're using expert opinion, but that's what burned-down charts are for. That's why I used a burned-up chart in the case of a number of test cases executed per day. I thought it could look like 'A' or the team that I thought it could look like. This is the number we can execute per day, per person, per hour kind of thing and sometimes we were close and sometimes we were not. But the metric itself told us whether our estimates were right or wrong, rather than people driving to meet a goal or expected time.

Joe: Okay but the important part I would assume is getting the team to agree upon that and be part of the agreement of the hours that it's going to take. I mean that's the most important part of it, right?

Jon Q: That is the very starting point, exactly. I've been in meetings where executives told me something was going to take two hours, and I knew there was no way that was going to be two hours or two days and I walked away feeling kind of -- what's the word, I felt less than motivated. But yes and that's not a trivial task getting the team to buy into the -- obviously, there are going to be people who think that it's going to take longer. They're going to be a little pessimistic maybe and there are some people who maybe lack a little bit of domain experience, and their estimates are going to be skewed the other way.

Joe: When I'm managing or going to have a scrum of scrum meetings, how is that meeting structured differently than your typical Scrum sprint maybe?

Jon Q: Well in the scrum of scrum meeting, your objective is to coordinate the activities of the other scrum participants. So instead of focusing on the internal part of an individual team, you're working more on the interfaces between those teams. Who's expecting what, how, what constitutes good, what constitutes closure or success, and how those interfaces work. So it's largely the same, except for that.

Joe: Are you having daily stand-ups when you have scrum of scrum meetings or is that sometimes more spread out?

Jon Q: I've seen both ways where if it's very dynamic, then they can be daily. I've been seeing like every other day, like Monday, Wednesday, and Friday kind of view for a scrum of scrum.

Joe: And one of the things I think of is there are certain team members that are valued higher than others, and they're on multiple scrums. And so is the scrum of scrum, is that meeting about fighting for that guy, and that guy's time a little bit?

Jon Q: Whoever that person is, yes, you're going to have to find time to secure them. Now that's not a long meeting, it's just like the other principles of Agile where it's a stand-up kind of thing. But you still have to find time, the hour, whatever it takes to be prepared and come back and distribute what is learned. So you're going to have to focus on that, whether it's

one individual but that may not be always the case. It could be that the topic area at the scrum of scrum meeting particularly touches on one person, and it may not be the person with the most rounded talent and that person may vary. It could be a software delivery, and that person is a test person. It could be one of those people that need to be involved. So it's not necessarily that one person would be burdened with the requirement to participate in this meeting. It's whoever's talent is best suited for the topic area at the scrum of scrum.

Joe: Yes and I would think that would be very dynamic, that it could be the testing guy would have nothing to do the first couple of weeks and then the last couple weeks all the projects are ready for testing.

Jon Q: Yes, it could be something like that exactly. Or it could be that the testing guy is missing some large systemic, system level information, and maybe they need to go to this thing at the beginning; it's a little tough to say. I think we talked about it earlier how it's not a prescription, it's an adaption. What do you know, what did you learn, and how do you apply those things? In this case, maybe it's a test guy, maybe it's a software person that needs to go, maybe it's even the Scrum master.

Joe: We've talked about selecting different participants in balancing out a Sprint, are team members dynamic or are we moving them around into the different positions of where they're needed and they're participating in different sprints for the week, is that what that scrum of scrum is all about is making sure my talents are distributed correctly?

Jon Q: It's more about making sure the information travels in a good way. When it also has

some elements of talent distribution in that you wouldn't want to burden one person; if somebody's very busy with certain tasks or objectives within the sprint, that may not be the best person to send to a scrum of scrum. If their objective is in the balance, as in not sure we're going to make it, then sending them is probably not necessarily the best solution. On the other hand, you got to balance that against what information do I need from the scrum of scrums to bring to this sprint to make sure that we continue to move together. Because one of the big problems with conventional project management in my mind is that it drills communications channels and the quality and timeliness of the information that comes into the project team, whatever executing team that is, whether it's the test team or the coding folks, that's a problem area for conventional projects and the larger the project, the more distributed the project, the more difficult that communication is to manage. So for me, the scrum of scrums is more about a quick and concise way of distributing communications from the top level part of the project from the hierarchical terms to the other; this is the mechanism that does that.

Joe: One of the things I found really interesting is that you actually PERT in a Scrum book. You're the first person I've seen do that. Why did you do that? Was that just kind of intermingling or why did you do that and what's the purpose of it?

Jon Q: Well as I said earlier, these people that think conventional project management is dead and is not effective or that think Agile is the be-all-end-all, that's not true. And in fact in my mind, for example in conventional project management -- let's look at that for just a second, in conventional project management nobody tells a PM that he needs to stay in his ivory tower. When people say Agile is good because the team gets together every day and

talk about these three things, well nobody tells a conventional project manager not to do that. Nobody tells a PM that gap should not go to your team, sit down with them for 15 minutes and talk about those, and ask them those three questions. So that's not exclusive to Agile, though it is exclusive to Agile so to say. So in my mind, whatever the tool is that could help you understand what you're up against, that's the tool to use and PERT works. It at lets gives you an envelope of what's possible as far as that estimate.

What I personally have a problem with, though I'm frequently asked to do is what I call point source estimates. A lot of times, without a lot of the scope of what success means, estimate what it takes to do this but I'm not going to tell you what I think success looks like in that estimate. That's what conventional project management often is like. But point source dates are very difficult. I was in a meeting one time with a project manager who came to me and it was a test related discussion. He was having a vehicle moved to a certain facility so that we could run tests on it. The vehicle was coming from a facility we've gotten vehicles from before, it was what we call a 'mule' in automotive terms which meant it was whatever was left, and he wanted me to do systems testing on the first day of his schedule. He said I needed you to start systems testing the first day that the truck, the vehicle got there. And is aid, well I have a distribution curve that says if you want me to start testing on this vehicle the first day, the probability of success is very low. I have no population of vehicle received from that place where I started on it, where my group could start on it in one day. The best was three days, and the worst was three weeks to make the vehicle system ready for testing, and that guy didn't like to hear that. But the reality is, that's what every project estimate is. It's a variable from three to three weeks or from X to X plus time.

So when I thought of Agile, I thought well with this planning poker, okay that's still guesstimated just like everything else, but PERT is not altogether not without merit. It tells you the range of possibilities. And if your range is too large, then perhaps you need to start decomposing what you mean by that task or that objective. It was intentional putting that in there. For me, conventional project management can benefit from Agile practices, and likewise Agile can benefit from conventional project management techniques. You seem to know when to mash them together.

Joe: *Well the other thing you mashed together in the book yourself is Lean and Six Sigma, and how does Lean thinking -- you know I think Scrum kind of evolved from Lean thinking, but how do you view Lean and Scrum?*

Jon Q: Its the project version of Lean; strip away all the clutter, strip away all the unimportant stuff and focus on what is very important. So for me, it's incarnation of all our projects.

Joe: *So you're saying that Scrum is really the Lean project tool of -- I don't want to say of choice maybe, but certainly of choice because it's that razor-like focus for that certain time period.*

Jon Q: Yes. I would say you could apply Lean principles to conventional project management but it's not the same thing, and there may be times when you have to do that. Those limitations we talked about earlier, why you would pick conventional over Scrum, for example, for your project management, well and you apply Lean techniques to tighten it up.

But if you're looking from a Lean project approach, that's Scrum.

Joe: *In one of our books, I think you very much was Lean-orientated in that it reduces the cost of a project. Could you tell me a little bit about that and what that book is about?*

Jon Q: Well, TQM for Project Management, that's more about reducing the cost of the project, PMO. The Reducing Process for Lean, Six Sigma, and Value Engineering Techniques, that's more of an operating kind of book, although there are some PMO aspects in there. But the TQM book is about using those total quality management techniques to understand where your PMO costs are, what in your PMO is most important and fundamental and contributes to your cost, and how to put those things under better control.

Joe: *Your new book, that's coming out this spring, is on configuration management. Can you tell me about that?*

Jon Q: It's coming out in May I believe, and it's about theory and practice of configuration management. It's more than just a technical book on CM; it's designed to show you that CM is a lot more places than you think. It's not just manufacturing part. For example, when you're doing a developmental project, designing a new product, you're getting input from marketing. Marketing is doing measurements - I hope, with their customers understanding what is important; that's where you get your requirements. Well, the marketing guys are using surveys and studies and things like that, well are those documents that they're doing getting these surveys and what-not from, are they under control? Are they known? Can we directly trace what our work will be downstream to some upstream something from the

customer? I don't think CM is typically seen in that light. It's mostly in, well what do we deliver into manufacturing? Can we trace it back to what we did before, the changes and things like that? It's not so much focused on the whole stream of configuration opportunities, but you better have under control if you want to build at the end, produce at the end what you said you were trying to do at the beginning.

Joe: *Is there some telltale signs besides just someone raising their hand or early in the process or a time thing, is there some leading indicators that I could see when those problems arise?*

Jon Q: When you're starting to lose that focus and that continuity, a lot of times that comes in the form of a hand raising, unfortunately. I don't know that there are a lot of telltale signs. It's usually when you start to put the parts together that you find out it was a misperception or miscommunication or a misunderstanding of the objective, and the parts don't go together very well. I'd say listen to the people that are in your team; that's as good as it could get. And where you have the possibility having a measurement or a metric that makes them sense to predict and monitor that. Unfortunately, that's not a metric; it's what is important and what do I need to focus on?

For example, is it arrival rate of a certain set of requirements or if I got the rate of execution of the test cases. Some of those external dependencies are a little tougher to track though. One of the points of risk with this with the Agile or Scrum is that's probably one of them, is the risk. Now the risk to schedule, that's not so risky because it's tracked; you're going to know that. But some ancillary risks, the things off to the side -- let me back up for a second.

You're going to know that because you have a burned-down chart that tells you whether you're going to make your schedule or not for this sprint. So some things are knowable, and others, because you're hyper-focused on this execution, are maybe not quite so knowable for your risk management. That becomes less of a forward thinking and more of an adaption to what happens in my mind. In other words, you don't want to see heavy risk management plans in Agile. I have not seen the use of heavy risk management plans in Agile, which leaves you open for something not good to happen.

The plus side is the fact that it's not adaptive kind of methodology is the theories you're going to quickly respond to whatever comes your way and I think if you're going to say where the weakness is in that hyper focus, it will be something like that. But if you're listening to your team, if your team even thinks that, "Oh you know what, this might not go right because of this…" And if you're paying attention to that, you can at least get a little bit of a telegraph of this inbound disaster or would be disaster.

Joe: *Does a Scrum project and Agile project, does that need a scope document?*

Jon Q: That's a good question. I believe that the product backlog is supposed to serve as a scope document, but I'm not sure that's sufficient in all cases. In many cases, it is because those are the particular objectives; the products shall do these things. But I'm not sure that solely addresses it every time.

Joe: *So you'd recommend maybe having a scope document and then the backlog and the users stories making up the scope document?*

Jon Q: Yes, that would round it out. And in the scope document, you'd have some things about those side possible risks and things like that. Narrowing on the objective or the latitude on the objective where you can. You have the variants. Now the product owner is supposed to be there to answer a lot of those questions as they arise in the middle of the sprint but a lot of times, finding a project owner when you need them is not as easy as you think.

Joe: How can someone contact you and find out more about the books? Should they go just to the Website?

Jon Q: They can go to the Website, there's a store. On the download section, I have my business card, so somebody could go and download the business card and give me a call.

Joe: And the Website's name?

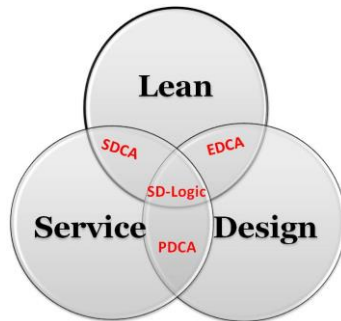**Jon Q:** www.valuetransform.com.

Joe: Well I'd like to thank you very much Jon. I appreciate it. This podcast will be available on the Business901 Blog Site and the Business901 iTunes store. Thanks again Jon.

Jon Q: Thank you very much, Joe.

Business901   *Podcast Transcription*

Implementing Lean Marketing Systems

Joseph T. Dager

Business901

Phone: 260-918-0438

Skype: Biz901

Fax: 260-818-2022

Email: jtdager@business901.com

Website: http://www.business901.com

Twitter: @business901

Joe Dager is President of Business901, a firm specializing in bringing the continuous improvement process to the sales and marketing arena. He takes his process thinking of over thirty years in marketing within a wide variety of industries and applies it through Lean Marketing and Lean Service Design.

Visit the Lean Marketing Lab: Being part of this community will allow you to interact with like-minded individuals and organizations, purchase related tools, use some free ones and receive feedback from your peers.