Application of Lean Product and Process Development in FIRST Robotics Competition

Bas Koomen[1,2][0000-0001-8253-6084] and Ron Visser[3,4]

[1] University of Twente, Enschede, The Netherlands
[2] Cadmes b.v., ‘s-Hertogenbosch, The Netherlands
[3] MTA b.v., Helmond, The Netherlands
[4] Team Rembrandts STEAM Foundation, Eindhoven, The Netherlands

s.p.koomen@utwente.nl, r.visser@m-t-a.nl

Abstract. In FIRST (For Inspiration and Recognition of Science and Technology) Robotics Competition, students have to design, build, and test a competition robot during a building season of six weeks. Lean Product and Process Development promises to shorten product development times and increase knowledge reuse. There is a knowledge gap for the application of Lean Product and Process Development in the context of student competitions. In this paper, we outline an approach to apply Lean Product and Process Development during the preparation and the building season. We hypothesize that the students can front-load knowledge in problem-based learning cycles before the game is published. Once the game is published, students can apply the front-loaded knowledge for the specific requirements of the game. The proposed approach includes an organizational structure, processes, and the use of Product Lifecycle Management software. We are going to test the approach with a larger FIRST Robotics Competition team in The Netherlands. The expected results of this case study are an increased insight in the effectiveness of Lean Product and Process Development and a measurable difference with the traditional design approach. Future research needs to be done on the results of this case study. Also, more similar case studies can be performed to obtain more general knowledge about the effectivity of the methodology.

Keywords. Lean Product and Process Development, Problem Based Learning, FIRST Robotics Competition, Set Based Concurrent Engineering.

1 Introduction

Educators are looking for ways to present modern industrial approaches to students. Problem-Based Learning is a commonly accepted approach, in which content and practice are holistically integrated [1].

For Lean Manufacturing, Tortorella and Cauchick-Miquel have proposed an initiative with industrial engineering graduates, based on Problem-Based Learning [2]. Their research paper references other research on the benefits of Problem-Based Learning and what the optimal circumstance should be. They conclude that a Problem-Based Learning approach enhances the ability of students to acquire and apply knowledge in real-world situations. Moreover, they can better meet the current demands of organizations and academia.

The challenge in Problem-Based Learning is to find complex, real-life projects that allow students to apprehend and experience the content and enables them to reflect on it within a limited timeframe. If real projects from companies or organizations are not available or feasible within the available time, educators have to use fictive problems.

With Lean Product and Process Development (LPPD), as defined by Ward and Sobek II [3], it is a challenge to find real-life problems in companies. Not many companies are looking into this theory yet, and LPPD projects involve a more profound organizational change before successes are achieved [4]. The consequence is that educators have to divert to fictive problems or serious games [5], and therefore compromise on the real-life aspect of Problem-Based Learning.

We identified the combination of the aforementioned issues (the lack of suitable cases for Problem-Based Learning with LPPD and the complexity of LPPD introduction in real organizations) as a limiting factor to let students experience and investigate LPPD. Therefore, we are using a student robotics competition program as an a context instead of industrial problems.

In this paper, we propose to apply LPPD in *FIRST* Robotics Competition (FRC) [6] (*FIRST* is an acronym for "For Inspiration and Recognition of Science and Technology"). In FRC, student teams collaboratively design, test, and build a robot (see Fig. 1 for an example) within six weeks, based on requirements that change every year. FRC has a high level of complexity and includes mechanical, electrical, and software engineering. The benefit of LPPD is that the students can prepare reusable knowledge in the months preceding the six weeks of the challenge.

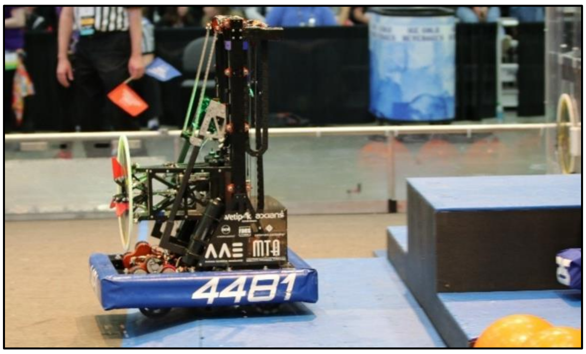


Fig. 1. FRC Robot (FRC team 4481, Season 2019)

We will verify the proposed methodology in a case study, with the Dutch FRC team #4481 (Team Rembrandts), that will take place between August 2019 and May 2020, during which we will measure the effectivity of LPPD. The results of this case study contribute to a better understanding of LPPD in an educational context.

The remainder of the paper has the following structure: in Section 2, we elaborate more on the research question and the context. Section 3 explains the LPPD methodology that we will use in the case study. The implementation plan for the case study is described in Section 4 and the measurement of the expected results in Section 5. Section 6 contains the conclusion and a brief outlook to future research.

2 Research question

FRC teams work intensely on the design of their robot during the building season lasts only six weeks. Before the start of this period, the exact requirements for the game are unknown. Therefore, it is a challenge to let students work on relevant engineering problems before the announcement of the game requirements.

From experience, we know that the six week period allows for only one design iteration, with little time for prototyping. When the team participates in a regional tournament, the first real test of the design is done. Only if the team participates in a second regional tournament, or qualifies for the World Championships, a next design iteration might take place.

Teams have to make many design decisions in different areas. For each of these areas, knowledge gaps emerge during the building season. Below, we give two examples:

1) *A shooter that has to shoot (or throw) an object in or on a specific target.* This feature was needed in most of the previous seasons in various forms. The objects have been: balls in various weights and sizes, cubes, frisbees, and crates. The shooter problem is very suitable for knowledge creation in an extensive range of requirements since it is unknown upfront what the exact rules of next year’s game will be.

Fig. 2. Ball shooter module (FRC team 4481, Season 2016)

2) A drive train that supports the robot and enables it to move around. Here the team can decide to use tracks, tank-drive wheels, omni-directional (Mecanum) wheels, swerve drive, or other solutions. The robot always needs a drive train, and the rules are mostly known. On the other hand, the best fit of the characteristics with the game is unknown upfront. In some years, an

agile and fast drive train is optimal. In other years, a sturdy drive train with much grip is better.

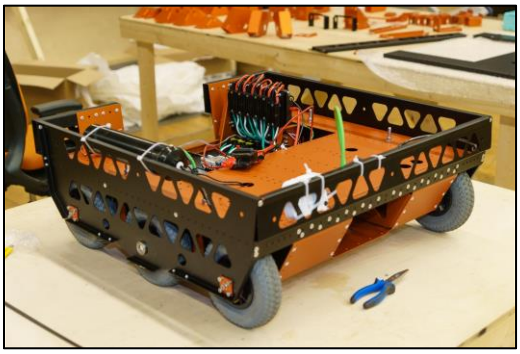


Fig. 3. Drive train (FRC team 4481, Season 2016)

Following the LPPD principles, it should be possible to front-load knowledge of technical solutions for typical challenges in the FRC game structure. Set-Based Concurrent Engineering on sub-problems - with potential ranges of requirements - could enable more design iterations. Reusable knowledge should emerge from these iterations.

Hence, our research question is: *"How can FRC teams use LPPD in order to improve the design outcome during the six weeks of the building season and overcome current knowledge creation and knowledge management challenges?"*

3 Lean Product and Process Development methodology

It is essential to give a clear definition of LPPD to understand the proposed approach. We based our approach on the theories of Ward and Sobek II [3]. Subsequently, M. Kennedy [4], B. Kennedy et al. [7], and Cloft et al. [8] have elaborated on these theories and came up with practical ways to apply LPPD in design processes. In the next paragraphs, we highlight the four pillars of LPPD (Fig. 4), according to Ward and Sobek II [3], and describe how we will apply them in the FRC context.

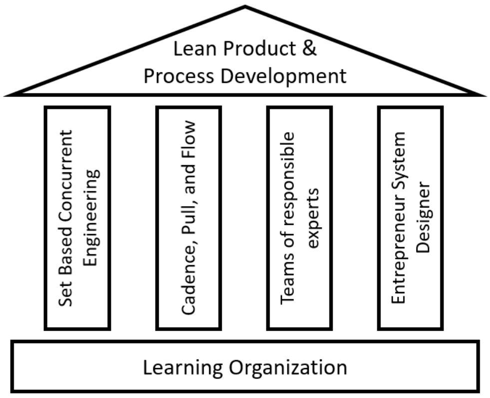


Fig. 4. Ward's "four-pillar model" of LPPD. (modified from [9])

3.1 Set-Based Concurrent Engineering

We can divide an FRC robot into subsystems. Some subsystems will be relevant in each season, like the drive train, steering control, or vision systems. Other subsystems are dependent on the specific game challenges of a season. Here, the team can learn from the past seasons which elements are likely to return in the next season. Systems we have seen are: lifting mechanisms for crates, shooters for balls or frisbees, intake mechanisms for balls or crates, handling devices for circular discs, and some other exotic features. Over the years, the games have used several variations on similar principles. For example, many games had the challenge to shoot balls, but there was variation in the size and material properties of the balls.

In a "traditional" design approach, as schools teach it to most engineering students, alternative solutions to a problem are evaluated in the shortest possible time. One candidate solution is selected, using various selection mechanisms. From that point on, the design process is focused on making the selected option work, with regular unexpected setbacks and rework. Most rework is the result of knowledge gaps, earlier in the process.

In Set-Based Concurrent Engineering, it is the aim to delay the decision, until enough knowledge is gathered to make it safely. Cloft et al. [8] explain how causal maps can help to visualize the decision-making process for a (sub)system. In the example of a ball shooter, the relation between all kind of attributes and functions can be visualized, like target accuracy, ball speed, exit angle, ball spin, ball diameter, ball weight, and ball stiffness.

3.2 Cadence, pull and flow.

Knowledge is required to make the decisions in the causal map. If not enough knowledge is available to make a decision, a knowledge gap is identified. Once a knowledge gap exists, actions have to be taken to fill the knowledge gap. Consequently, the team will execute the actions in the form of research or tests, and make the resulting knowledge available. In the example of the ball shooter, experiments can be done using different types of balls, different speeds, different angles, different shooter models, different ball spin, and other physical behavior.

The need to fill knowledge gaps ensures the pulling behavior of the process: knowledge is pulled from a need, arising from the causal map. To create cadence, we propose to use regular team meetings to discuss and update the causal map. The frequency outside the six weeks building season can be in terms of once per two or three weeks, inside the building season twice per week. These meetings are integration events, where the acquired knowledge is put together in context.

Furthermore, the team needs to create a constant flow of knowledge. The team can record knowledge in trade-off curves [3]. Again, in the ball shooter example, the effect of backspin of the ball on the target accuracy can be measured for different diameters of balls. Multiple lines in a graph can visualize this knowledge.

Sobek II and Smalley [10] describe how knowledge can be captured in A3 knowledge briefs (K-briefs). The K-brief describes the problem, explains the physics of the problem, describes solution proposals, and describes the decision-making process. These K-briefs may include one or more trade-off curves for generic robot solutions.

The role of the K-briefs is different in the six week building season. In this time frame, the decisions are made specifically for the game challenges. In the K-brief, the team members record the rationale for the final decisions for the game robot. Later, this rationale is reusable as knowledge for the next generation.

3.3 Team of responsible experts.

Each team member, in his or her specialty, needs to contribute to the overall success. It is not enough to concentrate only on the content of a specific task [3]. Members are expected to collaborate within and across the sub-team and know the context of their work. In the regular integration events, team members will share their findings and listen to others. Asking (why) questions is vital to get to the core of problems.

3.4 Entrepreneur System Designer

Ward and Sobek II [3] introduce the role of the *Entrepreneur System Designer* in the process of LPPD. The Entrepreneur System Designer has a central position in the design team. His responsibility is to keep the focus on the (real) customer interest and the causal map with all design decisions. For FRC, each subsystem design team could have a “subsystem Entrepreneur System Designer” who will focus on potential requirements for the subsystem. Furthermore, the team appoints an Entrepreneur System Designer for the entire robot. This “system Entrepreneur System Designer” will be responsible for the integration of subsystem knowledge for the specific game requirements of a particular season.

4 Implementation plan

For the experiment with an FRC team in a real game season, we have defined the following implementation steps:

4.1 Create awareness among mentors

The FRC consists of students and mentors. Mentors generally are more experienced engineers. Most of them have participated in FRC in previous years, and gathered experience with the FRC game structure.

To create awareness among the mentors, we will start with some Set-Based Concurrent Engineering and causal map exercises, to experience the contribution effect of this method. Potentially, the Set-Based Concurrent Engineering Serious Game by Kerga et al. [5] could be suitable for this phase.

4.2 Define subsystems

Next, the team will decide which subsystems they will identify as part of a generic robot. These subsystems must be relatively independent from other subsystems to enable focused prototyping and learning. Also, the number of subsystems should be appropriate for the size of the team. A larger team can work on more subsystems simultaneously than a smaller team.

4.3 Establish sub-teams (roles)

For each subsystem, the team will establish a sub-team. Each sub-team needs one leader: the subsystem Entrepreneur System Designer. The size of the team can be depending on the type of subsystem. Some subsystems require a single discipline, like mechanical experts. Other subsystems may also need software and hardware experts.

It may be smart to initiate a dedicated game strategy team for scoring trade-off knowledge. This sub-team might even include experts on game theory [11].

4.4 Train team members in the methodology

When the teams are established, the members need to be trained. The training will focus on the specific elements in the methodology:

- Causal maps. The members will learn how they can build a causal map for a specific design problem. The process is important, how to discuss and generate required decisions.
- Knowledge gaps. From the causal map, knowledge gaps emerge. The members are trained on how to approach the knowledge gap and learn to come up with the right questions.

- A3 K-briefs. The team members need to learn how to write a useable A3 K-brief. We will proved a predefined structure.
- Integration events. We will instruct the sub-teams, how to organize an integration event for the subsystem and the entire robot system.

4.5 Establish knowledge platform

It is crucial to establish a common platform to make the acquired knowledge available. The platform could be relatively low-tech, where cloud storage is used to host digital A3 K-briefs, causal maps, and trade-off curves, or a more advanced approach with industry-standard collaboration platforms.

For this case study, Dassault Systèmes will make the 3DExperience Platform [12] available in the cloud for the FRC team. The team has already experience with SOLIDWORKS 3D CAD and virtual prototyping, which will be used more extensively during the case study.

4.6 Coach teams to work LPPD-style

After training, the team will start their work of knowledge creation, ahead of the six week season start. The team will need coaching of LPPD practitioners to keep the required methodological quality.

5 Measuring the results

We need to measure two results from the experiment to answer the research question: the quality of the design outcome, and the performance of knowledge creation

5.1 Design outcome

We can measure the design outcome by the relative performance of the robot in the competition. The organization tracks many scoring data on each element of the game. This performance data is known for previous seasons, so after the next season, we can measure if there is an improvement. In Fig. 5, there is an example of such data from the 2018 game.

With the statistical data, the team can measure their relative performance compared to other teams and the performance compared to the potential maximum performance. If this method is valid, a change should be noticeable on longer-term (multiple years from now).

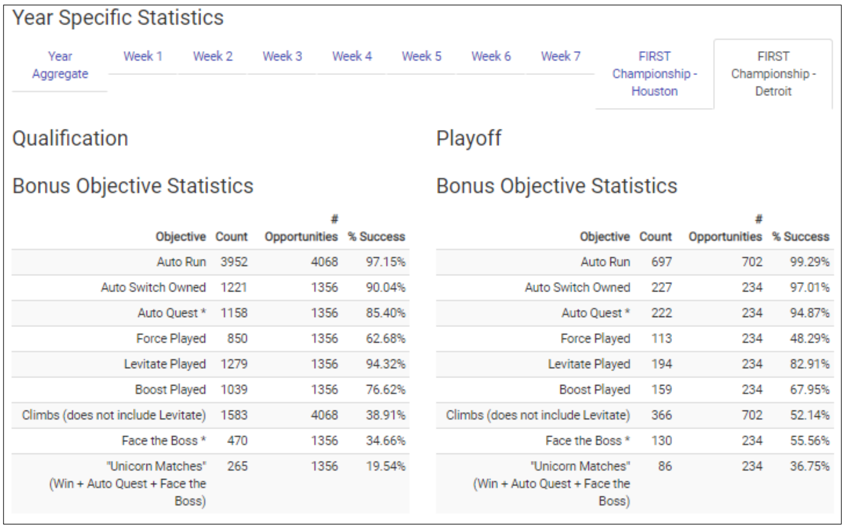
5.2 Knowledge creation and management.

Thomke and Fujimoto [14] have investigated the effect of “Front-Loading Problem-Solving” at Toyota. They identify two mechanisms of front-loading:

- Project-to-project knowledge transfer, which can be measured by the number of k-briefs that is created to transfer knowledge, or the number of k-briefs that can be used during the six week building season.
- Rapid problem-solving, which is a mechanism of rapid learning, where team members perform many tests (physical or virtual) on a large number of alternatives. Rapid learning would optimally result in a large number of trade-off curves, which can be measured.

Effect of this case study should be measurable in the amount of documented knowledge (A3 k-briefs, causal maps), and the interaction during integration events.

A specific aspect of FIRST competitions is that the organization encourage teams to proliferate their knowledge. LPPD offers a suitable context to improve this, so the knowledge transfer between teams could also be measured if more teams would adopt this methodology.



Qualification

Bonus Objective Statistics

Objective	Count	# Opportunities	% Success
Auto Run	3952	4068	97.15%
Auto Switch Owned	1221	1356	90.04%
Auto Quest *	1158	1356	85.40%
Force Played	850	1356	62.68%
Levitate Played	1279	1356	94.32%
Boost Played	1039	1356	76.62%
Climbs (does not include Levitate)	1583	4068	38.91%
Face the Boss *	470	1356	34.66%
"Unicorn Matches" (Win + Auto Quest + Face the Boss)	265	1356	19.54%

Playoff

Bonus Objective Statistics

Objective	Count	# Opportunities	% Success
Auto Run	697	702	99.29%
Auto Switch Owned	227	234	97.01%
Auto Quest *	222	234	94.87%
Force Played	113	234	48.29%
Levitate Played	194	234	82.91%
Boost Played	159	234	67.95%
Climbs (does not include Levitate)	366	702	52.14%
Face the Boss *	130	234	55.56%
"Unicorn Matches" (Win + Auto Quest + Face the Boss)	86	234	36.75%

Fig. 5. Relative scoring data vs. potential scoring from 2018 [13].

6 Conclusion and future research

We conclude from this conceptual design of the experiment that FRC is a proper context to let students experience the effect of LPPD. FRC offers several aspects that enable a very realistic environment:

- The design is sophisticated and can be divided into subsystems.
- There is a fixed time frame between the publication of the requirements and product delivery.

- Teams can use the months before the six week building season to front-load knowledge.
- There is an opportunity for learning from the past by investigation of previous seasons.

As a next step, we need to evaluate this case study during and after the experiment. We will need to measure short and long term effects for more definitive conclusions.

References

1. Jones, B.D., et al., *The effects of a collaborative problem-based learning experience on students' motivation in engineering capstone courses.* Interdisciplinary Journal of Problem-based Learning, 2013. **7**(2): p. 2.
2. Tortorella, G. and P.A. Cauchick-Miguel, *An initiative for integrating problem-based learning into a lean manufacturing course of an industrial engineering graduate program.* Producao, 2017. **27**(Specialissue).
3. Ward, A.C. and D.K. Sobek II, *Lean product and process development*, in *Second edition*. 2014, The Lean Enterprise Institute: Cambridge, Mass.
4. Kennedy, M.N., *Product development for the lean enterprise: why Toyota's system is four times more productive and how you can implement it*. 2003: Oaklea Press.
5. Kerga, E., et al., *A serious game for introducing set-based concurrent engineering in industrial practices.* Concurrent Engineering, 2014. **22**(4): p. 333-346.
6. FIRST. *FIRST Robotics Competition*. 2019 [cited 2019 16-07-2019]; Available from: https://www.firstinspires.org/robotics/frc.
7. Kennedy, B.M., D.K. Sobek II, and M.N. Kennedy, *Reducing Rework by Applying Set-Based Practices Early in the Systems Engineering Process.* Systems Engineering, 2014. **17**(3): p. 278-296.
8. Cloft, P.W., M.N. Kennedy, and B.M. Kennedy, *Success is Assured: Satisfy Your Customers On Time and On Budget by Optimizing Decisions Collaboratively Using Reusable Visual Models*. 2018, Cambridge, MA, USA: Productivity Press.
9. Flores, M., et al., *Lean Product Development Best Practices, Ten industrial success stories*. 2017, Carona, Switzerland: Lean Analytics Association.
10. Sobek II, D.K. and A. Smalley, *Understanding A3 Thinking: a critical component of Toyota's PDCA management system*. 2008, New York: Productivity Press.
11. Webb, J.N., *Game theory : decision, interaction, and evolution*. 2007, Springer: [London].
12. Dassault_Systèmes. *3DExperience Platform*. 2019 [cited 2019 1-8-2019]; Available from: https://www.3ds.com/about-3ds/3dexperience-platform/
13. The_Blue_Alliance. *2018 Insights*. 2019 [cited 2019 1-8-2019]; Available from: https://www.thebluealliance.com/insights/2018#year-specific-week-9.
14. Thomke, S. and T. Fujimoto, *The Effect of "Front-Loading" Problem-Solving on Product Development Performance.* Journal of Product Innovation Management, 2000. **17**(2): p. 128-142.