

FROM DBT TO TDB – A NEW APPROACH TO DESIGN-BUILD PROJECTS

G. Gustafsson, D. Raudberget

Abstract— Academic development projects are often carried out in the sequence Design – Build – Test (DBT), resembling the prevalent industrial development process. In this, an early decision is made on which solution to the problem at hand should be developed, after which a prototype is built and tested while attempting to fulfil the market requirements. One of several drawbacks associated with this approach is its premature selection of the best alternative among several unfinished designs. In recent years, an alternative development method has attracted positive attention. This “Lean Product Development” (LPD) philosophy implies that careful attention should be given to the earlier stages of the development process, in order to gain sufficient knowledge to solve the problem. To align academic design project with the LPD philosophy, we propose a shift in design-build experiences from the Design – Build – Test of today to the Test – Design – Build (TDB). This proposal also makes design-build experiences applicable to engineering disciplines where size and cost of prototypes have previously been obstacles to implementation, such as architecture and civil engineering.

Index Terms— Design project, Lean Product Development, TDB, DBT

I. INTRODUCTION

A design-implement project can be described by [1]:
“A design-build-test experience is a learning event where the learning takes place through the creation of a product or system. The product that is created in the learning event should be developed and implemented to a state where it is operationally testable by students in order to verify that it meets its requirements and to identify possible improvements.”

In academia, an often used acronym which designates the prevalent sequence in which these particular events appear in student projects is Design-Build-Test (DBT) [2].

Göran Gustafsson is a senior lecturer in Machine Design at the Chalmers University of Technology in Gothenburg, Sweden. (+46 31 772 13 57; e-mail: gorang@chalmers.se).

Dag Raudberget is a senior lecturer in Product Development at the School of Engineering, Jönköping University, Sweden.

Judging from popular European and North American literature [3-5], this is also their order in most industrial companies. Somewhat simplified and exaggerated, this development process often follows the pattern in figure 1. The development project begins with a detailed specification of the characteristics of the product to be developed, followed by a generation of different ideas of how to solve the problem. The proposed solutions are evaluated at the current state of development, and even though the choice of the best alternative is done in subsequent steps, it tends to be made through subjective evaluations, before the space of possible solutions has been explored. The chosen solution is then further designed and optimised and a prototype is built to test if it fulfills the conditions set out in the specifications.

Drawbacks of a traditional product development process

Despite the popularity of the DBT development process, it is easy to spot several weaknesses and potential sources of difficulties in the process: One difficulty is that if detailed conditions and requirements that have to be met by an approved design are set early in the project, there is an inherent risk that such a solution does not exist or at least is not possible to find within the time allotted [6].

Another weakness is that an early choice of the alternative to develop does not stimulate attempts to investigate the whole space of possible solutions [7], but rather the opposite, which means that better solutions are easily overlooked or even never discovered. A third disadvantage is the classic risk associated with putting all your money on one horse. For if the chosen candidate does not deliver what is expected of it, a loopback (see Figure 1) is necessary in order to find a replacement solution, which almost inevitably delays the project and increases the cost of it.

So, isn't it relevant to ask if there isn't a better way to develop new products than using DBT?

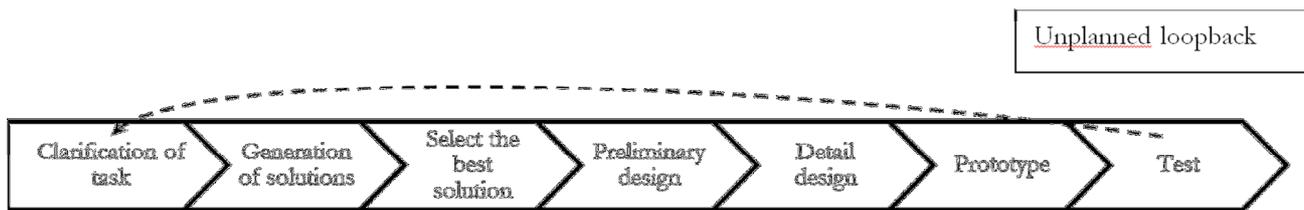


Figure 1: A generic product development process

II. NEW INSIGHTS INTO THE PRODUCT DEVELOPMENT PROCESS

An alternative PD process which is attracting increasing interest is that of Lean Product Development (LPD) [6]. Originally, LPD was a generic name for techniques, methods and a philosophy used in some Japanese industrial companies, notably the car maker Toyota [8]. Now, the acronym also represents developments of these methods in Western companies and universities. The underlying idea in LPD is to use work methods and tools that help reduce all sorts of waste in the product development process, that is, unnecessary work, delays and everything else that slows down work and hampers efficiency.

In our point of view, the LPD ideas are well suited for engineering education, not least in the design-build phase but also for testing. The LPD way of working has a learning focus that provides a foundation upon which deeper conceptual understanding can be built. It starts from the awareness of a problem and a rough idea of what the characteristics of a solution to it are, but no detailed conditions are set. The lack of precise requirements to be met stimulates a search for possible solutions that is not limited to and based on existing designs, but rather the contrary. Experimentation and knowledge discovery become natural if conditions that restrain the imagination are not emphasized. It also becomes obvious and natural to proceed along more than one track and investigate different alternatives in parallel. On basis of knowledge gained by increasingly rigorous evaluation and comparisons – which always deserves to take precedence over preconceived ideas – inferior solutions are successively discarded and eliminated in the PD process until the most promising one is arrived at, securing that all concepts have received proper attention.

The early LPD testing also reveals any remaining knowledge gaps that have to be closed. To commence detailed design before the technology employed is fully mastered is like introducing a time-bomb in the process which may go off at any time and create chaos. Closing the knowledge gaps greatly reduces the risk of an unpleasant later discovery that the chosen concept cannot be realized for one reason or another, and it thus minimizes the need for loopbacks. The LPD approach thus avoids delays in the PD process caused by overoptimistic designs and lack of knowledge necessary to deal with the chosen technology. It is therefore fair to question whether the “standard” PD process model is really the best approach to learning design, and even if it should be taught at

all in engineering education.

The rediscovery of a product development process

The emerging interest for the LPD type of processes may indicate that it is a new type of product development method, but that is not the case at all. In the early 20th century, the Wright brothers employed an LPD type strategy when they designed their first pioneering airplane. The Wrights, who ran a bicycle repair shop and had taken an interest in aeronautics, identified the three fundamental problems of flying: lift, control and propulsion. These all had to be solved, and the Wrights did not proceed into designing a complete system until they had the knowledge necessary to master them.

The Wright brothers designed their own laboratory equipment, such as a wind tunnel and balances, which they used for advanced aerodynamic experimentation. The data they gained was remarkably accurate, which can be exemplified by their propeller designs. The Wright propeller had near optimal design and was superior to their competitors’ designs [9], which relied on existing heuristics for steam-boat propellers that worked well in water but not in air.

III. DESCRIPTION OF THE TEST-DESIGN-BUILD PROCESS

The proposed sequence of events for Design – Build experiences is Test – Design – Build, TDB, rather than the common Design – Build – Test, DBT. TDB resembles the Lean design process used by the Wrights. Some of the differences between TDB and DBT are that with the former many tests are carried out early, and in order to learn, while DBT is associated with fewer tests later and mainly in order to verify that the design requirements are met. Early tests are cheaper though, when they can also be carried through all the way to failure in order to find out what the limits of the chosen technology are. Furthermore, if something doesn’t work, it is much better to discover that early than later, when a loopback – which causes a delay – is inevitable in order to find another (working) alternative.

The TDB and the DBT processes are illustrated in Figure 2, where early – and relatively inexpensive – knowledge accumulation with simple simulations, experiments and models is contrasted with late verification of requirements specifications with expensive prototypes.

It should be pointed out that putting the T in the beginning instead of at the end does not imply that TDB *excludes* late tests or makes them unnecessary. Not at all, but as explained above, a much larger share of the total practical work is dedicated to early investigations and simple and inexpensive

experiments. The late DBT testing in the “standard” PD process is of course also about learning, but testing to specifications doesn’t generate much useful knowledge. At best it confirms that the chosen design fulfills the requirements at hand, but how does one know that these exactly reflect the conditions that the real application will meet? It is much better to try to find the *limits* of what the design can withstand.

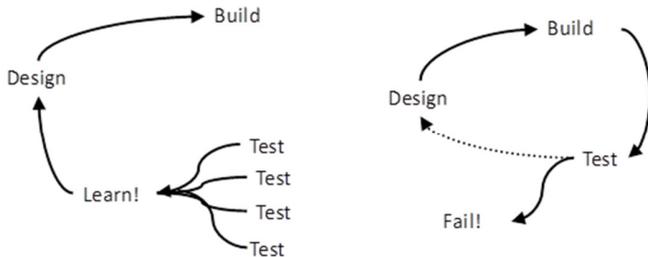


Figure 2 The TDB and the DBT processes.

One aspect of design-build experiences is that the difficulty of implementing them differs a lot between engineering disciplines. In Architecture and Civil Engineering education the cost and size of prototypes have previously been obstacles to running DBT projects. In these disciplines the teachers have, with few exceptions, used digital representations or scale models of products and systems. With the TDB approach however, also architects-to-be and CE students can carry out simple experiments to evaluate different versions of a window layout, to experience firsthand the effects of alternative exterior design.

The Test-Design-Build process focuses on learning

The TDB process reinforces the design-build experience as described in [1]: “A design-build-test experience is a learning event ... The product ... should be developed and implemented to a state where it is operationally testable by students in order to verify that it meets its requirements and to identify possible improvements.”

This can be compared with the Wright’s development practice. They flew only after they had mastered all the individual technical challenges in the form of propulsion, lift and control. Their testing and experimental work was extensive and carried out before system design.

The TDB approach intends to focus on *learning about the alternatives* in order to find out which of the conceived solutions to the problem that are possible to realize and to close the knowledge gaps. Since the technology employed has already been thoroughly investigated in earlier tests, the final testing is focused on verifying what is basically already known, which means that the level of confidence that things will work at this stage is much higher than what is common in the DBT process. Some differences between “ordinary” DBT and TDB have already been mentioned, and Table 1 is a more extensive account of them.

	DBT	TDB
<i>Number of alternatives tested</i>	One or few	Many
<i>Time frame</i>	Tests are carried out late in the process	Tests are carried out early
<i>Reason for tests</i>	Verify compliance with requirements	Exploratory testing in order to learn and find design limits or trade-offs
<i>Models used</i>	Detailed prototypes and advanced models	Simple experiments
<i>Cost</i>	High	Low, initially
<i>Test characteristics</i>	Test to ensure compliance to requirements	Testing to failure

IV.

DISCUSSION OF ACADEMIC IMPLICATIONS, AND CONCLUSIONS

This paper proposes an approach to the design-build experience that is different from the most common one. It emphasises engineering science based on discovery and knowledge of the fundamentals, and a sound, fact-based development methodology in which advancement to the next stage *is only made when there are no longer any unclear circumstances*.

We know that practical experience of design and manufacturing in a workshop greatly motivates many students for their further engineering studies, so from this point of view a design-build experience is well worth the extra resources needed to carry it out. The early tests and simulations performed in order to learn and explore gives engineering students active learning experiences, and it should be expected that:

- The courses need to be adapted to the new process in that more time will be devoted to lab work
- The total volume of experimentation and testing in a course will increase
- The longer time spent in labs and workshops makes it likely that the students will need more supervision, which puts extra strain on teaching resources as well as infrastructure
- More tools and machines will be needed, perhaps also more lab space
- The consumption of materials for building purposes will go up
- Lego type or similar equipment for experimentation might be needed
- Safety courses for the students before they are allowed into the labs must come earlier and perhaps also run faster

The proposal to shift from DBT to TDB in engineering education is based on insights from research in Lean Product Development. By employing the learning focus of TDB, we return to the roots of engineering design and the activities central to the process of developing new products and systems.

However, the TDB philosophy puts even higher demands

on the educational institutions compared to the present situation. If we want to move from today's DBT projects to projects run according to the TDB sequence, it is important to try to determine what the implications are in terms of time needed, laboratory and workshop resources as well as teacher competence.

REFERENCES

- [1] J. Malmqvist, Young, P. W., Hallström, S., Kutteneuler, J., Svensson, T., "Lessons Learned from Design-Build Test-Based Project Courses," presented at the Design 2004 Dubrovnik, Croatia 2004.
- [2] D. Raudberget and P. Cannmo, "Learning product development through a design-build-test project – why is coaching important?" Proceedings of the 4th International CDIO Conference, Gent, Belgium, 2008.
- [3] N. F. M. Roozenburg and J. Eekels, *Product design: fundamentals and methods*, Chichester: Wiley 1995.
- [4] D. G. Ullman, *The mechanical design process*, 3. ed., New York: McGraw-Hill, 2002.
- [5] K. N. Otto and K. L. Wood, "Product Evolution: A Reverse Engineering and Redesign Methodology," *Research in Engineering Design*, vol. 10, pp. 226-243, 1998.
- [6] M. Kennedy, K. Harmon, and E. Minnock, *Ready, Set, Dominate: Implement Toyota's Set-Based Learning for Developing Products and Nobody Can Catch You*. Richmond, Va: Oaklea Press, 2008.
- [7] A. C. Ward, *Lean product and process development*. Cambridge, MA: Lean Enterprise Institute, 2007.
- [8] J. M. Morgan and J. K. Liker, *The Toyota product development system: integrating people, process, and technology*. New York, NY: Productivity Press, 2006.
- [9] C. P. B. R. L. Ash, K. W. Hyde "prop-Wrights - How two brothers from Dayton added a new twist to airplane propulsion.," *Mechanical Engineering* 2003.