



26th annual **INCOSE**
international symposium

Edinburgh, UK
July 18 - 21, 2016

What does it mean to be Lean in SoSE?

Alexey Tregubov, Jo Ann Lane

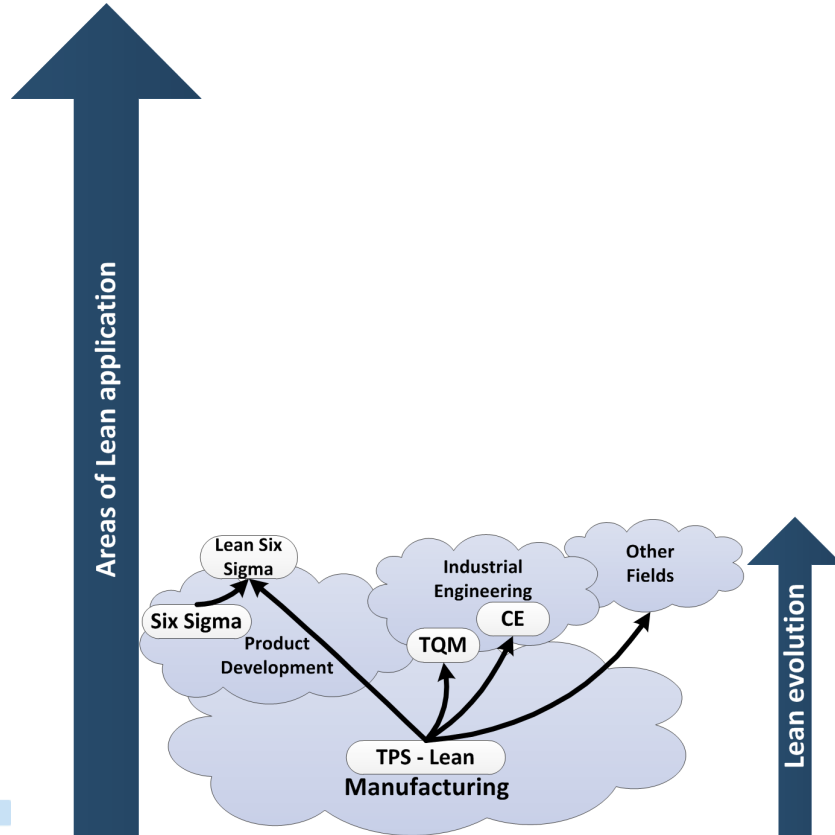


Outline

- Background – evolution of Lean
- Goals of the research
- Single system SE vs. SoSE
- Lean Principles in SoSE



Background



TPS - Lean

Toyota Production System

TQM

Total Quality Management

CE

Concurrent Engineering

Lean SW
Dev.

Lean Software Development

KSS

Kanban Scheduling System for SoS

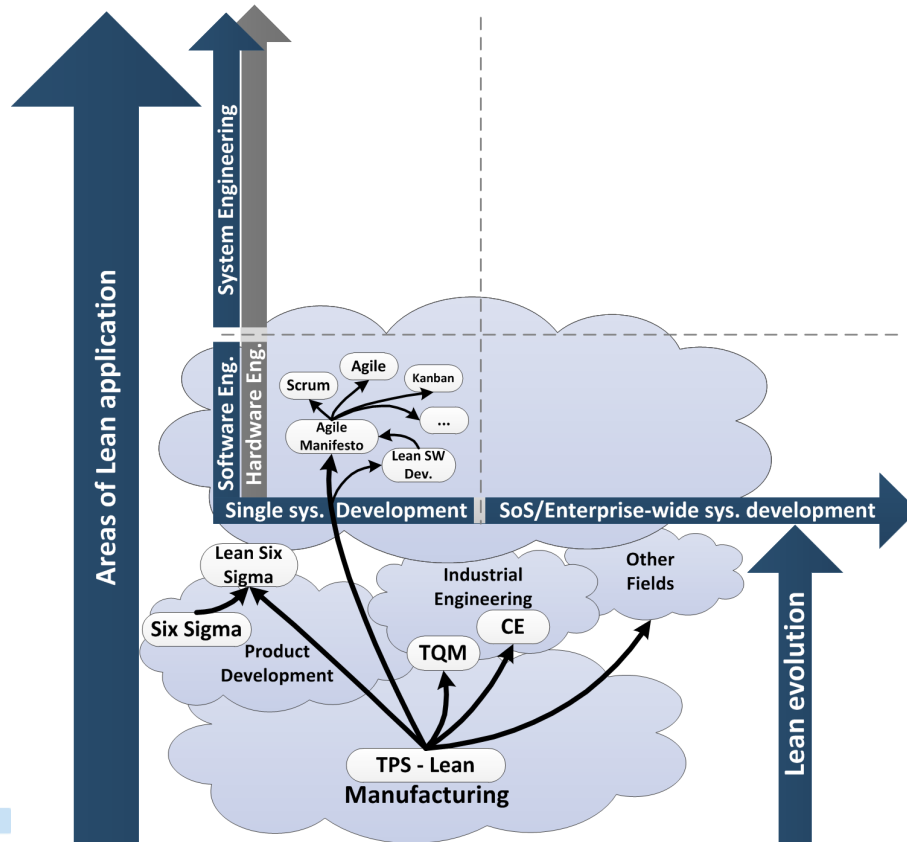
SEaaS

SE as a service

LEfSE

Lean for System Engineering

Background



TPS - Lean

Toyota Production System

TQM

Total Quality Management

CE

Concurrent Engineering

**Lean SW
Dev.**

Lean Software Development

KSS

Kanban Scheduling System for SoS

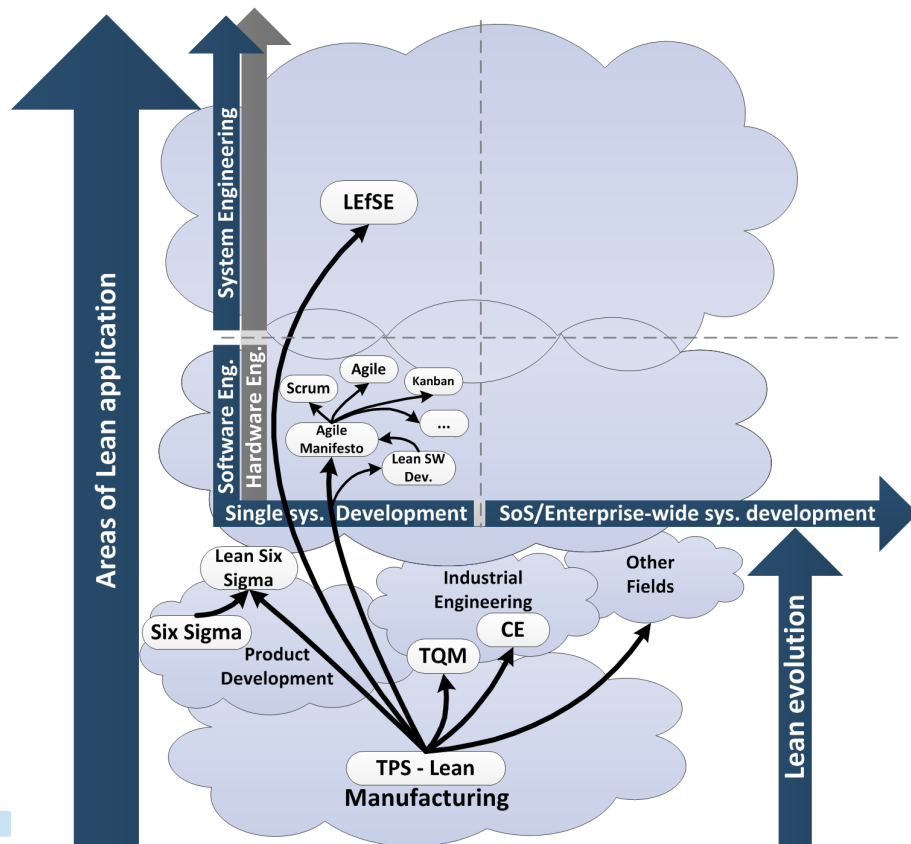
SEaaS

SE as a service

LEfSE

Lean for System Engineering

Background



TPS - Lean

Toyota Production System

TQM

Total Quality Management

CE

Concurrent Engineering

**Lean SW
Dev.**

Lean Software Development

KSS

Kanban Scheduling System for SoS

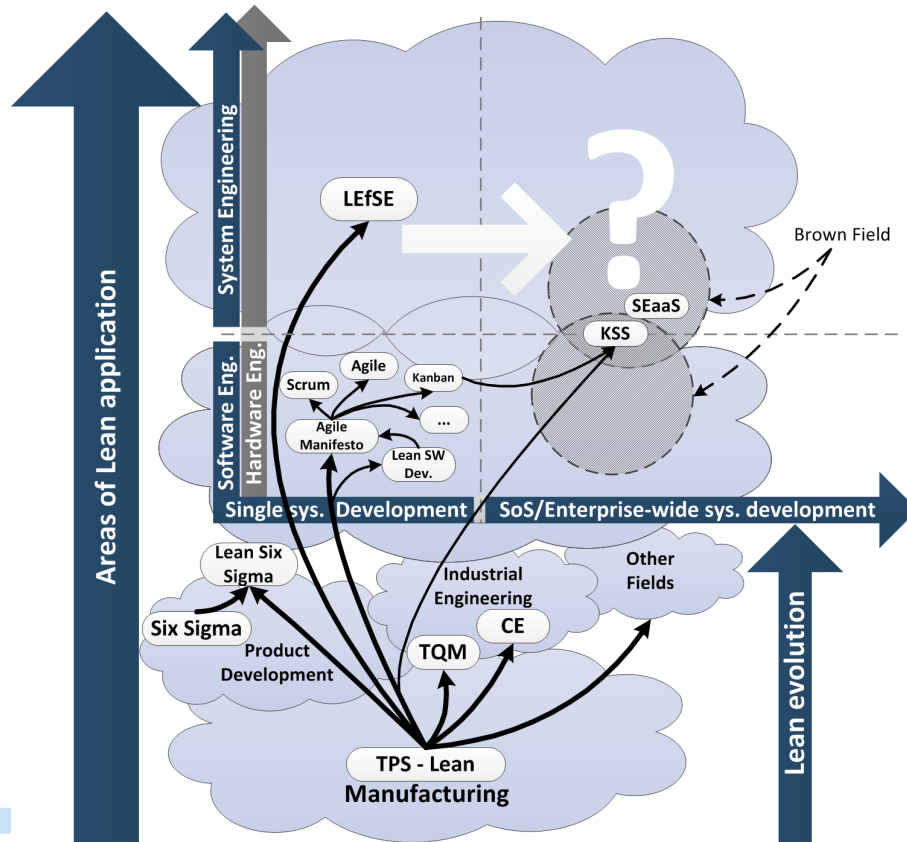
SEaaS

SE as a service

LefSE

Lean for System Engineering

Background



TPS - Lean

Toyota Production System

TQM

Total Quality Management

CE

Concurrent Engineering

Lean SW Dev.

Lean Software Development

KSS

Kanban Scheduling System for SoS

SEaaS

SE as a service

LEfSE

Lean for System Engineering

Goals

Identify key aspects and properties of system of systems engineering (SoSE) activities that support Lean adaptation in SoSE.



SoSE core elements*



26th annual **INCOSE**
international symposium

Edinburgh, UK
July 18 - 21, 2016

Ongoing oversight	Evolutionary improvements
<ul style="list-style-type: none">• Understanding systems and relationships• Monitoring and assessing constituent system changes	<ul style="list-style-type: none">• Translating capability objectives• Developing and evolving an SoS architecture• Addressing requirements and solution options• Orchestrating upgrades to SoS
<ul style="list-style-type: none">• Assessing performance to capability objectives	

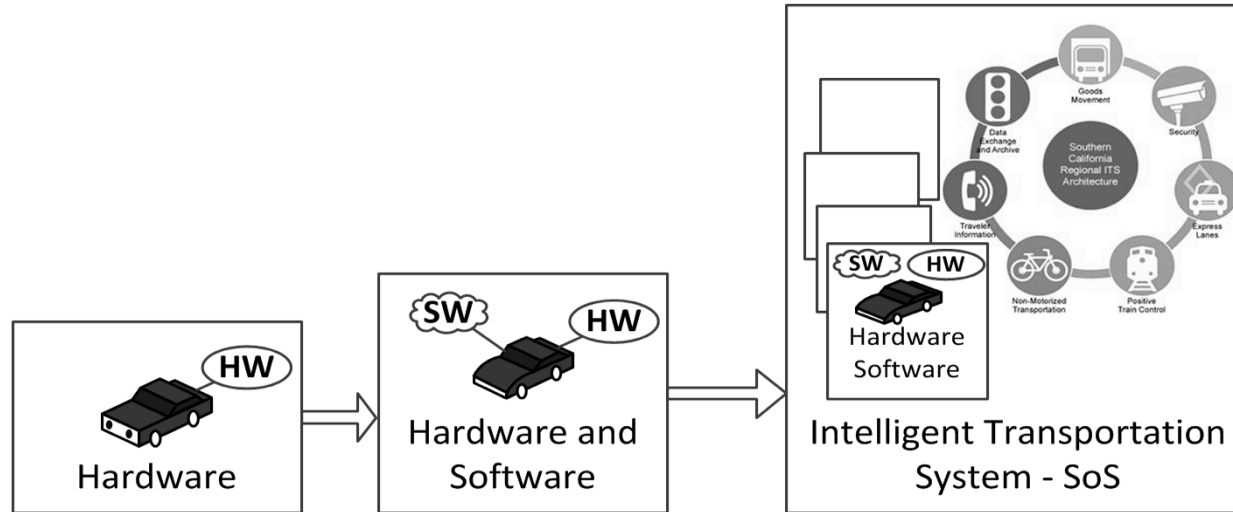
* US Department of Defense. 2008. *Systems Engineering Guide for Systems of Systems, Version 1.0*. (ODUSD(A&T)SSE, 2008)

Single system SE vs. SoSE



	SE in Single / Constituent System development	SE in System of Systems
Type	New system development and evolutionary system enhancement (green and brownfields).	Mostly evolutionary system enhancement (brownfield).
Life cycle	SE System Life Cycle Process Overview (ISO/IEC 15288) implementing a waterfall, incremental, or iterative approach.	Less constrained structure in timing or sequencing the SoSE core elements.
Org. structure	Organization-specific structures primarily based on single product development that may include subcontractors and vendors.	Typically, SoS structure has at least three levels: <ul style="list-style-type: none">• Executive/Stakeholder Management level.• SoSE level.• Product Development/Evolution level.
SE scope	SE activities focus on single product or system development.	The SoSE team's responsibility is to collaborate and negotiate stakeholders and developers of constituent systems to support SoSE capability implementation and evolution.

From SE to SoSE



Intelligent Transportation Systems

Lean Principles in SoSE*



- Identify Value
- Map the Value Stream
- Create Flow
- Establish Pull
- Seek Perfection

* LEfSE: Oppenheim, B. W. (2011). Lean for systems engineering with lean enablers for systems engineering (Vol. 82). John Wiley & Sons.

Identify Value



LEfSE:

- Establish the Value of the end product or system to the customer
- Frequently involve the customer

SoSE:

- New SoS capabilities requests mapped and translated to changes in constituent systems
- SoS stakeholders priorities are integrated with single system changes and priorities

Map the Value Stream



LEfSE:

- Map the systems engineering and product development value streams and eliminate non-value added elements
- Plan for frontloading the program
- Plan to develop only what needs developing
- Plan to prevent potential conflicts with suppliers development
- Plan leading indicators and metrics to manage the program

SoSE:

- The SoSE team continuously monitors and adjusts plans and priorities
- No “front-loading” is required, it is a continuous process
- “nice to have” changes are tracked as a low priority and wait for opportunities to include them with high value changes
- Obsolete changes dropped
- Track the SoSE performance and how it meets capability objectives

Create Flow



LEfSE:

- Clarify, derive, and prioritize requirements early and often during execution
- Front load architectural design and implementation
- Systems engineers to accept responsibility for coordination of product development activities
- Use efficient and effective communication and coordination
- Promote smooth SE flow
- Make program progress visible to all
- Use lean tools

SoSE:

- SoS capabilities drive SoS evolution and its architecture, the SoS architecture is rarely developed from scratch
- SoSE team works to understand the current “as is” state of the architecture and then works to improve the existing architecture and associated performance
- Critical to success: collaboration with the constituent system engineers, understanding current development activities of constituent systems
- The SoSE team and system engineering teams of constituent systems may use Lean tools (e.g. multi-level Kanban boards).

Establish Pull



LEfSE:

- Pull tasks and outputs based on need, and reject others as waste

SoSE:

- The flow of SoSE work to constituent systems for development is typically based on a value-based priority scheme
- There are ways to use pull scheduling in work queues, but at the SoS level work often require specialized engineering skills, which limits options who can pull work from the queue.
- Pull scheduling can be used within the constituent systems.

Seek Perfection



LEfSE:

- Strive for excellence of SE processes
- Use lessons learned from past programs for future programs
- Develop perfect communication, coordination, and collaboration policy across people and processes
- For every program, use a chief engineer role to lead and integrate the program from start to finish
- Drive out waste through design standardization, process standardization, and skill-set standardization
- Promote all three complementary continuous improvement methods to draw best energy and creativity from all employees

SoSE:

- Most SoSE teams have limited resources and committed to optimization of their processes.
- Most SoSE teams have limited authority over the constituent systems; therefore, they have limited abilities to seek perfection in processes used by constituent systems.
- The SoSE team seeks perfection on the SoS level by evolving architecture, improving performance and collaboration with constituent systems.

Summary

- Given these considerations, we explored how Lean principles can be applied to SoSE.
- The SE Lean Enablers provide a framework for Lean adaptation in SE.
- We showed how these enablers fit into SoS environments, which can facilitate and improve Lean adaptation in SoSE.

Q&A

- Do you agree with the point above?
- Is there anything to add?



Contacts



Authors:

- *Alexey Tregubov*
 - *University of Southern California*
 - *Los Angeles, CA, USA*
 - **tregubov@usc.edu**
 - **atregubov.com**

- *Jo Ann Lane*
 - *University of Southern California*
 - *Los Angeles, CA, USA*
 - **jolane@usc.edu**

References



- Oppenheim, B. W. (2011). Lean for systems engineering with lean enablers for systems engineering (Vol. 82). John Wiley & Sons.
- Poppendieck, M., & Poppendieck, T. (2003). Lean software development: an agile toolkit. Addison-Wesley Professional.
- Turner, R., & Lane, J. A. (2013). Goal-Question-Kanban: applying lean concepts to coordinate multi-level systems engineering in large enterprises. *Procedia Computer Science*, 16, 512-521.
- Tregubov, A., & Lane, J. A. (2015). Simulation of Kanban-based Scheduling for Systems of Systems: Initial Results. *Procedia Computer Science*, 44, 224-233.
- US Department of Defense. 2008. *Systems Engineering Guide for Systems of Systems, Version 1.0*. Washington, DC (US): ODUSD(A&T)SSE: Office of the Deputy Under Secretary of Defense for Acquisition and Technology, Systems and Software Engineering.
- Boehm, B., Lane, J. A., Koolmanojwong, S., and Turner, R. (2014); *The Incremental Commitment Spiral Model: Principles and Practices for Successful Systems and Software*, Addison-Wesley, ISBN-13: 978-0-80822-6.