

Semestrální práce ke kurzu 4IT421 Zlepšování procesů budování IS	
Semestr	letní 2017/2018
Autoři	Žárský Jiří, Bc. (xzarj08) Sobíšek Ondřej, Bc. (sobo01) Bayer Jiří, Bc. (bayj00)
Téma	Sustainable Cultural Agile Release in the Enterprise (SCARE)
Datum odevzdání	2.5.2018

## Abstrakt

Existují již poměrně zavedené agilní metodiky, které je ovšem možné dále vylepšovat, nebo měnit. Téma SCARE je vlastně další z metod škálování agilního přístupu. Cílem práce je objasnit kdo a kdy s myšlenkou agilního škálování SCARE přišel. Dále také srovnat s ostatními agilními škálovacími metodami.

## Klíčová slova

SCARE, agilní metodika, Teorie omezení, agilní škálování, řízení projektů

## Obsah

1	Úvod.....	2
2	Charakteristika metody SCARE .....	3
3	Teorie omezení.....	4
4	Aplikace teorie omezení na agilní metodiky .....	5
5	Srovnání metod agilního škálování .....	7
6	Hodnocení použití metody SCARE .....	8
7	Závěr .....	9
	Seznam literatury.....	10

---

# 1 Úvod

SCARE (Sustainable Cultural Agile Release in the Enterprise) je jedna z metod škálování agilního přístupu napříč organizací. Zmínku o této nové metodě prezentoval Matthew Heusser v pěti minutové bleskové přednášce na konferenci Agile2014.

První agilní přístupy, které byly vyvinuty v souladu s agilním manifestem z roku 2001, se zaměřovaly především na malé týmy a softwarový vývoj (např. SCRUM, DSDM nebo XP). Tyto metody umožnily malým týmům pracovat efektivněji a dodávat hodnotu zákazníkovi rychleji. V dnešní době se celé organizace snaží posunout k agilnímu fungování a aplikovat “agile” na větší týmy a koordinaci mezi těmito týmy. S tím přicházejí nové výzvy - provádět velké změny ve fungování celého IT oddělení je rizikovější, než v malém týmu o pěti lidech.

Existuje několik metod škálování agilních metod. Nejpopulárnější metodou je SAFe, která adresuje koordinaci mezi více agilními týmy. Dalším přístupem je LeSS, který na škálování aplikuje SCRUM. Metodika DaD je další alternativou, jež se snaží zkombinovat více agilních přístupů (SCRUM, agilní modelování, “lean” softwarový vývoj, ...). SCARE je spíše okrajová, nepříliš popsána metoda využívající teorii omezení, která našla již uplatnění v řízení projektů, výrobě nebo logistice.

## **Cíle práce a přístupy k jejich dosažení**

Hlavní cílem práce je přiblížit co vlastně SCARE metoda znamená. Kdo s touto novou myšlenkou přišel a jak dlouho je metoda vymyšlena.

Dalším cílem práce je posoudit, jak by SCARE mohl být prakticky použit pro škálování agilních přístupů. Prvním krokem bude shrnutí teorie omezení a popsání možností její aplikace na různé agilní metodiky. V dalším kroku bude SCARE porovnán s ostatními způsoby škálování (SAFe, LeSS a DaD). Ke konci práce bude hodnoceno použití metody SCARE.

---

## 2 Charakteristika metody SCARE

Metoda SCARE začíná prvním krokem, ve kterém se pokusíte vybudovat jeden vysoce výkonný multidisciplinární tým lidí. V týmu nebude využito multiprojektové řízení a těžké procesy, nebo těžké svazující metodiky. Tento tým bude schopen kompletně vykonávat svou práci. (Techtarget, 2014)

Tým začíná svou práci na nejkritičtějším projektu. Což může být v současnou chvíli třeba nový produkt, nová verze produktu atd. V rámci projektu nalezneme kritické místo, které omezuje tok. Zjistíme, co omezuje tok a jak je možné toto kritické místo řídit. Přizpůsobujeme okolí tohoto omezení, aby mohlo podat co nejvyšší výkon. Omezení následně upravíme do fáze, kdy zvýší výkonnost celého projektu. Stejně jako v teorii omezení nesmíme toto kritické místo jen přesunout jinam, protože by hrozilo opětovné hledání a odstraňování nového omezení. (Techtarget, 2014)

Každý den, kdy je tým ve zpoždění při dodávce projektu, hraje velkou roli a zároveň tím zpomalí a oddálí uskutečnění dalšího projektu. To vytváří řetězový efekt. Zpoždění osob na projektu se násobí a zvyšuje se úsilí jednotlivých pracovníků. (Techtarget, 2014)

Většinou se dobře ví, kde jsou na projektu omezení a není těžké je najít. V klasickém projektovém řízení jednotliví pracovníci z různých profesí jako jsou vývojáři, analytici a projektový manažeři bývají různě umístěni po budově či po jednotlivých patrech budovy a vzájemná komunikace probíhá složitěji. Mají na sebe telefonní spojení a emaily, ale přímá komunikace není častá. Tyto lidi musíte umístit do jedné místnosti, ve které pomocí moderních metod vytvoříte příjemné a důstojné pracovní prostředí. Tak zvýšíte výkonnost projektů až o tisíce procent. Pracovníci se nepřidělují k jiným projektům. Pracuje se zde od začátku do konce na jednom projektu v rámci týmu. Po dokončení jednoho projektu tým zůstane připraven přijmout další projekt. (Techtarget, 2014)

Metoda SCARE nemusí být vhodná pro všechny činnosti ve firmě. Například pro údržbu, nebo pro již probíhající projekty. Tyto projekty mohou být dále řízeny tradičním způsobem řízení.

Není vždy lehké přecházet na agilní metodiky, protože síla zvyku dlouholetých zaměstnanců, kteří pracují v rámci tradičního způsobu řízení projektů je vysoká.

---

## 3 Teorie omezení

Jak již bylo zmíněno v úvodu, metodika SCARE využívá teorii omezení. Jedná se o metodologii, kdy se snažíme v systému nalézt takový faktor, který nejvíce brání v dosažení jeho cíle. Poté, co tento faktor identifikujeme, se neustále snažíme o jeho zlepšení až do té doby, dokud žádný takový faktor neexistuje. V takovém případě by měl systém začít dokonale fungovat. (Goldratt, 1998)

S touto teorií přišel v roce 1984 Dr. Eliyahu Goldratt a uvedl ji ve své knize „The Goal“, která se okamžitě stala bestsellerem. Důležitou charakteristikou této teorie je, že kontinuálně prioritizuje příležitosti ke zlepšování jednotlivých aktivit. Aktivita s nejvyšší prioritou je právě to omezení, které je nutné eliminovat, aby systém mohl fungovat lépe a efektivněji. (Hamant, 2009)

Pokud chceme pracovat se systémem podle teorie omezení, znamená to řídit se těmito následujícími kroky: (Goldratt, 1998)

1. Nalezení omezení, které brání efektivnějšímu fungování systému
2. Zjištění, co omezení ovlivňuje a jak je možné ho řídit
3. Přizpůsobení okolí omezení tak, aby mohlo podávat pokud možno co největší výkon
4. Pokud omezení stále přetrvává je nutné alokovat další zdroje k jeho odstranění
5. Opakování celého cyklu na další omezení, které v systému nalezneme.

Těchto pět kroků se v cyklu opakuje stále dokola a postupně jednotlivá omezení odstraňujeme. Ideálním výsledkem je systém bez existujících omezení, což ovšem v praxi není nikdy možné. Pokud by totiž takový systém existoval, dosahoval by nekonečného výkonu, jelikož by nic nebránilo v jeho růstu. (Heusser, 2015)

V souvislosti s teorií omezení je nutné si položit otázku, co to vlastně omezení je. Zjednodušeně řečeno, omezení může cokoliv, co brání organizaci v dosahování jejích cílů. Omezení můžeme obecně rozdělit do 4 kategorií: (Heusser, 2015)

- Fyzické: Typicky vybavení, ale obecně i jiné hmotné věci, jako nedostatek materiálů, prostoru, či pracovníků
- Pravidla a nařízení: Například neformální zásady práce v dané firmě či legislativní nařízení
- Paradigma: Vychází z hluboce zakořeněných zvyklostí v dané organizaci.
- Tržní: Příkladem může být, pokud produkce přesahuje prodej.

Teorie omezení má také velký význam v souvislosti s agilními metodikami. Tento vztah je rozebrán v další kapitole.

---

## 4 Aplikace teorie omezení na agilní metodiky

Teorii omezení je možné aplikovat na mnoho odvětví, jako jsou výroba, finance, marketing, nebo projektové řízení. V této kapitole rozebereme její aplikaci na agilní metodiky se opět budeme řídit pěti kroky, které jsou uvedeny v předchozí kapitole.

Prvním krokem při aplikaci teorie omezení je nalezení samotného omezení. Tento krok se zdá být přímočarý, protože v metodikách Lean nebo Agile je celkem jednoduché identifikovat procesy, které jdou „proti“ principům těchto metodik. Nicméně, zlepšení každého identifikovaného problému nemusí nutně znamenat zlepšení celého systému. Zastánci agilních metodik se totiž někdy až příliš soustředí na dodávku systému, ale nevěnují se business modelu dostatečně. Pokud si uvědomíme, že naším největším problémem je například nedostatečná kapacita vývojového týmu, můžeme do projektu angažovat další lidi, kteří s projektem pomohou. Potom ale najdeme další problém, kterým může být klesající prodej a ačkoli jsme první problém s nedostatkem lidí vyřešili, musíme se hned věnovat dalším identifikovaným problémům a tento cyklus by se mohl opakovat do nekonečna. (Fuqua, 2013)

Druhým krokem je zjištění, co omezení ovlivňuje a jak je možné ho řídit. Jak už bylo zmíněno v předchozím kroku, nalezení samotného omezení nestačí. Je důležité najít souvislosti mezi jednotlivými omezeními, abychom je mohli efektivně řídit a věnovat se tomu, které má v danou chvíli největší prioritu. Není nejefektivnější se věnovat mnoha omezením najednou, ale soustředit se raději na několik hlavních a po vyřešení těch se zaměřit na další. Řekněme, že naše největší omezení je nedostatečná kapacita vývojového týmu. Po tom, co toto omezení odstraníme, vzniknout logicky další omezení, jako merging větví do produkce či delší čas na provádění code review. Toto jsou přesně ta omezení, kterým bychom se měli věnovat dále, jelikož spolu všechna souvisí a dohromady tvoří naše identifikované úzké hrdlo ve funkčnosti celého systému. (Fuqua, 2013)

Třetím krokem je přizpůsobení okolí omezení. V praxi to znamená, že bychom měli téměř vše podřídit tomu, abychom omezení co nejdříve odstranili. Příkladem může být omezení počtu schůzek s managementem či refaktORIZACE kódu místo přidávání nových funkcionalit do starého špatně navrženého kódu. (Fuqua, 2013)

Čtvrtý krokem je alokace dalších zdrojů, pokud omezení stále přetrvává. V tomto případě může být řešení několik. Buď bychom mohli angažovat ještě další lidi do projektu, ale to

---

nemusí nutně pomoci. Jiné a lepší řešení by mohlo zaplatit lidem dodatečná školení, aby se vývoj celého systému zrychlil, nebo najmout specializované odborníky na problematiku, ve které se domínváme, že náš tým zaostává. Další možností by bylo analyzovat používané nástroje a případně investovat do modernějších. (Fuqua, 2013)

Poslední krok se týká zpětného vyhodnocení toho, jestli se nám omezení podařilo odstranit. Pokud ano, věnujeme se dalším identifikovaným omezením, které s původním omezením nemusí souviset. A pokud ne, hledáme další způsoby, jak dané omezení odstranit. (Fuqua, 2013)

---

## 5 Srovnání metod agilního škálování

Pravděpodobně nejpoblárnější metodu škálování agilního přístupu je metoda SaFE, jejíž autor je Dean Leffinwell. Díky její popularitě je k dispozici mnoho tutoriálů a videí online. Existují i možnosti certifikace. SaFE dává organizacím přesný návod, jak dosáhnout cílů a aplikovat agilní myšlení napříč celou svojí strukturou. Základní myšlenkou SaFE je tzv. „vlak nasazení“ („release train“), což je tým složený z dalších týmů. Typicky se jedná o 50 až 125 lidí. Tento velký tým podporuje celý program (sestavu projektů) a zahrnuje různé role od analytiků přes vývojáře až po testery. Mezi hlavní metriky SaFE patří zákaznická spokojenost, time-to-market nebo kvalita software. (Heusser, 2015)

Další metoda Large Scale Scrum (LeSS) používá ke škálování známou agilní metodu SCRUM a aplikuje ji na větší celek týmu složených z týmů. Každý den probíhá „standup meeting“, kterého se účastní jeden až dva členové z každého týmu. Retrospektiva se odehrává v týdnu po skončení sprintu. Zahrnuje více lidí z každého týmu a diskutují se na ní problémy s celým programem. LeSS přidává i další prvky koordinace mezi týmy jako „open space“ nebo „town hall meetings“. Pro velké korporace také existuje rozšíření známé jako LeSS Huge. (Heusser, 2015)

Metodika Disciplined Agile Delivery (DAD) sice staví na SCRUMu, ale snaží se zahrnout další oblasti, které SCRUM záměrně neadresuje. DAD nabízí komplexnější přístup k agilní dodávce. Podporuje robustní sadu rolí, kombinuje různé agilní elementy (agilní modelování, extrémní programování, unified process, kanban a lean software development), je otevřený (zdarma), podporuje různé typy životních cyklů dodávky, zahrnuje veškeré aspekty dodávky (programování, dokumentace i governance) a nabízí spíše možnosti, než aby předepisoval postupy. (Heusser, 2015)

LeadingAgile je společnost, jenž nabízí další přístup k agilní transformaci organizací. Místo škálovacího rámce LeadingAgile spíše nabízí rámec transformace společnosti. Tato transformace začíná posouzením podnikových cílů s ohledem na jejich predikovatelnost a přizpůsobivost. Metodika zjišťuje, zda se produkt má spíše „zjevit“ („emerge“) na základě potřeb trhu nebo „konvergovat“ („converge“) na základě dodávání specifických požadavků a features v předem definovaných intervalech. (Heusser, 2015)

Metoda SCARE je alternativou k výše zmíněným hlavním metodám škálování agilních přístupů. Její zhodnocení následuje v další kapitole.

---

## 6 Hodnocení použití metody SCARE

SCARE na rozdíl od ostatních metod škálování nenabízí hotové koncepty a postupy. SCARE není nijak oficiálně zdokumentovaný. Neexistuje žádný systém certifikací, ani společnost, která by ho zaštiťovala. Jedná se pouze o nápad formulovaný Mattem Heusserem. Tento nápad poprvé prezentoval na konferenci Agile2014, kde se dle jeho slov setkal s úspěchem. Informace o SCARE lze dohledat jen v jediném článku na Internetu. (Techtarget, 2014) S robustními přístupy popsanými výše ho lze jen těžko srovnávat.

SCARE je zkratka pro „Sustainable, Cultural Agile Release for Enterprise“. Autor tento název zvolil tak, aby působil vtipně a kontrastně k metodice SaFE. Toho údajně lituje, protože se nemá jednat o vtip. (Techtarget, 2014) Na rozdíl od SaFE se SCARE nesnaží organizacím poskytnout holistický návod, jak se agilně transformovat. Zaměřuje se spíše na nalezení kritického místa (týmu) a tak přispívá k agilní transformaci celého celku. SCARE doplňuje pouze malý střípek do celé skládačky agilního škálování.

Celá myšlenka se opírá o budování klíčových týmů v rámci organizace. Tyto týmy obsahují všechny možné role, ačkoliv autor přesně nespecifikuje jaké a ani není dán počet lidí v těchto týmech. Členové klíčového týmu by měli pracovat pouze na jednom projektu zároveň. Až je projekt dokončen, celý tým se přesouvá k práci na jiném projektu. Lidé z toho týmu nesmí být ad-hoc přiřazováni na jiné projekty. (Techtarget, 2014)

Jakmile je první klíčový tým funkční a dodává výsledky, najdou se v organizaci pravděpodobně další lidé, kteří se budou chtít přidat. (Techtarget, 2014) Takto může vzniknout další „vysoce funkční“ tým a dochází ke škálování. Stará organizační struktura zůstává na místě, akorát je „zespodu“ agilně transformována. Výhodou tohoto přístup je jeho dobrovolnost a přirozenost. Nařízení shora vždy vytváří v lidech odpor ke změně. Tomu se SCARE snaží předejít.

Na druhou stranu SCARE nijak neřeší důležité otázky jako komunikaci mezi týmy, metriky, řízení portfolia projektů, životní cyklus dodávky nebo podnikové cíle. V praxi by SCARE mohl být využit jenom jako komplement jiné škálovací metody. Je představitelné, že by se SCARE aplikoval zároveň s metodou LeSS. Agilní transformace by nezahrnovala všechny týmy od začátku, ale postupně by se rozšiřovala zespodu v iteracích (sprintech) napříč celou organizací.

---



## 7 Závěr

Tato práce měla za cíl seznámit čtenáře s metodou SCARE a posoudit, jak by mohla být tato metoda prakticky použita pro škálování agilních metodik. V úvodní části práce a první kapitole byla tato metoda charakterizována a bylo ukázáno, že tato metoda vychází z teorie omezení. Teorii omezení se proto věnuje následující kapitola. Teorii omezení je v praxi možné aplikovat na agilní metodiky, což bylo předvedeno v kapitole 4. Další kapitola srovnávala metodu SCARE s dalšími metodikami, jako jsou LeSS či DAD. V poslední části byla metoda SCARE zhodnocena a bylo ukázáno, kde by se dala prakticky použít.

Metoda SCARE zní zajímavě jako další doplněk k metodikám agilního škálování. Bohužel k ní však neexistuje dostatek informačních zdrojů, jelikož byla prezentována na jediné konferenci během 5 minut v roce 2014 a na internetu k ní existuje pouze jediný článek. Pokud by bylo informačních zdrojů více, bylo by jistě zajímavé se metodě do budoucna věnovat dále a podrobněji.

---

# Seznam literatury

1. GOLDRATT, Eliyahu M, 1998. Essays on the theory of constraints. Great Barrington, MA: North River Press, 1998. ISBN 0884271595.
  2. Techtarget, 2014. Introducing the SCARE Method [online]. 2014 [cit. 2018-03-07]. Dostupné z: <http://itknowledgeexchange.techtarget.com/uncharted-waters/introducing-the-scare-method/>
  3. Forbes, 2016. What Does It Mean To Scale Agile? [online]. 2016 [cit 2018-03-05]. Dostupné z: <https://www.forbes.com/sites/stevedenning/2016/04/15/what-does-it-mean-to-scale-agile/>
  4. HAMMANT, Paul, 2009. Theory of Constraints and Agile [online]. 2009 [cit. 2018-03-05]. Dostupné z: <https://paulhammant.com/blog/theory-of-constraints-andagile.html>
  5. Leanproduction, 2016. Theory of constraints [online]. 2016 [cit. 2018-04-16]. Dostupné z: <https://www.leanproduction.com/theory-of-constraints.html>
  6. HEUSSER, Matt, 2015. Comparing scaling agile frameworks [online]. 2015 [cit. 2018-04-17]. Dostupné z: <https://www.cio.com/article/2974436/agile-development/comparing-scaling-agile-frameworks.html>
  7. FUQUA, Andrew, 2013. The Theory of Constraints In Agile [online]. [cit. 2018-05-02]. Dostupné z: <https://dzone.com/articles/the-theory-of-constraints-in-agile>
-