

<b>Semestrální práce ke kurzu 4IT421 Zlepšování procesů budování IS</b>	
<b>Semestr</b>	ZS 2017/2018
<b>Autoři</b>	Lukáš Lazar (xlazl00) Martin Kapal (xkapm25) Martin Zákřavský (zakm05) Michal Krokosch (xkrom31)
<b>Téma</b>	Doing Scrum with Multiple Teams: Comparing Scaling Frameworks
<b>Datum odevzdání</b>	17.12.2017

## **Abstrakt**

Tato práce se zabývá problematikou použití agilní metodiky Scrum ve více týmech. Řešením této problematiky jsou škálovací frameworky, které jsou zde popsány a následně porovnány.

## **Klíčová slova**

Scrum, škálování, agilní, LeSS, SAFe, Scrum@Scale.

# Obsah

<b>Úvod</b>	<b>2</b>
<b>1 Scrum</b>	<b>3</b>
1.2 Artefakty	3
1.2 Klíčové role v týmu	4
1.3 Workflow	4
<b>2 Škálování Scrumu</b>	<b>5</b>
<b>3 Škálovací frameworky</b>	<b>6</b>
3.1 LeSS	6
LeSS (do 8 týmů)	6
Less Huge	7
3.2 SAFe	8
Základní údaje	8
Struktura pohledu a týmů	8
Aplikace a odezva	9
3.3 Scrum@Scale	10
<b>4 Srovnání frameworků</b>	<b>11</b>
<b>Závěr</b>	<b>12</b>
<b>Literatura</b>	<b>13</b>

# Úvod

Agilní metodiky vývoje softwaru se stávají čím dál víc oblíbenými v nejrůznějších projektech. Mezi ně patří i Scrum, který se osvědčil především v malých vývojových týmech. Na větší týmy nebo velké množství týmů už však samotný Scrum nestačí, a proto je třeba hledat pomoc v podobě škálovacích frameworků.

Cílem této semestrální práce je oslovit čtenáře s předpokládanými předchozími znalosti ve světě agilního vývoje, a to ve formě rešerše tří nejčastějších škálovacích frameworků vycházejících z agilní metodiky Scrum. Výstupem je srovnání popsaných frameworků a poukázání na jejich kvality a nedostatky.

# 1 Scrum

Metodika Scrum je jednou z nejčastěji využívaných metodik agilního vývoje. Jedná se o velmi “lehkou” metodiku, která je jednoduchá na pochopení, ale je obtížné ji dokonale zvládnout (Schwaber a Sutherland 2016, str. 3). Její síla spočívá v iterativním vývoji, úzkou spoluprací s vlastníkem vyvíjeného produktu a komunikací mezi všemi členy týmu.

Vývoj po krátkých iteracích je klíčovou vlastností agilního přístupu. Oproti rigorózním přístupům, kde je životní cyklus projektu striktně rozdělen na několik fází (analýza, návrh, implementace, testování), agilní metodiky tyto fáze zahrnuje do každé z iterací, což má za následek zlepšení kvality, neboť je do procesu zapojen vlastník produktu, který může kdykoliv mezi každou iterací měnit své požadavky, a kontrolovat tak výslednou podobu dodaného produktu (Rasmusson 2016).

Scrum definuje několik vlastních pojmů, které jsou popsány v následujících podkapitolách.

## 1.2 Artefakty

Scrum pracuje s těmito artefakty (Schwaber a Sutherland 2016, str. 13–15):

- **User Story** je požadavek, který by měl přinést přidanou hodnotu celkového produktu, a to neohledně na pořadí, ve kterém byl implementován. Může se jednat o požadavek na funkcionalitu produktu, ale i opravu chyby nebo jiných nefunkčních požadavků.
- **Product Backlog** je seznam všech požadavků na produkt, který má na starost Product Owner a z něhož čerpají členové vývojového týmu. Je řazen dle priority stanovené Product Ownerem.
- **Sprint Backlog** je výběr požadavků z Product Backlogu, které by měly být implementovány do konce příštího Sprintu.

## 1.2 Klíčové role v týmu

Týmy ve Scrumu se skládají z těchto rolí (Schwaber a Sutherland 2016, str. 5–7):

- **Product Owner** je osoba zodpovědná za maximalizaci hodnoty produktu a správu Product Backlogu.
- **Scrum Master** je osoba zodpovědná za správné pochopení a implementace metodiky Scrum v celém projektu. Slouží také jako poradce Product Ownera a ostatních členů týmu.
- **Development Team** je samostatný tým vývojářů, kteří mají za úkol dostat inkrementaci produktu do požadovaného stavu “Done” tak, jak jej určil Product Owner. Neexistuje vnitřní hierarchie nebo rozdělení rolí vývojářů – celý tým je zodpovědný za dodání přidané hodnoty jako celku.

## 1.3 Workflow

Scrum definuje vlastní workflow, které se skládá z pravidelných schůzek (Schwaber a Sutherland 2016, str. 7–9):

- **Sprint** je základní jednotka vývoje ve Scrumu. Jedná se o vymezený časový úsek, během kterého Scrum Team pracuje na požadavcích definovaných ve Sprint Backlogu a snaží se je dostat do takového stavu, aby mohl být dodán. Délka sprintu se obvykle stanovuje na 14 dní, ale může trvat až jeden měsíc.
- **Sprint Planning** je schůzka všech členů týmu před sprintem, během které se určují, které User Stories se mají dokončit v následující iteraci. Doba trvání Sprint Planningu by měla být maximálně osm hodin, pokud se jedná o jednoměsíční sprint. Pro kratší sprinty je plánování úměrně kratší.
- **Daily Scrum** je 15minutová schůzka vývojového týmu za účelem synchronizace aktivit a vytvoření plánu na další den. Zároveň slouží jako prostor pro řešení vzniklých překážek, které by bránily dalšímu vývoji a znamenaly by nedodání všech požadavků.
- **Sprint Review** představuje předvedení hotových požadavků ze Sprint Backlogu a jejich zhodnocení Product Ownerem. Obvykle trvá čtyři hodiny, jedná-li se o review měsíčního sprintu. Zároveň se během Sprint Review určí hrubé složení požadavků pro další sprint.
- **Sprint Retrospective** slouží jako příležitost týmu zhodnotit dosavadní práci na sprintu a případné vytvoření plánu pro zlepšení v dalším sprintu. Probíhá hned po Sprint Review, ještě před dalším Sprint Planningem. Probírají se zde nejen technické aspekty, ale i mezilidské vztahy mezi členy týmu.

## 2 Škálování Scrumu

Přestože Scrum obsahuje mnoho předností a výhod, které mohou vývojové týmy využít, nejedná se o metodiku univerzální. Scrum nelze jednoduše aplikovat na jakýkoliv tým a očekávat bezproblémový chod.

Jako hlavní nevýhoda Scrumu by se dal považovat fakt, že je primárně určen pro použití v malých týmech až do 10 lidí. V takto malém počtu lidí může stále probíhat úzká komunikace mezi jednotlivými členy týmu a Product Owner má ucelený přehled o veškeré probíhající práci na projektu.

Problémy nastávají, kdy se tým rozroste do takové míry, že není dále možné udržet konzistenci vývoje a spolehlivost dodání produktu. Buď se jedná o jeden velký tým s řádově stovkami lidí, nebo jsou vývojáři rozděleni do několika menších týmů. Zpravidla se jedná o vývoj složitějších produktů, u kterých existuje velké množství závislostí mezi členy týmu nebo mezi týmy, a tak je obtížné koordinovat dodání výsledného produktu.

Samotný Scrum na tyto problémy nestačí, a tak je potřeba tuto metodiku tzv. škálovat, tedy přizpůsobit většímu počtu lidí komplexity produktu. K tomu slouží škálovací frameworky, jež jsou popsány v následující kapitole.

# 3 Škálovací frameworky

## 3.1 LeSS

LeSS neboli Large Scale SCRUM je škálovací framework pro řízení agilního vývoje, vycházející z metodiky SCRUM (“škálovaný SCRUM”). LeSS se snaží být minimalistický, s co nejmenším počtem předem definovaných procesů. Je permissivní v tom, že se jedná o “lehký rámec” (v podstatě je to jen sada pravidel), který si podniky/týmy mohou přizpůsobit a pozměnit. Tvůrci frameworku jsou Craig Larman a Bas Vodde, kteří se na něm podílejí od roku 2005.

LeSS se snaží najít způsob jak aplikovat principy, prvky a eleganci SCRUMu ve velkém. Počítá s vícero týmy, které pracují spolu na jednom produktu. LeSS se dělí na dva frameworky, první je do osmi týmů a nazývá se LeSS, druhý je LeSS Huge a počítá se stovkami vývojářů.

### LeSS (do 8 týmů)

Jeden Product Owner, jeden Product Backlog, každý tým pak svůj vlastní Sprint Backlog. LeSS běží stejně jako SCRUM na principu sprintů, ty jsou synchronizované napříč týmy a na konci každého sprintu dohromady týmy “odevzdávají” potenciálně dodavatelný produkt. Synchronizované jsou tím pádem i plánování, konce sprintů, sprint review a retrospektivy.

Na začátku projektu je potřeba definovat co je produktem, to pomůže v rozhodnutí kolik týmů bude potřeba, jestli se má použít LeSS nebo LeSS Huge, kdo bude Product Ownerem a o co vše se bude starat (oproti SCRUMu zde PO zastává spíše pozorovatelskou roli a orientuje se na výnos investice projektu).

**Plánování sprintů** se odehrává ve dvou úrovních:

- 1) Co bude výsledkem celého sprintu?
  - a) Co a jak?
  - b) Výběr “kartiček” pro sprint backlog (za všechny týmy)
  - c) Zástupci týmů, PO
- 2) Co bude náplní sprintu jednotlivých týmů
  - a) Plán jak dostat kartičky do “done” (jak úkoly zvládnout)
  - b) Rozdělení kartiček do jednotlivých týmů a v nich přiřazení jednotlivým vývojářům
  - c) Členové týmu

LeSS adaptuje také **sprint review**, které proběhne na konci každého sprintu, a to zase ve dvou úrovních.

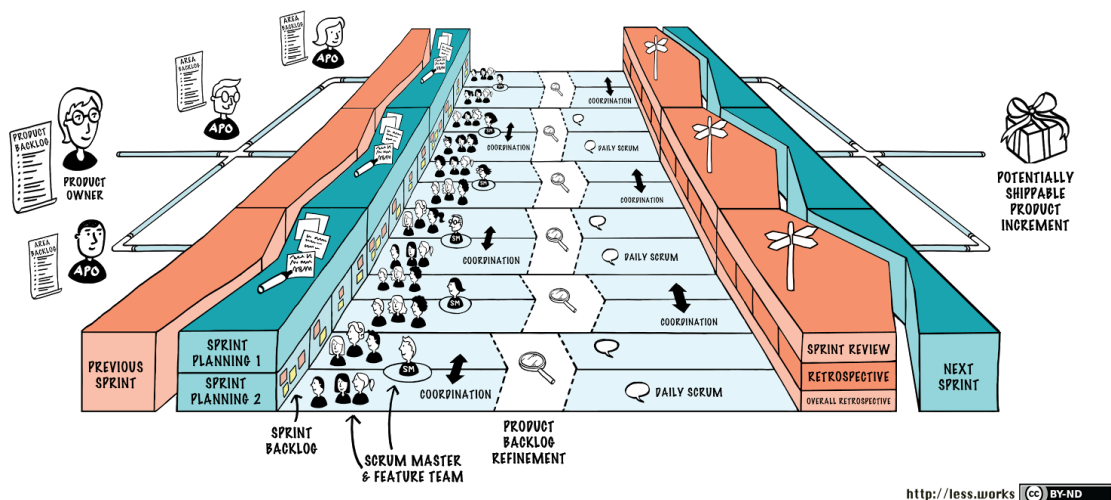
- 1) V týmech
- 2) Týmy s Product Ownerem + zákazníkem (uživateli)

Dále každý sprint probíhá také **retrospektiva**:

- 1) V týmech
- 2) Manažeři, scrum masteri, PO, zástupci týmů

## Less Huge

Tato adaptace frameworku LeSS je orientovaná na velmi velké projekty se stovkami vývojářů. Hlavním rozdílem je rozdělení Product Backlogu na různé oblasti dle požadavků. každá tato oblast pak má svého vlastního oblastního Product Ownera (Area Product Owner), hlavní Product Owner ovšem zůstává (avšak jeho role na projektu je ještě abstraktnější).



Plánování sprintů tedy dostává novou úroveň. Nejdříve se naplánuje celkový sprint (sprinty stále zůstávají synchronizované) a jeho sprint backlog, dále se naplánují oblastní sprint backlogy a ty jsou nakonec rozděleny do týmů v dané oblasti. Zachovává se i sprint review a retrospektiva s tím, že i zde je zase přidána úroveň oblastí.



## 3.2 SAFe

### Základní údaje

SAFe (Scaled Agile Framework) je jeden z možností, kterou mohou organizace zvolit pro škálování svých agilních metodik. Kombinuje principy lean a agilních frameworků. Jeho první release proběhl on-line v roce 2011 a od té doby se neustále vyvíjí skrze své oficiální stránky, kde je zdarma šířen. V současnosti (prosinec 2017) je jeho poslední verze 4.5, která umožňuje mimo jiné škálování DevOps a Continuous Delivery (PRWeb 2017, str. 1).

V závislosti na velikosti týmu můžeme SAFe dělit na tříúrovňový (3-level) pro malé týmy do sta lidí (nebo kombinací několika takových týmů, která spolu úzce nespolupracuje) a čtyřúrovňový pro celé stovky účastníků (Isaacs 2017, str. 1). Jednotlivé úrovně reprezentují klíčové vrstvy cílové architektury – team, program a portfolio. Čtvrtá úroveň přidává tzv. “value stream”, což v rozsáhlejších organizacích reprezentuje organizovanou posloupnost kroků pro stálé dodávání přidané hodnoty koncovému klientovi. Organizuje závislosti a snaží se vytěžit co největší hodnotu z úrovně portfolio.

### Struktura pohledu a týmů

SAFe využívá pro organizaci práce tzv. “big picture” – pohled na chod společnosti jako celku. Pohled portfolio odhalí hlavní body strategie a “epics” (což je v agilním ohledu kolaborace několika stories (Atlassian 2017, str. 1)). Pokud nevyužíváme čtyřúrovňový SAFe, řadíme zde i value stream. Program reprezentuje rozvrh úkolů pro ART (“agile release train”, jemuž je podrobně věnována sekce níže). Pod vrstvou programu nalezneme úroveň team. Hlavní odlišností SAFe od neškálovaných frameworků je právě větší soustředění na analytickou vrstvu, které přesahuje tento team level, na které se běžně orientují metodiky jako Scrum či XP (Atlassian 2017, str. 1).

Jednou ze základních jednotek struktury dělby práce při využití SAFe frameworku je ART – agile release train. ART je někdy označován jako “tým týmů” a skládá se z pěti až deseti týmů, což dohromady dává 50-125 členů ART. Jednotlivé týmy v ART jsou klasickými agilními týmy, které mohou pro svou práci využívat různé metodiky jako např. Scrum, Kanban či XP<sup>1</sup>.

Mimo samotných členů týmů a jejich vedoucích nalezneme v ART ještě dvě role. *Release Train Engineer* je role, jejíž hlavní náplní je exekutivní práce nad výstupem ART jako celku – jsou zodpovědní za organizaci rizik a eskalace vylepšování na programové úrovni. Jako výsledek inspirace metodikou Scrum se někdy této roli přezdívá “Chief Scrum Master” (Solutions IQ

---

<sup>1</sup> Ačkoliv je možné využít několika variant, vzhledem k preskriptivní povaze frameworku je vhodné, aby byla metodika atomických týmů standardizována.

2017, str. 1). Druhou rolí je *Product Manager*, který má na starosti potřeby zákazníka a nese zodpovědnost za výsledný produkt ART.

ART spolupracuje jako celek během období, které je označované jako Program Increment – období osmi až dvanácti týdnů, které se dělí na dvoutýdenní sprinty. Úkolem týmu během tohoto období je zhotovit funkční demo, které přinese znatelnou přidanou hodnotu finálnímu produktu. Předchází mu tzv. “big room planning” – masivní událost sešlosti týmů, která může zabrat jeden až dva dny a jejíž cílem je specifikovat Program Increment pro další iteraci. Tato schůzka může být extrémně náročná na prostředky a realizaci (sehnání vhodného prostoru, získání účasti co nejvíce členů), stejně jako na energii a soustředění jednotlivých účastníků. I přesto je ale tento prvek považován za časem prověřený a efektivní (Burns 2017, str. 1).



*Záznam z meetingu “big room planning” ilustrující rozsah síly SAFe frameworku ()*

## **Aplikace a odezva**

Dle většiny zdrojů je SAFe hodnocen velice pozitivně. Je považován za nejpoblárnější z dostupných škálovacích frameworků pro Scrum a jako jediný, který prokázal dostatečnou dospělost a dokumentovanost pro využití na vysoké korporacní úrovni (Johnson 2017, str. 1). Někdy bývá ale také kritizovaný za svou silnou inklinaci k retrogradnímu plánování (“top-down”), které nejprve stanovuje globální cíle a pak teprve implementuje jejich dosažení na nižších úrovních, důsledkem čehož může být nedostatek zpětné vazby (Management Mania 2017, str. 1). Díky své vysoké preskriptivnosti je také velmi neflexibilní a nenabízí velké množství pro změny.

SAFe obyčejně praktikujeme ve velkých korporacích. Díky své vysoké škálovatelnosti a dvouúrovňovému systému je schopen se přizpůsobit od stovky až po tisíc

spolupracovníků. Dle oficiálních případových studií možnosti tohoto frameworku využívají mj. společnosti jako např. Intel, Phillips a Cisco.

### 3.3 Scrum@Scale

Scrum@Scale je poměrně nový framework, představen byl totiž až v roce 2014 firmou Scrum Inc. Jedním ze zakladatelů samotného Scrumu Jeffem Sutherlandem (Scrum Inc., 2017). Od již zmíněných frameworků LeSS a SAFe se odlišuje především mnohem vyšší modularitou. Můžeme na tento framework nahlížet také jako na seznam otázek, nad kterými se musíme zamyslet, chceme-li dovést náš proces do zdárného konce.

Otázky, na které si chceme odpovědět, dělíme v tomto případě na cyklus Scrum Mastera, cyklus Product Ownera a metriky. Jak již z názvu vyplývá, otázky v každém cyklu se zabývají oblastí, jíž se daný člověk zabývá běžně ve své profesi. Cyklus Scrum mastera se zaměřuje na oblast koordinace týmů, jak co nejlépe zařídit, aby si jednotlivé týmy ve své práci navzájem nepřekážely a aby se v práci nevěnovaly stejné oblasti, zároveň se však snaží jednotlivé týmy neustále a rozvíjet jejich potenciál na maximum. Na druhé straně cyklus Product Ownera se zaměřuje na věci z hlediska produktu a potažmo celé firmy a její strategie. V první řadě se zaměřuje na celkové směřování podniku a jeho strategie, či zda se cíle projektu shodují s cíli podniku. V druhé řadě se však zaměřuje také na rozdělení úkolů mezi týmy, schůzky s týmy, na kterých jim budou úkoly přiděleny a vysvětleny, plánování a následné spuštění release managementu a také zúžitkování zpětné vazby od zákazníka a jeho distribuce zpět k týmům. Součástí frameworku Scrum@Scale jsou také metriky, jimiž se snažíme změřit, nakolik dosahujeme kýženého výsledku, a také transparentnost, jež zajišťuje přístupnost daných metrik komukoliv.

Problémem, který se může vyskytnout v některých organizacích, je velmi komplikované prosazování změn a jejich následná implementace. Vysoce postavení pracovníci, mající dané prosazovací pravomoci totiž většinou nebývají zapojeni do procesu a je obtížné k nim požadavek na změnu dostat. Proto bývá vhodné založit tzv. Executive Action Team, jež má na starost prosazování změn dle potřeby týmů a manažerů zainteresovaných v daném projektu.

## 4 Srovnání frameworků

Sumarizací výhod a nevýhod jednotlivých popsaných frameworků jednoduše získáme možnosti reálných aplikací. Nejsilnější stránkou LeSS je jeho podobnost a návaznost na Scrum, je tedy snadný na pochopení pro organizace, které již Scrumu využívají. Nabízí se tedy pro použití v týmech, které již na nějaké úrovni aplikují Scrum přístupy a kvůli rozšiřování potřebují své metodiky škálovat.

LeSS můžeme v některých ohledech označit jako protipól SAFe, z čehož plyne i jeho nevýhoda, a to sice poměrně velké množství svobody v jeho organizaci. Organizace rozsáhlejších týmů, kde může být komunikace mezi členy a orientace na stejný cíl velmi náročná, obecně vyžaduje co nejstabilnější a nejpevnější přístup. Právě zde můžeme aplikovat SAFe, jehož největší předností je jeho striktní dokumentace a přesně popsané postupy. Vysoká úroveň regulovanosti se ovšem může ukázat jako problém v menších projektech, kde by byl například laxnější přístup LeSS daleko více žádoucí.

Scrum@Scale je skvělý nástroj, pokud hledáme podpůrný prostředek pro zdokonalení již fungujících škálovaných procesů. Jeho abstraktní povaha se ovšem může ukázat jako poměrně slabý podklad pro založení nové organizace.

# Závěr

V této semestrální práci byly připomenuty některé základní principy metodiky Scrum a byly nastíněny některé problémy vyvstávající při jeho aplikaci na větší týmy. Čtenáři je poskytnut základní úvod do problematiky škálování a byla nabídnuta tři konkrétní řešení. Jednotlivé kapitoly byly dedikovány samotným výše uvedeným frameworkům. Provedly rešerši a zkonstruovaly komplexní pohled na základní principy každého z těchto frameworků – jejich základní principy, strukturu, výhody a nevýhody. Ze všech těchto dostupných informací se následně podařilo provést analýzu a nabídnout vlastní úvahu nad reálnou aplikací těchto řešení.

Kromě sumarizace mnoha souvisejících informací z pestrého množství zdrojů podává tato semestrální práce přidanou hodnotu jejich uvedením do kontextu a vyvozením závěrů v kapitole o porovnání. Podařilo se nám naplnit cíl oslovení čtenáře s předpokládanými předchozími znalostmi ve světě agilního vývoje, efektivní rešerše a uvedení oddělených dat do celkového závěru.

# Literatura

BJÖRKHOLM, Tomas, Hans BRATTBERG a Shane HASTIE, 2017. Doing Scrum with Multiple Teams: Comparing Scaling Frameworks. InfoQ [online] [vid. 2017-10-19]. Dostupné z: <https://www.infoq.com/articles/scrum-multiple-teams-frameworks>

RASMUSSEN, Jonathan, nedatováno. Agile vs Waterfall. Agile vs Waterfall [online] [vid. 2017-12-12]. Dostupné z: [http://www.agilenutshell.com/agile\\_vs\\_waterfall](http://www.agilenutshell.com/agile_vs_waterfall)

SCHWABER, Ken a Jeff SUTHERLAND, 2016. The Scrum Guide™ [online] [vid. 2017-10-19]. Dostupné z: <http://www.scrumguides.org/docs/scrumguide/v2016/2016-Scrum-Guide-US.pdf>

HEUSSER, Matt, 2015. Comparing Scaling Agile Frameworks [online] [vid. 2017-10-22]. Dostupné z: <https://www.cio.com/article/2974436/agile-development/comparing-scaling-agile-frameworks.html>

HAYES, Will, 2017. Five Perspectives on Scaling Agile [online] [vid. 2017-10-22]. Dostupné z: [https://insights.sei.cmu.edu/sei\\_blog/2017/02/five-perspectives-on-scaling-agile.html](https://insights.sei.cmu.edu/sei_blog/2017/02/five-perspectives-on-scaling-agile.html)

THE LESS COMPANY B.V., 2017. Overview - Large Scale Scrum (LeSS) [online] [vid. 2017-10-19]. Dostupné z: <https://less.works/>

SCALED AGILE, INC., 2017. SAFe® 4.5 Introduction: Overview of the Scaled Agile Framework® for Lean Enterprises [online]. [vid. 2017-10-19]. Dostupné z: [http://www.scaledagileframework.com/wp-content/uploads/2017/10/White\\_Paper\\_SAFe-4.5.-FINAL.pdf](http://www.scaledagileframework.com/wp-content/uploads/2017/10/White_Paper_SAFe-4.5.-FINAL.pdf)

Scaled Agile, Inc. Releases SAFe® 4.5 Enabling Enterprise Innovation with Lean Startup, Scalable DevOps, and Continuous Delivery. PRWeb [online]. [cit. 2017-12-16]. Dostupné z: <http://www.prweb.com/releases/2017/06/prweb14450991.htm>

ISAACS, Malcolm. What's new in SAFe 4.0? The value stream level and new roles. TechBeacon [online]. [cit. 2017-12-16]. Dostupné z: <https://techbeacon.com/whats-new-safe-40-value-stream-level-new-roles>

Epics, stories, versions, and sprints. Atlassian [online]. [cit. 2017-12-16]. Dostupné z: <https://www.atlassian.com/agile/delivery-vehicles>

Release Train Engineer (RTE). Solutions IQ [online]. [cit. 2017-12-16]. Dostupné z:  
<https://www.solutionsiq.com/agile-glossary/release-train-engineer-rte/>

JOHNSON, Eva. SAFe or Scrum at Scale – Which Framework is Best for You? Intland Software [online]. [cit. 2017-12-16]. Dostupné z:  
<https://intland.com/blog/agile/safe/safe-or-scrum-at-scale-which-framework-is-best-for-you/>

Retrográdní plánování (Top-Down). Management Mania [online]. [cit. 2017-12-16]. Dostupné z: <https://managementmania.com/cs/retrogradni-planovani>

BURNS, Martin. Top 3 reasons to do Big Room Planning [online]. [cit. 2017-12-16]. Dostupné z: <https://www.ca.com/en/blog-highlight/top-3-reasons-big-room-planning.html>

Big Room Planning in Action: Harnessing the Power of Agile at Scale. YouTube [online video]. CA Technologies [cit. 2017-12-16]. Dostupné z:  
<https://www.youtube.com/watch?v=1LR01OMvt1o>

Scrum at Scale, Part 1. Scrum Inc. [online]. [cit. 2017-12-16]. Dostupné z:  
[https://www.scruminc.com/scrum\\_at\\_scale\\_part\\_i/](https://www.scruminc.com/scrum_at_scale_part_i/)

Introduction to Large-Scale Scrum (LeSS) [online]. [cit. 2017-11-29]. Dostupné z:  
<https://www.agilealliance.org/resources/sessions/introduction-to-large-scale-scrum-less/>

Short introduction to LeSS, YouTube [online video]. CA Technologies [cit. 2017-12-17]. Dostupné z: <https://www.youtube.com/watch?v=dmMZ0pZhOgA&t=419s>