Agile Methoden in Grossfirmen: Lösungsansätze

Wer agile Methoden im Grossbetrieb einführt, tut gut daran die folgenden Punkte zu berücksichtigen:

Thema	Ausprägung in KMU mit Softwarefokus	Ausprägung in der Großfirma	Herausforderungen für die Einführung agiler Methoden in der Großirma	Lösungsansätze
Mitarbeiter, Personal	• als Generalisten in der Lage und bereit sein, verschiedene Rollen zu übernehmen	• starke Spezialisierung, Segmentierung in verschiedene, fest zugewiesene und organisatorisch separierte Rollen (Business-Analyst, Entwickler, Tester, Projektleiter) • Fokussierung auf Spezialistenrolle, wird durch ein Bonussystem belohnt	• geringere Bereitschaft und verkümmerte Fähigkeit, Tätigkeiten außerhalb des persönlichen Spezialgebietes „zum Wohle des Teams oder Projekts“ zu übernehmen	• im agilen Team unter den Mitgliedern den Generalistenansatz propagieren und vorleben • über die Zeit in der IT-Organisation wieder stärker auf Generalisten setzen und die Menge an Spezialisten reduzieren
Teams	• Teams bleiben in ihrer Zusammensetzung häufig über längere Zeit bestehen und können sich zu effizienten, eigenständigen „Feature-Fabriken“ entwickeln • Teams befinden sich häufig geografisch am gleichen Ort • hohes Maß an Selbstorganisation innerhalb der Teams, dadurch laufend bessere Zusammenarbeit auf der Metaebene	• Matrixorganisation weit verbreitet • Projektteams häufig geografisch verteilt • Projektteams werden häufig aus Pools von Spezialisten zusammengestellt und schnell wieder aufgelöst • Projektzugehörigkeit oft nur in Teilzeit (eine Person arbeitet an mehreren Projekten gleichzeitig) • Projektleitung hat hohen Stellenwert und trägt Verantwortung für Teamorganisation	• iterative Einführung und Weiterentwicklung von Software wird erschwert durch ständige Wechsel in der Team-Zusammensetzung • Widerstand gegen Selbstorganisation der Teams, insbesondere von Seiten bestehender Team- und Projektleiter • Teams gehen bei jedem Projekt wieder durch die gleichen Findungsprozesse	• Reservierung der Projektmitarbeiter für die weiteren <n> Monate (soweit möglich), um die Team-Zusammensetzung zu erhalten • bestehende Team- und Projektleiter, z.B. in einer Rolle als Projektkoordinator, ins Team integrieren

Thema	Ausprägung in KMU mit Softwarefokus	Ausprägung in der Großfirma	Herausforderungen für die Einführung agiler Methoden in der Großirma	Lösungsansätze
bestehende Entwicklungs-methodik	• häufig bereits mit agilen Ansätzen • Große Änderungsbereitschaft und Flexibilität	• starre Prozesse mit harten Milestones, die fertige Artefakte gemäß Wasserfall-Philosophie verlangen • geringe Änderungsbereitschaft auf höheren Management-stufen und im Gesamtbetrieb • Änderungen sind nicht von heute auf morgen möglich (auch wenn man dies wollte)	• Integration agiler Methoden in die bestehende Methodik und deren Abläufe • Transformation ist schwieriger als ein Greenfield-Ansatz • bestehende Prozesse widersprechen direkt der Einführung agiler Methoden (z. B. harte Prozessmeilensteine) • Einführung muss deutlich wahrnehmbare Verbesserung für die Firma bringen	• schrittweises, evolutionäres Vorgehen • Ansatz der Koexistenz (traditionell und agil) • minimale Prozessanpassungen (z. B. Einführung von Delta-Meilensteinen), damit agiles Vorgehen nicht komplett unmöglich wird
finanzielle Anreize	• abhängig vom Gesamterfolg der Firma oder des Teams durch Beteiligungsprogramm, Team-Bonus	• abhängig von individueller Leistung gemäß festgeschriebener Rollen (Spezialisten) • Management by Objectives (MBO)	• geringe Motivation, im Team Tätigkeiten zu übernehmen, die nicht bonusrelevant sind und nicht zur Erfüllung der persönlichen Ziele beitragen	• zumindest Teile der Bonussumme für erfolgreiche Projekte reservieren
Testing	• Testing als Bestandteil der Entwicklungsarbeit • enge Zusammenarbeit zwischen Entwicklungs- und Testingenieuren • „Eat your own dogfood“-Mentalität	• klare Separierung zwischen Entwicklung und Testing • Testing-Team testet an mehreren Projekten gleichzeitig • „Over-the-wall“-Mentalität • Zeitverlust durch Warteschlangen zwischen den Teams	• Testing muss als enger Bestandteil der Entwicklung akzeptiert werden • wie kann das Testing-Wissen zu einer Applikation vom Entwicklungsteam an eine separate Testorganisation (z. B. für Abnahmetests) übertragen werden? • Umstellung der Testorganisation nötig	• traditionelle Testeinheiten und die Business-Seite müssen näher an die Entwicklung geführt werden • Projektteam begleitet die Software enger auf dem Weg in die Produktion und übernimmt dadurch mehr Verantwortung

Thema	Ausprägung in KMU mit Softwarefokus	Ausprägung in der Großfirma	Herausforderungen für die Einführung agiler Methoden in der Großirma	Lösungsansätze
Firmenstruktur	• flache Hierarchien • partizipativer Führungsstil	• hohe Bedeutung von Hierarchien • direktiver Führungsstil	• Widerspruch zur bisherigen Struktur und zum Führungsstil im Unternehmen • ein Teil des mittleren Managements wird überflüssig oder degradiert • Ängste im oberen Management vor vermeintlichem Kontrollverlust	• Selbstorganisation als Grundlage der agilen Methoden muss von der Führungsebene akzeptiert werden • eine Form der Planung und Reviews auch mit dem Management institutionalisieren • Vorhandenes Bonus-Budget von Einzelpersonen zu Teams übertragen und damit, durch finanzielle Anreize, eine Team-Leistungskultur fördern
Rolle der Software-Entwicklung	• Software-Entwicklung als kommerzielles Geschäft, das verkäufliche Produkte und Dienstleistungen erzeugt • steht im Zentrum, hat großen Einfluss auf Geschäftsabläufe und Art der Zusammenarbeit • Business-Seite ist bereit, eine aktive Rolle in der Entwicklung zu übernehmen	• Software-Entwicklung ist eine Kostenstelle • Software-Entwicklung als Dienstleister für nicht-IT Einheiten, eingebunden in deren Vorgaben und Abläufe • Business-Seite hat wenig Interesse, Zeit und Anreiz, sich stärker in die Entwicklung einzubringen	• Befürchtungen des Managements, dass die Kosten aus dem Ruder laufen • Interesse der Business-Seite an IT-Projekten muss geweckt werden	• dank agilem, iterativen Vorgehen: schnelle erste Lieferungen forcieren und damit bessere Kostenkontrolle und frühere Interventions-möglichkeiten bieten • Business-Seite näher an die IT-Projekte heranziehen, indem man ihr den Nutzen agiler Methoden erlebbar demonstriert

Thema	Ausprägung in KMU mit Softwarefokus	Ausprägung in der Großfirma	Herausforderungen für die Einführung agiler Methoden in der Großirma	Lösungsansätze
Finanzen	• Planungs- und Budgetunsicherheiten sind als Realität akzeptiert • Unsicherheiten wird durch iterative Lieferung begegnet	• hohe Planungs- und Budgetsicherheit durch das Management gewünscht (aber in der Regel nicht erreicht) • detaillierte Pläne suggerieren (nicht vorhandene) Planungssicherheit • komplexe, langatmige Budgetierungs- und Bewilligungsprozesse • jährliche Budgetierung	• beim Management Vertrauen schaffen dass Projekte ihre Ziele erreichen • Anforderungen bezüglich Planungs- und Budgetsicherheit zumindest anfangs kaum zu erfüllen	• Erstellen von Roadmaps anstelle von pseudogenauen Projektplänen • rollende Budgetplanung zumindest innerhalb der Software-Entwicklung • auch für die Finanzen iterativ vorgehen und aufgrund früherer Resultate entscheiden, ob weitere Geldtranchen freigegeben werden
Termine und Releases	• schnelle Entwicklungszyklen • häufige Releases • rasche Anpassung • Flexibilität bei den implementierten Features	• fixe Einführungstermine in der Produktion • so wenige Releases wie möglich • wenig Flexibilität betreffend implementierter Features	• Konflikt mit fixen Release-Fenstern (nur wenige Male pro Jahr) • Flexibilisierung der Produktivsetzung gefordert	• isolierter Pilotbetrieb von Release-Kandidaten für testmäßige Verwendung durch Vertreter der Business-Seite, dadurch Erhöhung des Drucks in Richtung häufigerer Einführungstermine • Verbesserung der Testabdeckung und damit Sicherstellung der Qualität auch bei hoher Release-Kadenz • regelmässige Repriorisierung der Backlog-Einträge durch die Business-Seite und damit Übernahme der Verantwortung für Inhalte der Releases

Thema	Ausprägung in KMU mit Softwarefokus	Ausprägung in der Großfirma	Herausforderungen für die Einführung agiler Methoden in der Großirma	Lösungsansätze
Tooling	• wenige, häufig schlanke Tools für die Projektabwicklung	• große Tooling-Landschaft mit Integration der Tools in die Gesamtprozesskette • Großes Investment in die Schnittstellen und Tool-Plug-Ins	• Integration des gesamten Toolings wird erwartet als Vorbedingung für einen Projektstart, aber die benötigten Tools ergeben sich bei agilen Ansätzen erst aus den konkreten Projekterfahrungen heraus	• Tooling in den ersten Schritten eher im Hintergrund halten • Tooling mit agilen Ansätzen schrittweise und anhand der Anforderungen aus realen Projekten verbessern (ohne Anspruch auf Perfektionismus)
System-Landschaft	• überschaubare Systemlandschaft, klar definierte Schnittstellen des Softwareprodukts	• große, komplexe Systemlandschaft mit vielen Schnittstellen • Projekte haben häufig Anforderungen an andere Systeme, die Erweiterungsarbeiten müssen koordiniert werden	• agiles Vorgehen wird durch Systemlandschaft sicher nicht gefördert • kurzfristige Änderungen an den Systemen und Schnittstellen (Entkopplung) nicht möglich	• möglichst mit isolierten Projekten mit wenigen Schnittstellen beginnen • Anforderungen an Umsysteme hoch priorisieren und traditionell behandeln • schrittweise Entkopplung der Systeme, damit eine bessere Autonomie erreicht werden kann

Thema	Ausprägung in KMU mit Softwarefokus	Ausprägung in der Großfirma	Herausforderungen für die Einführung agiler Methoden in der Großirma	Lösungsansätze
Besetzung der Rolle ProductOwner	• Softwareentwicklung ist sehr nahe am Hauptgeschäft • ein früherer Product Manager als ideale Besetzung für den ProductOwner	• die Business-Seite ist häufig weit weg von der IT und sieht sich als reinen Kunden derselben • keine tägliche Interaktion der Business-Seite mit dem Projekt	• Widerstand oder Passivität ist zu erwarten, wenn die Busineess-Seite einen ProductOwner stellen soll • Alternativen für die Besetzung der ProductOwner Rolle sidn notwendig	• das Bewusstsein für die Wichtigkeit der ProductOwner Rolle muss gefördert werden • Identifikation von Alternativen mit Business-nahen Profile/Rollen innerhalb der IT, z. B. Business Analysten oder Solution Domain Manager • Definition von Kriterien, für welche Situation, welche Art von Besetzung der ProductOwners-Rolle gewünscht ist • Delegation der Entscheidungskompetenz durch die Business-Seite an den ProductOwner (falls er von der IT gestellt werden muss) • Zusage des Managements einholen, dass der ProductOwner genügend Zeit für seine Aufgabe erhält
Projekt-Schnittstellen	• wenige Schnittstellen aus dem Projekt in den Rest der Organisation • große Nähe der verschiedenen Bereiche • Flexibilität und niedriger Formalitätsgrad	• hohe Anzahl Schnittstellen aus dem Projekt in den Rest in die Organisation • tiefe Flexibilität, stark vorgeschriebene Prozesse • hoher Formalitätsgrad	• Schnittstellen können nicht einfach von heute auf morgen eliminiert oder geändert werden	• agile Projekte müssen sich so aufstellen, dass sie die Schnittstellen bedienen können • das Team als Ganzes und nicht eine einzige Person soll die Schnittstellen abdecken

Thema	Ausprägung in KMU mit Softwarefokus	Ausprägung in der Großfirma	Herausforderungen für die Einführung agiler Methoden in der Großirma	Lösungsansätze
Formalitätsgrad (Degree of ceremony) pro Iteration	• tendenziell eher tief, es werden nicht viele Dokumente und weitere Nicht-Software-Artekfakte erwartet	• das Vorgehensmodell schreibt die Erstellung einer beachtlichen Zahl (teilweise über 50) Artefakte vor, die pro Release geliefert werden müssen • zu liefernde Artefakte sind hauptsächlich Dokumente: Von der Requirements-Analyse, über das Solution Design bis zu Testplänen	• große Zahl erwarteter Artefakte kann zu behindernden großem Overhead pro Iteration führen • es muss ein Weg gefunden werden, diesen Aufwand zu minimieren, ihn in der Ausprägung effizienter zu gestalten	• die Artefakte sollten, wo sinnvoll, ebenfalls erstellt werden, allerdings müssen effiziente Wege und Formen gefunden werden, dies zu bewerkstelligen und die Tasks müssen in die Sprints integriert werden • mittels sogenannter Bucket-Tasks werden Artefakte erst nach einer gewissen Anzahl Sprints wieder auf den neuesten Stand gebracht • voller Formalitätsgrad jeweils nur für „echte“ Releases (z. B. alle sechs Monate)