

# Design und Implementierung von Reifegradmodellen für das „Reifegrad-Controlling“ im ERM-Kontext

Eine Veranschaulichung anhand des ERM-MA-Artefakts

Anhand des konzeptuellen Modellierungsprozesses wird gezeigt wie für das ERM-Reifegrad-Controlling ein Reifegradmodell zu designen, bzw. zu konzeptualisieren und operationalisieren ist, das auch IT-mäßig gut implementiert werden kann. Zur Erlangung der dafür benötigten Kompetenzen gilt es den Controlling-Fachbereich mit „informatischer Literalität“ anzureichern.

**Walter S. A. Schwaiger und Micheal Brandstätter**

## 1. Reifegrad von ERM-Implementierungen: Messung, Benchmarking und Monitoring

Im Zeitalter der Digitalisierung ergeben sich neue Möglichkeiten durch die IT-gestützte Ausgestaltung von Planungs- und Steuerungssystemen. Um derartige Systeme in der Controlling-Domäne zu etablieren, bedarf es einer „gut durchdachten“ Spezifizierung der vom System zu erfüllenden Anforderungen, sodass diese in der IT-Domäne erfolgreich umgesetzt werden können. In diesem Artikel geht es um ein Planungs- und Steuerungssystem hinsichtlich des Reifegrads des im Unternehmen implementierten unternehmensweiten Risikomanagements (Enterprise Risk Management, kurz: ERM). In diesem „ERM-Reifegrad-Controllingsystem“ gilt es den Reifegrad, bzw. die Qualität des implementierten ERM-Systems zu messen. Beim Reifegrad handelt es sich um ein latentes Konstrukt, welches nicht direkt messbar ist. Die Definition des Reifegradkonstrukts führt somit zu einem Messproblem. Zur „gut durchdachten“ Spezifizierung gilt es die Konstruktmessung mittels einer geeigneten Messtheorie vorzunehmen, wobei die verschiedenen Ausgestaltungsmöglichkeiten von ERM-Implementierungen – von der Erfüllung von Mindestanforderungen bis zur Best Practice-Ausgestaltung – qualitativ eindeutig zu unterscheiden

sind. Im Rahmen des ERM-Reifegrad-Controllings wird der angestrebte Zielreifegrad vorgegeben und mit dem im Monitoring gemessenen Reifegrad verglichen. Die erforderlichen Maßnahmen zur Verbesserung des Reifegrads liefert das ERM-Reifegradmodell in Form eines Entwicklungsmodells, wobei die für den nächsthöheren Reifegrad erforderlichen ERM-Attribute als Feedback-Information benannt werden.

Angenommen, Sie sind in Ihrem Unternehmen für das ERM-Reifegrad-Controlling zuständig. Zur Nutzung der Digitalisierungsmöglichkeiten soll das Software-Artefakt „ERM-Maturity Assessment“ (kurz: ERM-MA) aufgearbeitet werden, wofür Ihnen die fachliche Projektleitung übertragen wird. Dieses Engineering-Projekt umfasst Design- und Implementierungsaufgaben, sodass neben den Fach- auch IT-Kenntnisse erforderlich sind. Zur erfolgreichen Erfüllung der Ihnen übertragenen Verantwortlichkeiten gilt es insbesondere das ERM-Reifegradmodell „gut durchdacht“ zu spezifizieren, in einerseits inhaltlich angemessener und andererseits auch für die IT-Domäne gut verständlicher Form.

Der zentrale Beitrag des vorliegenden Artikels besteht darin, Ihnen Hilfestellung für eine erfolgreiche Projektleitung zu geben. Zu diesem Zweck wird das Reifegradmodell zuerst entsprechend der konstruktivistischen Messtheorie (vgl. *Rossiter*,



Prof. Dr. **Walter S. A. Schwaiger** ist Ordinarius für Rechnungswesen und Controlling am Institut für Managementwissenschaften der Fakultät für Maschinenwesen und Betriebswissenschaften der TU Wien.



**Michael Brandstätter**, M. Sc. hat seine Abschlussarbeit zum Thema Enterprise Risk Management-Maturity Assessment (ERM-MA), an der TU Wien, verfasst.



Abb. 1: Das ERM-MA-Artefakt – <https://ermma.imw.tuwien.ac.at>

2002) und unter Verwendung des Capability Maturity-(CM)-Messmodells betrachtet. Anschließend wird das dadurch entstandene Verständnis des ERM-Reifegradmodells unter Verwendung des konzeptuellen Modellierungsprozesses konzeptualisiert und operationalisiert. Die dazu verwendete Unified Modeling Language-(UML)-Modellierungssprache stellt sicher, dass das daraus resultierende UML-Reifegradmodell in der IT-Domäne gut verständlich ist. Zur Veranschaulichung wird das „ERM-MA-Artefakt“ (vgl. Abb. 1) verwendet. Für das weitere Verständnis ist es wichtig, dass es in diesem Beitrag nicht um das konkrete Reifegradmodell des ERM-MA-Artefakts geht. Die diesbezüglich interessierte Leserschaft sei auf die Literatur verwiesen (vgl. *Schwaiger/Brandstätter, 2021; Schwaiger/Brandstätter, 2020*). Vielmehr geht es um den konzeptuellen Modellierungsprozess, welcher es erlaubt reifegradbasierte Controllingsysteme für alle möglichen Anwendungsbereiche zu designen.

## 2. IT-Artefakt-Engineering: Design und Implementierung

In der Designphase des Engineering-Projekts ist das dem Software-Artefakt zugrundeliegende ERM-

MA-Konstrukt mangels direkter Messbarkeit zu spezifizieren. Bei der Spezifizierung dieses im ERM-Bereich kontextualisierten MA-Konstrukts liegt folglich ein Messproblem hinsichtlich der Messung von Reifegraden vor. Zur inhaltlich angemessenen Lösung des Messproblems ist die traditionelle positivistische Messtheorie (vgl. *Diamantopoulos, 2005*) nicht geeignet, zumal es beim vorliegenden Messproblem nicht um die Messung von individuellen Empfindungen bzw. Einstellungen (Attitudes) geht. Vielmehr gilt es die Reifegrade anhand von evidenzbasierten Fakten zu messen, was sich mit *Rossiter's* konstruktivistischer Messtheorie („C-OAR-SE-Theorie“ als Abkürzung für Construct-Object/Attribute/Rater Entity-Scale/Enumeration, vgl. *Rossiter, 2002*) gut bewerkstelligen lässt. Dabei wird das ERM-MA-Konstrukt (Construct) anhand der Spezifizierung des Objekts (Object), der Attribute (Attributes) und des Raters/Beurteilers (Rate Entity) definiert.

Für eine Reifegradmessung fehlt in der konstruktivistischen Messtheorie allerdings die explizite Einbeziehung von Attributen für die verschiedenen Reifegrade. Zur Beseitigung dieses Versäumnisses wird eine spezielle Variante, welche als „Capability Maturity (CM)-Messmodell“ bezeichnet wird, des an der Carnegie Mellon University entwickelten CMMI-Reifegradmodells (vgl. *SEI, 2010*) verwendet, wobei CMMI für Capability Maturity Model Integration steht. Beim CMMI-Reifegradmodell werden die Reifegrade anhand einer progressiv kumulativen Anordnung der Attribute gemessen, wobei höhere Reifegrade mit der Erfüllung von sukzessiv kumulierten Attributen verbunden sind. Im

### Zentrale Aussagen

- Progressive Reifegrade sind anhand von evidenzbasierten Fakten zu spezifizieren, um Feedback-Information zur Reifegradverbesserung liefern zu können.
- Digitalisierung erfordert das Zusammenspiel von Controlling und IT-Domäne: Die UML-Modellierung liefert das dazu benötigte gemeinsame Verständnis.

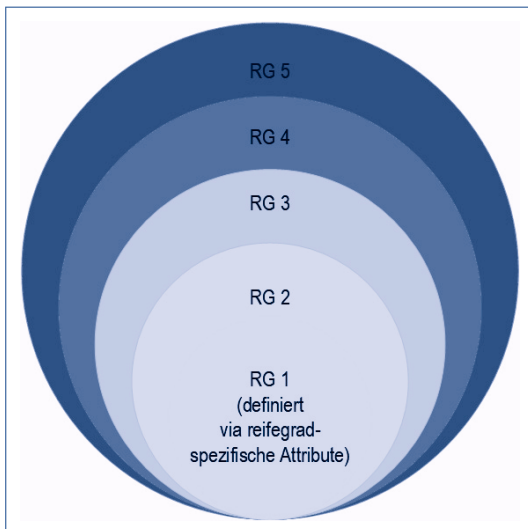


Abb. 2: CM-Messmodell – Nach Reifegraden geordnete Menge von Attributen

ursprünglichen CMM(I)-Reifegradmodell (vgl. Humphrey, 1988) werden faktenbasierte Attribute anhand von dichotomen (binären) Indikatorvariablen gemessen, wobei eine Eins das Vorliegen und eine Null das Nichtvorliegen des Attributs kennzeichnet. Aus mengentheoretischer Sicht handelt es sich dabei um eine nach Reifegraden geordnete Menge an Attributen, welche in Abb. 2 skizziert wird.

Die mit höheren Reifegraden einhergehende progressive Anordnung der Attribute ist auch das Fundament von Entwicklungsmodellen. Diese Modelle ermöglichen die Verfügbarmachung von Feedback-Informationen bezüglich Verbesserungen. Zur Verbesserung des Reifegrads gilt es die noch nicht erreichten Attribute für den nachfolgenden Reifegrad zu erfüllen. Wird bei der Reifegradmessung aller Unternehmen die gleiche Attributsstruktur verwendet, so ist auch die Vergleichbarkeit der damit erzielten Reifegrade für das Benchmarking sichergestellt.

In der Implementierungsphase gilt es das MA-Konstrukt in geeigneter IT-Architektur zu programmieren. Bei der IT-Programmierung ist das Object Oriented Programming (OOP)-Paradigma (vgl. Evens, 2004) von zentraler Bedeutung. Dabei werden Objekte mit gleicher Struktur und gleichem Verhalten in Entitäten (Entities) zusammengefasst, die zwischen ihnen bestehenden Beziehungen (Relationships) betrachtet, und das entstehende Entity/Relationship (ER)-Gefüge mittels IT-Klassenkonstrukten softwaremäßig implementiert.

### 3. ERM-MA-Artefakt: Konzeptuelle Modellierungsperspektive

Zur inhaltlichen Kontextualisierung des ERM-Begriffs wird das COSO-ERM-Framework (vgl. COSO-ERM, 2021) verwendet. Kompakt zusammengefasst definiert es ein ERM-System als die Verankerung

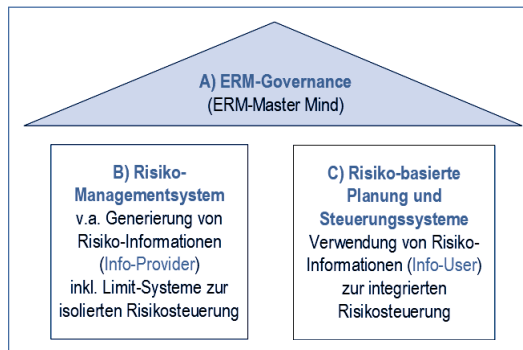


Abb. 3: ERM-MIS – COSO-ERM als Managementinformationssystem

von Risikoüberlegungen in den im Unternehmen eingerichteten Planungs- und Steuerungssystemen. Als Framework ermöglicht es zudem verschiedene Varianten der Ausgestaltung solcher Systeme, sodass eine qualitative Anordnung möglich wird, von der einfachsten Variante, wobei Mindestanforderungen erfüllt werden, bis zur fortschrittlichsten Variante, bei welcher sämtliche Best Practice-Vorgaben vorliegen.

Zur inhaltlichen Präzisierung wird die informatische Perspektive verwendet. Dabei wird COSO-ERM als Managementinformationssystem (ERM-MIS) verstanden (vgl. Schwaiger/Brandstätter, 2021; Schwaiger/Brandstätter, 2020), wobei in einer ERM-Governance A) festgelegt wird, welche Risikoinformationen B) generiert und in welcher Form diese in den bestehenden Managementsystemen C) verwendet werden. Mit den drei in Abb. 3 skizzierten ERM-MIS-Dimensionen wird eine 3-dimensionale Grundstruktur des ERM-MA-Konstrukts definiert. Durch die Hinzunahme von jeweils fünf Reifegradstufen, welche die verschiedene ERM-MIS-Ausgestaltungen in den drei Dimensionen beinhalten, ergibt sich für das ERM-MA-Konstrukt eine 3-dimensional/5-stufige, bzw. 3x5-Grundstruktur (Core Structure). Folglich ist die Grundstruktur des ERM-MA-Konstrukts spezifiziert und das daraus resultierende Modell wird als „ERM-Reifegradmodell“ bezeichnet.

Im Lichte der konstruktivistischen C-OAR-SE-Messtheorie (vgl. Rossiter, 2002) handelt es sich beim CM-Messmodell um ein Messkonstrukt (Construct) in Form eines „abstrakten Kollektivobjekts“ (Abstract Collective Object) mit „geformten Attributen“ (Formed Attribut) für ein „Self Assessment“ (Rater Entity), wobei das Vorliegen der Attribute mit „dichotomer Antwortskala“ (Scale) gemessen und die Reifegrade durch „Multiplikation der Indikatorvariablen“ (Enumeration) ermittelt werden.

Die bisherigen Ausführungen haben sich auf das Design des ERM-MA-Konstrukts bezogen, in welche insbesondere die Controlling-Domäne involviert ist und das nach der C-OAR-SE-Messtheorie charakterisierte CM-Messmodell verwendet wurde. Bei Betrachtung des ERM-MA-Artefakts gilt es aber auch die Implementierung des Konstrukts in eine

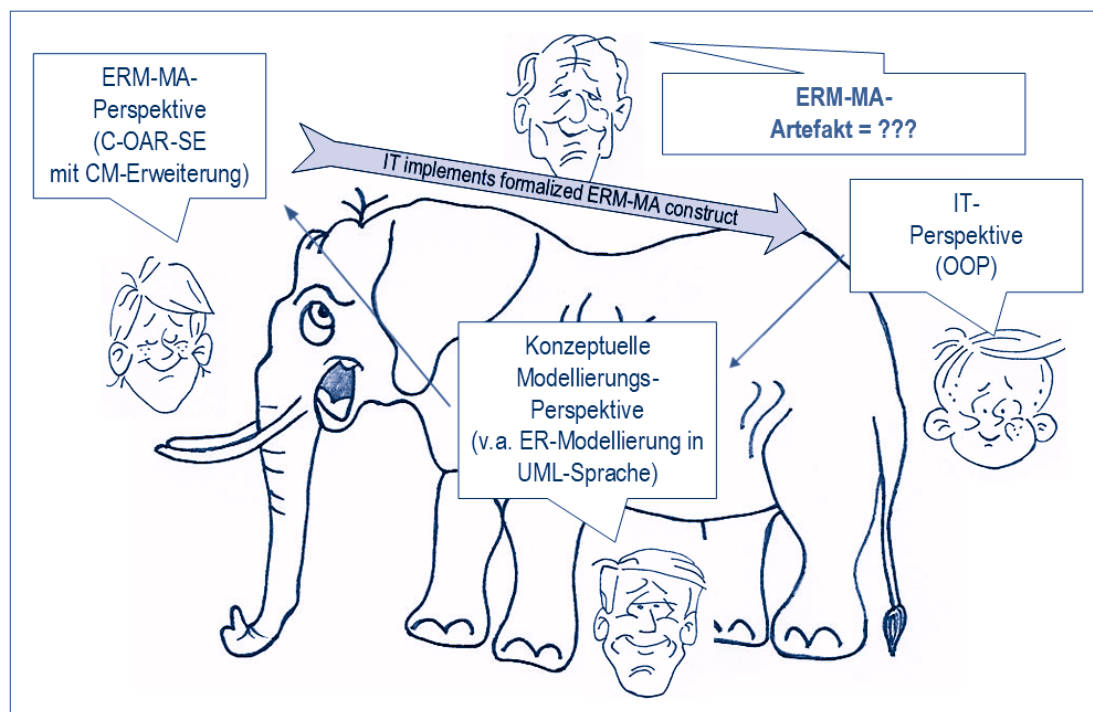


Abb. 4: ERM-MA-Artefakt – The Blind Men and the Elephant (John Godfrey Saxe)

entsprechende Software-Applikation zu berücksichtigen, welche in der IT-Domäne erfolgt. Die Expertisen und Denkweisen in den beiden Domänen sind aber grundsätzlich verschieden. Während in der Controlling-Domäne das Denken in Management- und Messkonzepten vorherrscht, wird in der IT-Domäne insbesondere in OOP-Programmierkonzepten gedacht (vgl. *Evans*, 2004).

Die sich daraus ergebende Bruchstelle zwischen dem Design des ERM-MA-Konstrukts im Controlling und dessen Implementierung in der IT-Domäne gilt es zu beseitigen. Zur Veranschaulichung dieser Problematik wird die in **Abb. 4** dargestellte Metapher „The Blind Men and the Elephant“ (vgl. *Saxe*, 2021) verwendet. Der „Weise König“ ist sich der unterschiedlichen Sichten auf das ERM-MA-Artefakt bewusst, und er erkennt, dass es zur Elimination der Bruchstelle eines „formalisierten ERM-MA-Konstrukts“ bedarf, welches in der IT-Domäne verstanden wird und auch implementierbar ist. Zu diesem Zweck bringt er die „konzeptuelle Modellierungsperspektive“ (vgl. *Mylopoulos*, 1992) ins Spiel, welche auf einer „Entity/Relationship (ER)-Modellierung“ (vgl. *Chen*, 1975) in der Unified Modeling Language (UML)-Sprache (vgl. *Object M. Group*, 2021) basiert.

Die UML-Sprache ist formal präzise formuliert. Somit wird sie nicht nur in der IT-Domäne verstanden, sondern kann auch von Computern verarbeitet werden. Des Weiteren inkludieren die Klassendiagramme (Class Diagram, kurz: CD) der UML auch Grundkonzepte der objektorientierten Programmierung (OOP), sodass sowohl Entities als auch Relationships als Klassen, d. h. als Mengen von Objekten modellierbar sind. Dadurch wird es möglich, die

dem ERM-MA-Konstrukt zugrundeliegende ER-Modellstruktur in der UML/CD-Sprache generisch zu modellieren und sodann OOP-mäßig zu implementieren. Nachfolgend wird gezeigt wie dies konkret funktionieren kann.

#### 4. ERM-MA-Konstrukt: Konzeptualisierung – ERM-Reifegradmodell

Ausgestattet mit der C-OAR-SE-Charakterisierung des CM-Messmodells liegt nun ein Denkraum vor, welcher die konzeptuelle Modellierung des ERM-MA-Konstrukts ermöglicht. Zu diesem Zweck wird die C-OAR-SE-Spezifikation in dessen zwei Bestandteile zerlegt, und zwar den sich auf das Konstrukt beziehenden „Construct“-Teil (C-OAR) und den sich auf die Frage/Antwort-Kombinationen (Question/Answer Items, kurz: QA-Items) beziehenden „QA-Item“-Teil (SE). Diese Zerlegung hilft, Misspezifikationen bei der Messung von latenten Konstrukten zu beseitigen, indem die Messung nicht bei der Erstellung von QA-Items beginnt. Vielmehr beginnt die Messung mit der Construct-Spezifizierung. Dies hat den Vorteil, dass sich bei der nachfolgenden QA-Item-Spezifizierung die inhaltliche Korrektheit (Content Validity) der QA-Items durch Abgleich der formulierten Frage/Antwort-Kombinationen mit den jeweils geforderten Attributen überprüfen und sicherstellen lässt. Im Predictive Validity Framework (vgl. *Bisbe/Batista-Foguet/Chenhall*, 2007) wird diese Vorgehensweise als Lösung des Missspezifikationsproblems postuliert, wobei die „Konzeptualisierung“ (Construct-Spezifizierung) der „Operationalisierung“ (QA-Item-Spezifizierung) vorgelagert ist.

**SCHWERPUNKT**

Nachfolgend wird die Terminologie dieses Frameworks verwendet, um den „Prozess der konzeptuellen Modellierung“ des ERM-Reifegradmodells in drei Phasen zu gliedern, und zwar die „Dimensionierung“, die „Konzeptualisierung“ und die „Operationalisierung“.

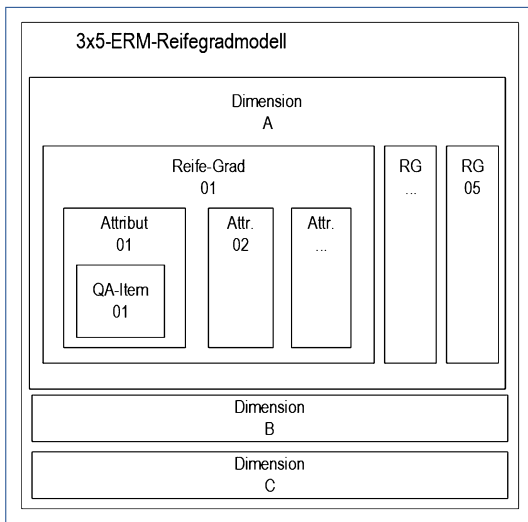
Die Modellierungsphase der Dimensionierung erfolgte bereits bei der Definition des ERM-Reifegradmodells, wobei die Grundstruktur des ERM-MA-Konstrukts festgelegt wurde. Das dabei definierte 3x5-ERM-Reifegradmodell enthält für jeden Reifegrad eine qualitativ unterschiedliche Ausgestaltung in den drei ERM-MIS-Dimensionen. In der Konzeptualisierungsphase werden die unterschiedlichen ERM-MIS-Ausgestaltungen des CM-Messmodells als abstraktes Kollektivobjekt im Construct-Teil (C-OAR) gesehen. Der Dimensionalität des Konstrukts und der Reifegrade entsprechend,

besteht das abstrakte Kollektivobjekt aus fünf unterschiedlichen ERM-MIS-Ausprägungen in drei Dimensionen. Die Charakteristika der jeweiligen Ausprägungen werden anhand von Attributen spezifiziert. Schließlich wird zur Messung des Vorliegens der Attribute jedem Attribut genau ein QA-Item zugeordnet.

In **Abb. 5** wird das derart konzeptualisierte ERM-Reifegradmodell skizziert. Dabei zeigt sich die informatische Struktur des Reifegradmodells als ein Kollektiv von drei Dimensionen, wobei es in jeder Dimension fünf Reifegrade gibt, welche anhand von über QA-Items gemessenen Attributen definiert sind.

Nach der Festlegung der informatischen Modellstruktur erfolgt die Kontextualisierung des ERM-Reifegradmodells, womit sich die anfänglich abstrahierte Betrachtung des Reifegradmodells wiederum explizit auf den ERM-Kontext bezieht. Die Komplexität des ERM-Reifegradmodells macht eine „Top-Down-Kontextualisierung“ erforderlich, welche bei der Kontextualisierung des COSO-ERM als ERM-MIS (**Abb. 3**) beginnt und dann zur Kontextualisierung der Dimensionen bzw. Subdimensionen geht, wo schließlich die Attribute für die verschiedenen Reifegrade festgelegt werden. Entsprechend der in diesem Artikel verfolgten Zielsetzung wird an dieser Stelle auf die Kontextualisierung nur überblicksmäßig eingegangen. Die sich für die detaillierte Kontextualisierung interessierende Leserschaft sei auf die Literatur verwiesen (vgl. *Schwaiger/Brandstätter, 2021; Schwaiger/Brandstätter, 2020*).

**Abb. 6** zeigt die Kontextualisierung der drei Dimensionen des ERM-Reifegradmodells. Dabei ist zu erkennen, dass mit Zunahme der Reifegrade die Ausprägungen in den drei Dimension sukzessive



**Abb. 5: 3x5-ERM-Reifegradmodell – Skizzierung der Modellstruktur**

		Reifegrade				
		RG 1	RG 2	RG 3	RG 4	RG 5
<b>Dimensionen</b>	<b>A. ERM-Governance</b> A1: Risikostrategie A2: Risikoverständnis A3: Risikoorganisation	Prozess-Perspektive in partiellen Bereichen (Silo-Sicht)	Prozess-Perspektive inkl. Prüfung und Management (single loop)	Unternehmensweite (holistisch-differenzierte) Perspektive	Unternehmensübergreifende (corporate) Perspektive (double loop)	Vom Top-Management interaktiv gemanagte Systeme
	<b>B. Risiko-Managementsystem</b> B1: RM-Prozess B2: RM-Schulungssystem B3: RM-Informationssystem	Risiko-management-Prozess	Risiko-management-Prozess (inkl. Monitoring und Review) (single loop)	Unternehmensweit standardisierter RM-Prozess (inkl. ...)	Unternehmensweiter und -übergreifender RM-Prozess (inkl. ...) (double loop)	
	<b>C. Risiko-basierte Planungs- und Steuerungssysteme</b> C1: Strategisches Management C2: Finanzielles Management C3: Operatives Management	Risiko-Limit-Systeme in partiellen Bereichen	Key Risk-basierte Planung (inkl. Strategie- bzw. Zielfestlegung)	Key Risk-basierte Steuerungssysteme (i.e. Performance-Management)	Management-Systeme mit Risiko-adjustierten Performance-Kennzahlen	

**Abb. 6: 3x5-ERM-Reifegradmodell – Kontextualisierung der drei ERM-Dimensionen (entnommen aus Schwaiger/Brandstätter, 2021, S. 10)**

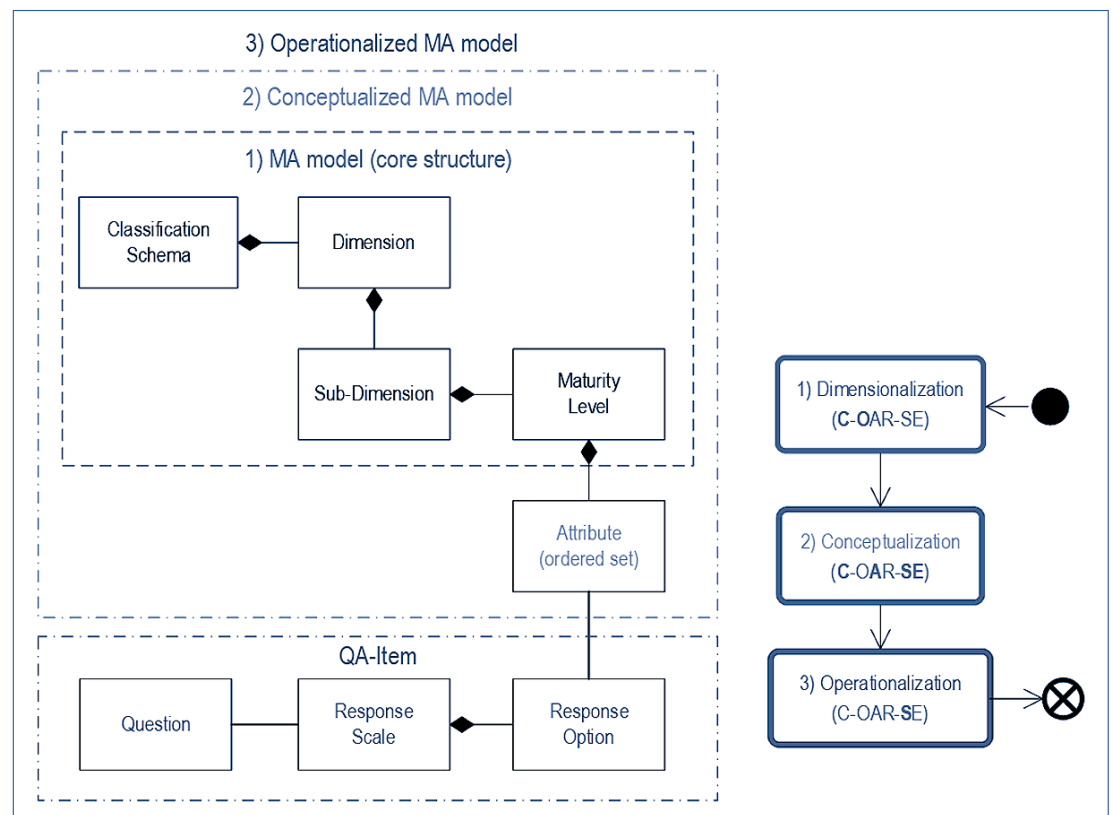
höhere Anforderungen erfüllen. Des Weiteren gliedern sich die Dimensionen in drei jeweilige Subdimensionen, anhand derer die Dimensionen definiert werden. In jeder Subdimension sind wiederum fünf Reifegrade mit den dazugehörigen Attributen definiert, womit die Top-Down-Kontextualisierung auf Ebene der Subdimensionen vollzogen wird. Die Reifegradattribute der Subdimensionen beziehen sich auf evidenzbasierte Fakten hinsichtlich der jeweiligen Ausprägungen, welche mit dichotomen Indikatorvariablen gemessen werden. Ein Beispiel für ein dichotomes Attribut für den Reifegrad 1 in der Subdimension „Risikomanagementprozess“ (B1 in Abb. 6) ist „im Unternehmen werden Risiken identifiziert“, wobei es sich um ein Faktum handelt, welches im Unternehmen vorliegt oder nicht. Die konkrete Ausformulierung der sich auf die Attribute beziehenden QA-Items erfolgt nicht in der Konzeptualisierungs-, sondern in der ihr nachgelagerten Operationalisierungsphase.

**5. ERM-MA-Konstrukt: Konzeptualisierung – UML-Reifegradmodell**

Die in Abb. 5 skizzierte Modellstruktur des ERM-Reifegradmodells ist mangels hinreichender Formalisierung für die IT-Implementierung noch nicht ausreichend. Dieses Problem wird durch die Verwendung der UML/CD-Sprache in der Entity/Relationship-Modellierung behoben. Das sich aus dieser Modellierung ergebende „UML-Reifegradmodell“ ist in Abb. 7 zu sehen.

Zumal UML eine graphische Modellierungssprache ist, sind die UML-Sprachkonzepte in Form von Symbolen verfügbar. Bei den Rechtecken handelt es sich um Klassen (Entity). Die Verbindungen zwischen den Klassen sind Assoziationen (Relationship). Um die Modellierung der Klassendiagramme (Class Diagram) einfach zu halten, werden nur Linien bzw. Kompositionen für die Assoziationen verwendet. Die kompositionelle Assoziation ist an der ausgefüllten Raute am Assoziationsende zu erkennen. Sie verbindet in Richtung der als Ganzheit (z. B. die menschliche Hand) verstandenen Klasse die diesbezüglich existentiell abhängigen Objekte der assoziierten Klasse (z. B. Finger). Auf der rechten Seite der Abbildung wird zusätzlich ein UML-Aktivitätsdiagramm (Activity Diagram) bezüglich des 3-phasigen Modellierungsprozesses dargestellt. Zur UML/AD-Modellierung werden abgerundete Rechtecke für die Aktivitäten verwendet, welche durch Pfeile miteinander verbunden sind, wobei die Spitze des Pfeils die Abfolge anzeigt. Schließlich werden noch ein ausgefüllter bzw. gekreuzter Kreis verwendet, um den Start bzw. das Ende des Prozesses anzuzeigen.

Abb. 7 enthält auf der linken Seite das UML-Reifegradmodell. In der Dimensionierungsphase wird das abstrakte Kollektivobjekt des CM-Messmodells nunmehr anhand von kompositionellen Assoziationen zwischen der Classification Schema-, der Dimension-, der Sub-Dimension- und der Maturity Level-Klasse modelliert. Diese Modellierung hat den Vorteil, dass sie nun auch in der IT-Domäne



**Abb. 7: UML-Reifegradmodell – ER-Modellstruktur und Modellierungsprozess**

verstanden und in der OOP-Implementierung umgesetzt werden kann.

In der Konzeptualisierungsphase geht es insbesondere um die Festlegung der zur Reifegradspezifikation verwendeten Attribute, und zwar in inhaltlicher als auch formaler Sicht. Inhaltlich werden die Reifegradattribute im Rahmen der Top-Down-Kontextualisierung bestimmt. Die formalen Eigenschaften der Attribute ergeben sich aus der beabsichtigten Messung des Vorliegens von evidenzbasierten Fakten mit Indikatorvariablen. Dadurch wird eine dichotome Antwortskala bei den QA-Items induziert. Des Weiteren erfolgt im Rahmen der Enumeration die Reifegradermittlung anhand des Wahrheitsgehalts der den Reifegraden kumulativ zugeordneten Attributen. Durch die Verwendung von dichotomen Indikatorvariablen liegt der Reifegrad vor, wenn das Produkt der ihm zugeordneten Attribute entsprechend der Booleschen Logik gleich eins und folglich wahr ist.

### 6. ERM-MA-Konstrukt: Operationalisierung – UML-Reifegradmodell

In der Operationalisierungsphase werden den in der Konzeptualisierungsphase spezifizierten Reifegradattributen valide und reliable QA-Items zugeordnet. Wie in **Abb. 7** zu sehen ist, wird dabei der mit der Attribute-Klasse assoziierte QA-Item-Ausdruck durch die Question-, Response Scale- und Response Option-Klasse inhaltlich spezifiziert. Dabei werden Antwortskalen mit den Fragen verbunden, welche wiederum kompositionell mit Antwortmöglichkeiten verbunden sind. Beachtenswert ist auch die nicht kompositionelle Beziehung zwischen Response Option und Attribute, was die eindeutige Messung des Vorliegens eines Attributs durch genau eine Antwortmöglichkeit kennzeichnet.

Zumal es sich bei den Reifegradattributen um Indikatorvariablen zur Messung von evidenzbasierten Fakten handelt, werden zu deren Operationalisierung ausschließlich dichotome Antwortskalen zur Beantwortung der Attributsfragen benötigt. Die Messung der hinter den Reifegraden stehenden Qualitäten erfolgt somit durch Vorliegen der die Reifegrade definierenden Fakten.

Das in **Abb. 7** dargestellte UML-Reifegradmodell inkludiert die Anforderungen hinsichtlich des Benchmarking (standardisierte Reifegradattribute und diesbezügliche QA-Items) und der zur Verbesserung von Reifegraden erforderlichen Feedback-Informationen aus einem Entwicklungsmodell (geordnete Menge an Reifegradattributen). Um die dazu notwendigen Reifegradmessungen durchführen, die Feedback-Informationen generieren und das Benchmarking vornehmen zu können, wird das UML-Reifegradmodell in Web-basierter Informationstechnologie implementiert. Diese Implemen-

### Implikationen für die Praxis

- Eine „gut durchdachte“ Spezifikation von Reifegradmodellen basiert auf der konstruktivistischen Messtheorie und erfolgt in der UML-Modellierungssprache.
- UML-Sprachkompetenzen in der Controlling-Domäne sind der Schlüssel für erfolgreiche Engineering-Projekte.

tierung generiert das ERM-MA-Artefakt und sie basiert auf einer 3-schichtigen IT-Architektur, wobei zwischen der graphischen Benutzeroberfläche (GUI), der Ebene der Geschäftslogik (Business Logic) und der Datenebene (Persistenz) unterschieden wird. Die konkrete Ausgestaltung der 3-Schicht-Architektur erfordert ein tieferes Eintauchen in die IT-Domäne, was aber nicht Intention des vorliegenden Artikels ist.

### 7. Konklusion

In diesem Artikel wird gezeigt, wie ein ERM-Reifegradmodell in der UML-Sprache „gut durchdacht“ zu designen, bzw. zu Konzeptualisieren und zu operationalisieren ist, damit es einerseits für das ERM-Reifegrad-Controlling geeignet und andererseits in der IT-Domäne gut implementierbar ist. Im Mittelpunkt steht dabei die große Diskrepanz in den Expertisen und Denkweisen in der für das Design derartiger Software-Artefakte zuständigen Controlling-Domäne und in der für dessen Implementierung zuständigen IT-Domäne. Im durch die Funk-Stiftung geförderten „ERM-MA“-Projekt (vgl. Funk-Stiftung) hat sich gezeigt, dass für das dabei designte und implementierte ERM-MA-Artefakt (**Abb. 1**) ein gemeinsames Verständnis in der Controlling- und IT-Domäne essentiell für den erfolgreichen Engineering-Projektabschluss war. Des Weiteren hat sich gezeigt, dass die Erweiterung der Kompetenzen und der Modellierungssprachen in der Controlling-Domäne hinsichtlich einer konzeptuellen Modellierung eine Möglichkeit bietet, die verständnisbezogene Bruchstelle zwischen den beiden Domänen zu beseitigen. Die zentrale Intention des Beitrags ist zu zeigen, wie die Integration der „informatischen Literalität“ (Computational Thinking) in die Controlling-Domäne aussieht und wie der konzeptuelle Modellierungsprozess und die damit verbundene ER-Modellierung in der UML-Sprache funktioniert.

### Literatur

- Bisbe, J./Batista-Foguet J. M./Chenhall, R., Defining management accounting constructs: A methodological note on the risks of conceptual misspecification, in: Accounting, Organizations and Society, 32. Jg. (2007), H. 6, S. 789–820.
- Chen, P. P. S., The entity-relationship model: Toward a unified view of data, in: ACM SIGIR Forum, 10. Jg. (1975), H. 1, S. 9–36.

- COSO-ERM, Enterprise Risk Management: Integrating with Strategy and Performance (Committee of Sponsoring Organizations of the Treadway Commission), <https://www.coso.org/Pages/default.aspx>, Stand: 15.09.2021.
- *Diamantopoulos, A.*, The C-OAR-SE procedure for scale development in marketing: A comment, in: International Journal of Research in Marketing, 22. Jg. (2005), H. 1, S. 1–9.
- *Evans E.*, Domain-Driven Design: Tackling Complexity in the Heart of Software, Addison-Wesley, Boston, 2004
- Funk-Stiftung, Projekte – ERM-MA-Online Tool, <https://www.funk-stiftung.org/de/projekte/risikomanagement/ermma-enterprise-risk-management-maturity-assessment-monitoring-tool>, Stand: 15.09.2021.
- *Humphrey, W. S.*, Characterizing the Software Process: A Maturity Framework, in: IEEE Software, 5. Jg. (1988), H. 2, S. 73–79.
- *Mylopoulos, J.*, Conceptual modelling and Telos, in: Loucopoulos P./R. Zicari (Hrsg.), Conceptual Modeling, Databases, and CASE: An integrated view of information systems development, New York 1992, S. 49–68.
- Object M. Group, Unified Modeling Language, <https://www.uml.org>, Stand: 15.09.2021.
- *Rossiter, J. R.*, The C-OAR-SE procedure for scale development in marketing, in: International Journal of Research in Marketing, 19. Jg. (2002), H. 4, S. 305–335.
- *Saxe, J. G.*, The Blind Men and the Elephant, [https://en.wikisource.org/wiki/The\\_poems\\_of\\_John\\_Godfrey\\_Saxe/The\\_Blind\\_Men\\_and\\_the\\_Elephant](https://en.wikisource.org/wiki/The_poems_of_John_Godfrey_Saxe/The_Blind_Men_and_the_Elephant), Stand: 15.09.2021.
- *Schwaiger, W. S. A./Brandstätter, M.*, Unternehmensweites Risikomanagement: Reife der Implementierung – Fitness Check, in: WingBusiness, 54. Jg. (2021), H. 1, S. 9–14.
- *Schwaiger, W. S. A./Brandstätter, M.*, Qualitätsmessung von ERM-Systemen anhand von Reifegraden, in: Controller Magazin (CM), 46. Jg. (2020), H. 2, S. 73–77.
- SEI, CMMI for Development, Version 1.3: Improving Processes for Better Products and Services (CMMI-DEV, V1.3), Carnegie Mellon, Software Engineering Institute, Pittsburgh 2010.

---

### Stichwörter

# Capability Maturity-(CM)-Messmodell # Informatische Literalität # Konstruktivistische Messtheorie # Konzeptueller Modellierungsprozess # Unified Modeling Language (UML)

---

### Keywords

# Capability Maturity (CM) measurement model # Computational thinking # Constructivist measurement theory # Conceptual modeling process # Unified Modeling Language (UML)

---

### Summary

Using the conceptual modeling process in the ERM maturity level management it will be shown how to design, i.e. to conceptualize and operationalize a maturity model in a way suitable for implementation in the IT domain. For achieving these competences, the management control domain has to be enhanced by computational thinking considerations.