

DAIMLER

MBEES 2010 - Research Abstract

Ein Framework zur automatisierten Ermittlung der Modellqualität bei eingebetteten Systemen

Jan Scheible (jan.scheible@daimler.com)

Daimler AG – Group Research and Advanced Engineering

Inhalt

Problemstellung

Lösungsansatz

Prototyp

Zusammenfassung und Ausblick

Problemstellung

Ausgangssituation

- Automobilindustrie verwendet verstärkt Modellbasierte Softwareentwicklung
 - Umfang der Modelle wird immer größer
 - Beispiel eines großen Matlab Simulink-Modells aus dem PKW-Bereich:
 - ca. 15.000 Blöcke
 - 700 Subsysteme
 - Subsystemhierarchie mit 16 Ebenen
 - hoher Zeitdruck in der Entwicklung
 - Entwickler haben trotz höherer Abstraktionsebene viele Freiheiten
- viele Möglichkeiten für potentielle Fehler

Inhalt

Problemstellung

Lösungsansatz

Prototyp

Zusammenfassung und Ausblick

Lösungsansatz

Beschreibung

- automatisierte Ermittlung der Modellqualität
 - Ermittlung eines Ist-Stands
 - abschließende Bewertung der Modellqualität
- Handlungsempfehlungen zur Verbesserung der Modellqualität auf Basis der Qualitätsbewertung

Modellqualität

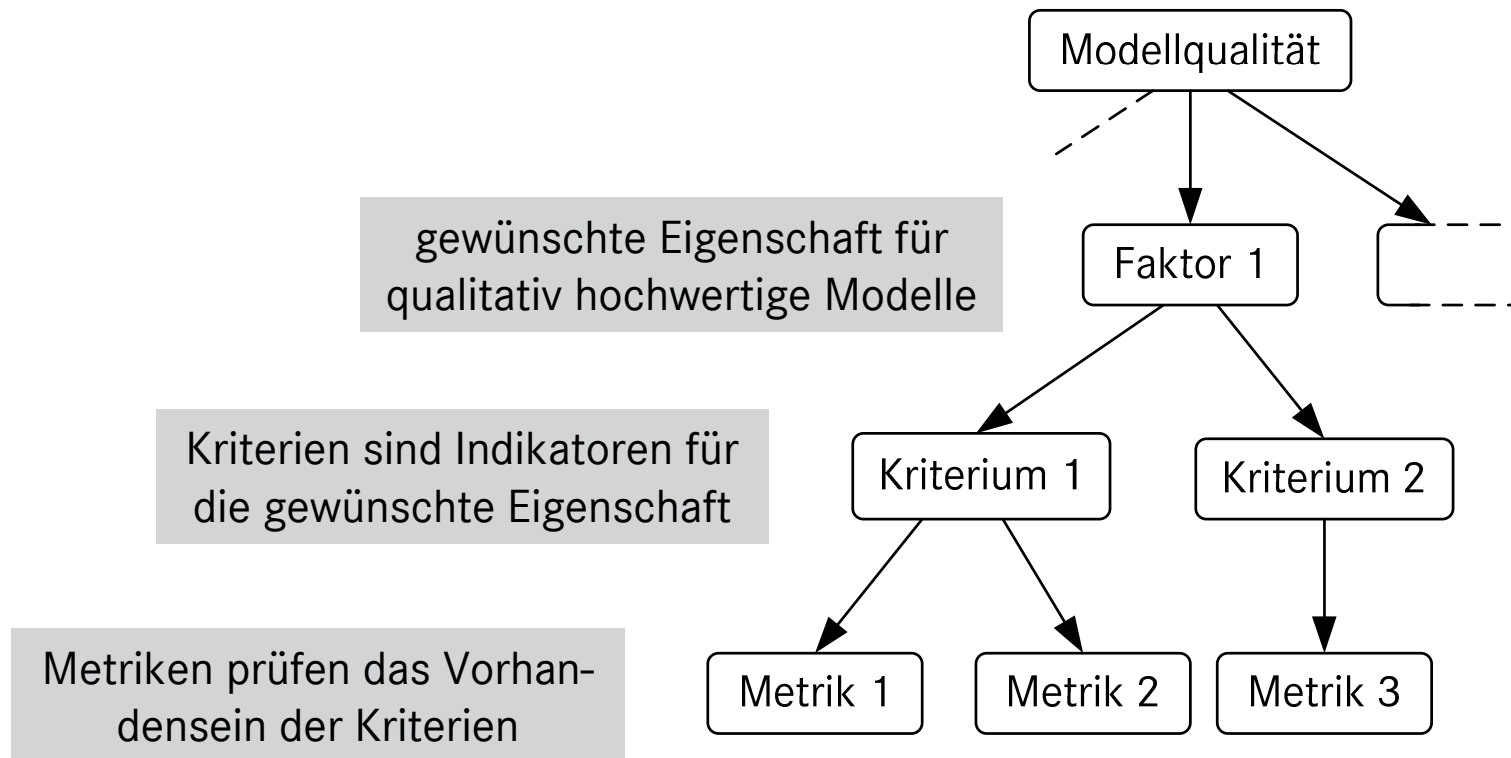
Wann hat ein Modell eine hohe Qualität?

- Kriterien mit positivem Einfluss auf gewünschte Faktoren (z.B. Wartbarkeit, Lesbarkeit oder Robustheit) sind vorhanden
- Modell erfüllt alle Kriterien \triangleq hohe Qualität
- alle Kriterien sind von Interesse, die einen Rückschluss auf gewünschte Faktoren zulassen

Qualitätsmodell (1/3)

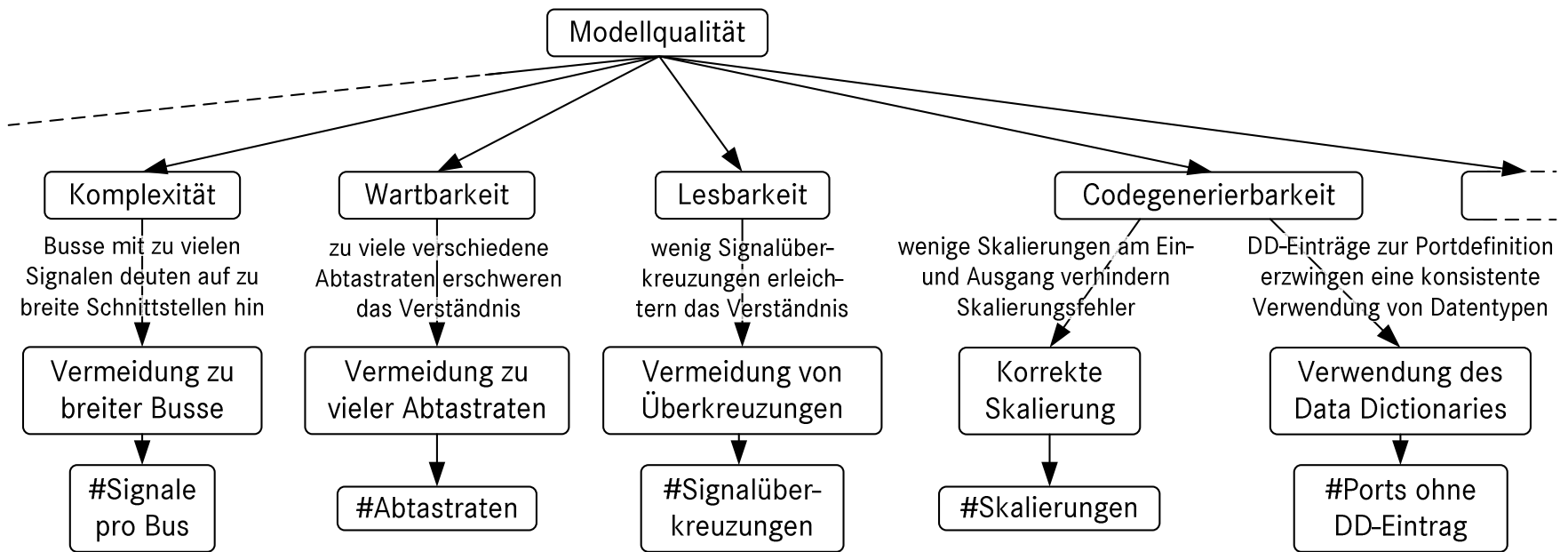
Wie kann die Modellqualität objektiv erfasst werden?

Mit Hilfe eines Qualitätsmodells:



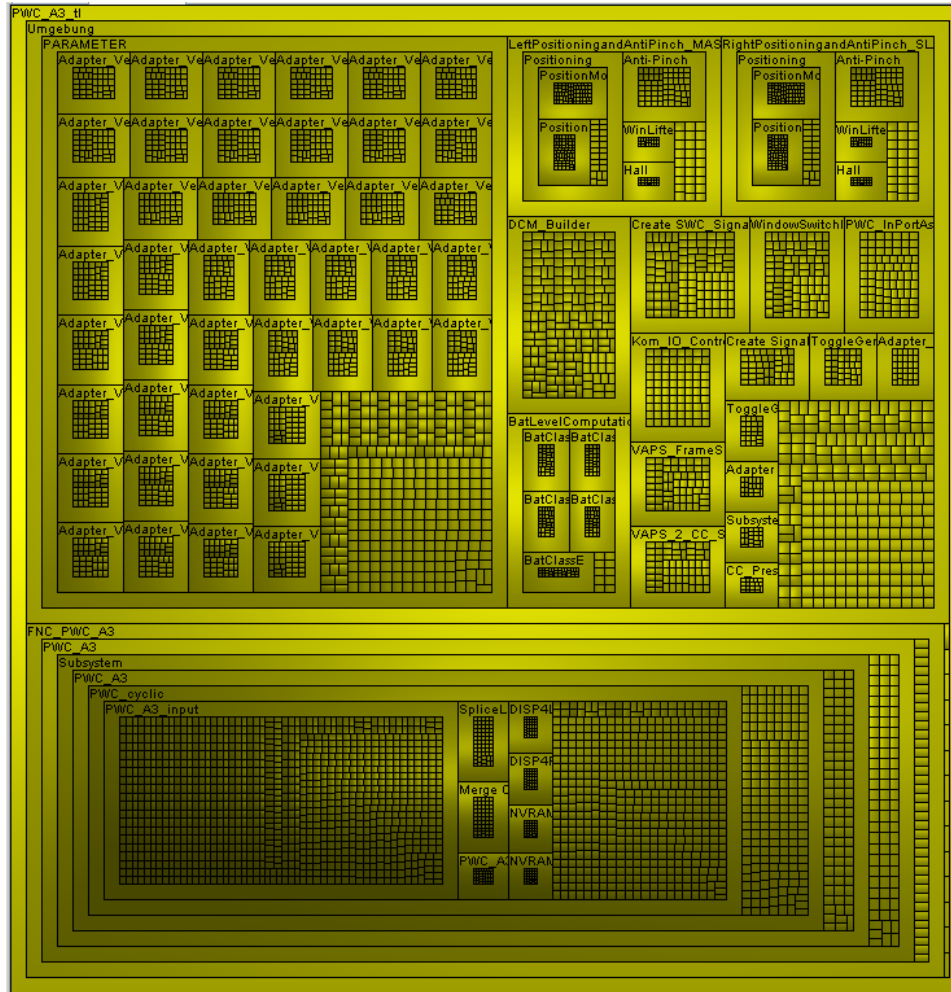
Qualitätsmodell (2/3)

Wie sieht das instanzierte Qualitätsmodell aus?



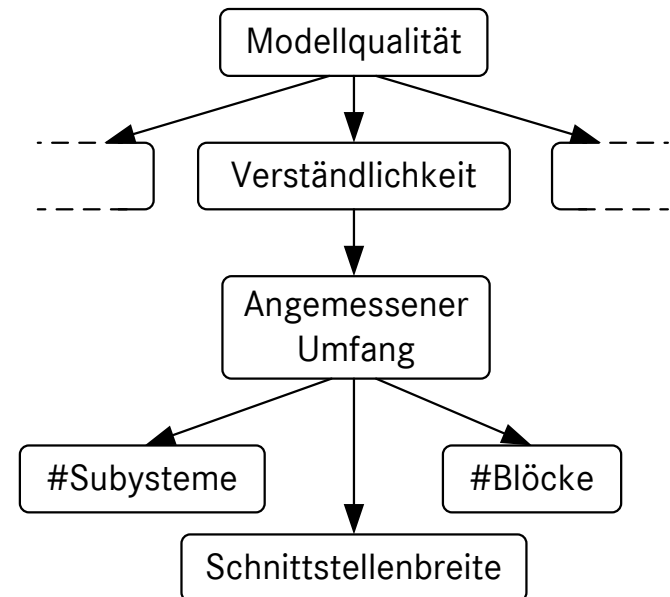
Auszug aus dem instanziierten Qualitätsmodell

Qualitätsmodell (3/3)



Darstellung der Hierarchie eines Simulink-Modells als Tree Map

- Auswirkung des Kriteriums „Angemessener Umfang“ auf die „Verständlichkeit“



Auszug aus dem instanziierten Qualitätsmodell

Modellmetriken

Wie kann die Erfüllung der Kriterien gemessen werden?

- Modellmetriken messen auf den Modellen
- ihre Aufgabe ist, die Erfüllung von Qualitätskriterien zu messen
- Eigenschaften der Metriken
 - direkt/indirekt
 - primitiv/berechnet
 - Minimal- und Maximalwert
 - Typ (atomar, akkumulierend, arithmetisches Mittel)
 - Priorität (1 = normal, 2 = hoch)
- ein Kriterium ist erfüllt, wenn alle Messwerte seiner Metriken innerhalb ihrer Minimal- und Maximalwerte liegen

Modellqualitätsbewertung

Wie wird die Modellqualitätsbewertung gebildet?

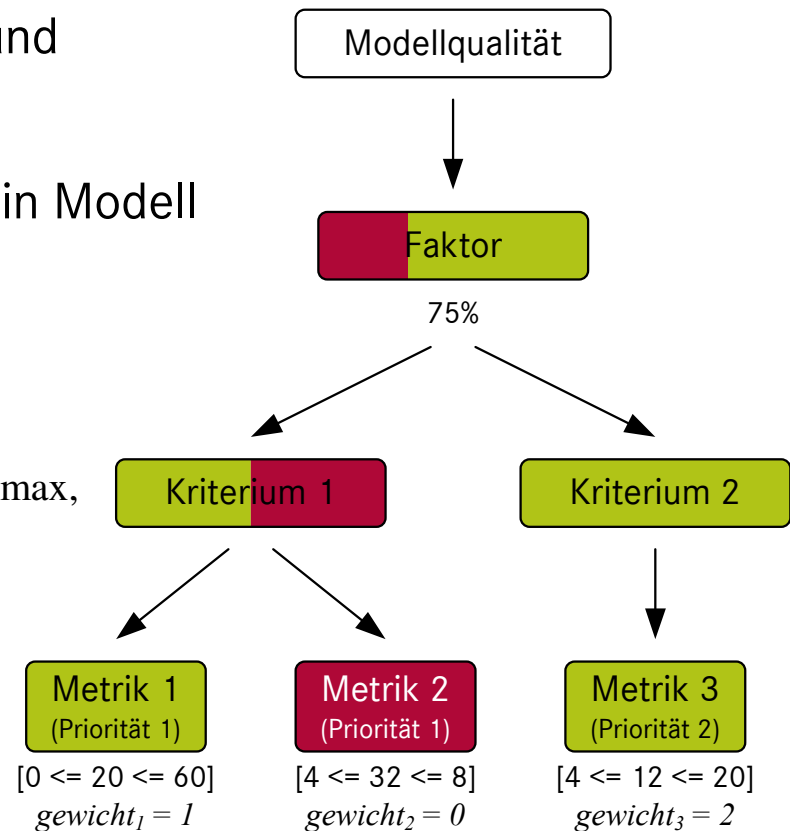
1. Problemgröße definieren, um Minimal- und Maximalwerte der Metriken anzupassen
 - z.B. Anzahl der Anforderungen für ein Modell
2. Berechnung pro Faktor

- Gewichten der Metriken:

$$gewicht_i = \begin{cases} priorität_i, & \text{falls } \min \leq \text{messwert} \leq \max, \\ 0, & \text{sonst.} \end{cases}$$

- Erfüllungsgrad des Faktors:

$$erfüllungsgrad = \frac{\sum_i gewicht_i}{\sum_i priorität_i}$$



→ die Qualitätsbewertung ist ein n-Tupel (n ist die Anzahl der Faktoren)

Handlungsempfehlungen

Wie kann die Qualität der Modelle verbessert werden?

- Handlungsempfehlungen für nicht erfüllte Kriterien
- können Messwerte als Orientierungshilfe verwenden (z.B. Subsysteme mit vielen Ausreißern sind interessant), um Empfehlung präziser zu machen
- bilden eine Wissensdatenbank zur qualitätsorientierten Modellierung

Framework

Wie kann das Qualitätsmodell angepasst werden?

- das Qualitätsmodell wird durch verschiedene Qualitätsaspekte gebildet

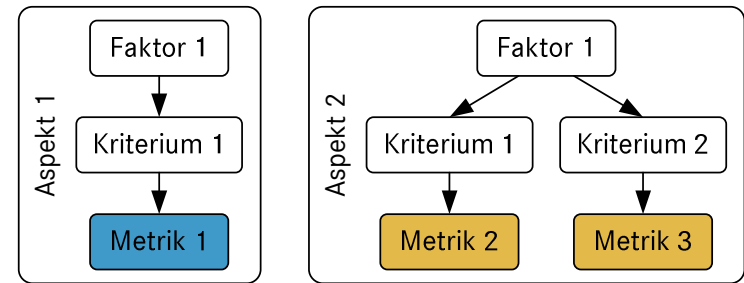
- die einzelnen Aspekte sind an- und abschaltbar

- neue Aspekte können hinzugefügt werden

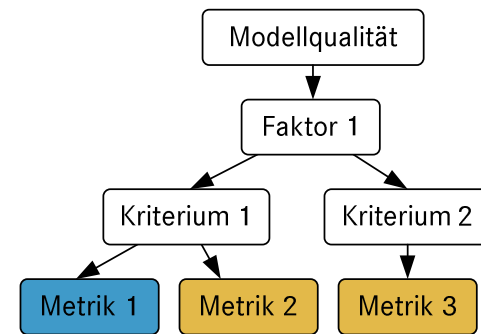
→ projektspezifische Erweiterbarkeit

- momentan vorhandene Aspekte:

- Anforderungen
- C-Code
- Modellierungsrichtlinien
- Targetlink
- Architektur
- Komplexität
- Simulink
- Test



II



Schematische Darstellung der Vereinigung zweier Aspekte

Inhalt

Problemstellung

Lösungsansatz

Prototyp

Zusammenfassung und Ausblick

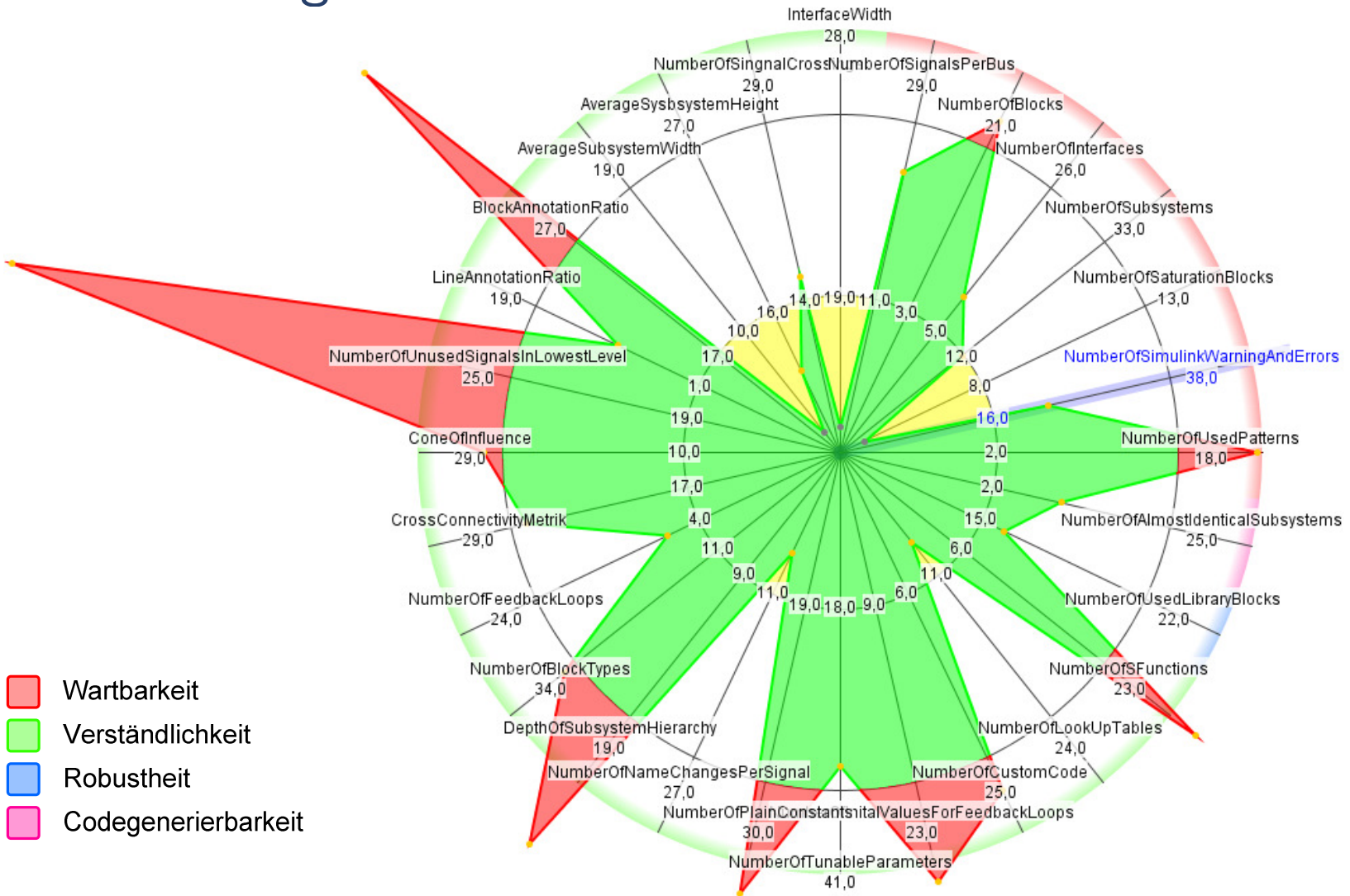
Prototyp (1/3)

Wie kann der Ansatz evaluiert werden?

- mittels eines in Java geschriebenen Prototyps
- verwendet zum Parsen der Simulink-Modelldateien einen Parser der TU München
- erste Metriken sind bereits implementiert
- implementiert bereits das vorgestellte Framework
 - verwaltet die Qualitätsaspekte in XML-Dateien
 - vereinigt ausgewählte Aspekte zu einem Qualitätsmodell, das zur Qualitätsbewertung verwendet wird

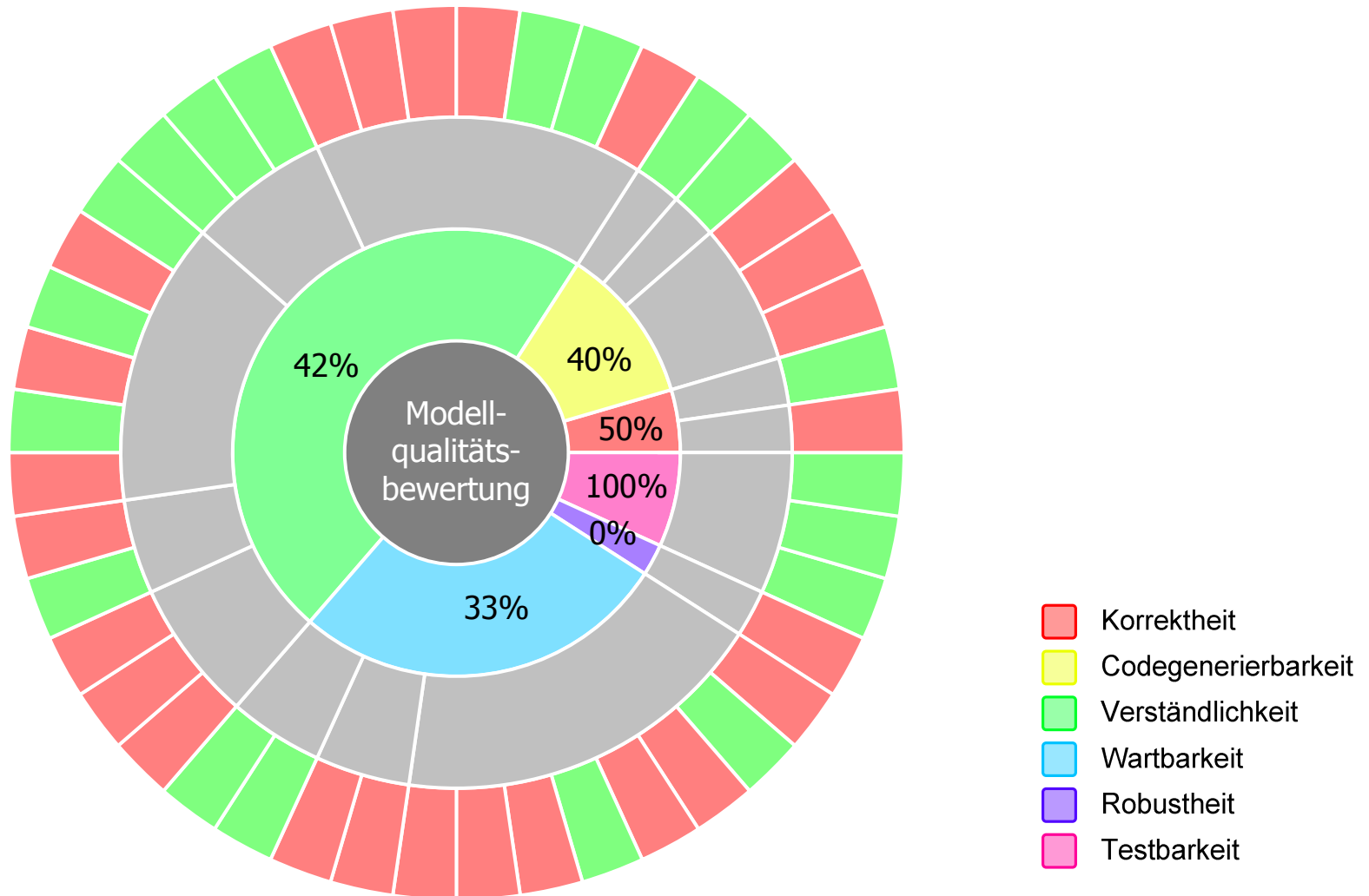
Prototyp (2/3)

Visualisierung der Messwerte



Prototyp (3/3)

Visualisierung der Modellqualitätsbewertung



Inhalt

Problemstellung

Lösungsansatz

Prototyp

Zusammenfassung und Ausblick

Zusammenfassung

- Ansatz zur automatisierten Qualitätsbewertung anhand von Qualitätskriterien
- Strukturierung der Kriterien in einem Qualitätsmodell
- Erfüllung der Kriterien wird mit Hilfe von Modellmetriken gemessen
- Handlungsempfehlungen für nicht erfüllte Kriterien
- Framework zur Anpassbarkeit und Erweiterbarkeit
- Prototyp zur Evaluierung

Ausblick

Nächste Schritte:

1. Vervollständigung des Prototyps (Implementierung der Metriken)
2. Evaluierung des bestehenden Qualitätsmodells
3. Anpassung/Erweiterung des Qualitätsmodells um neue Faktoren und Kriterien

Vielen Dank!