

Anforderungsanalyse und Wissensrepräsentation

*Theoretische Grundlagen und praktische Nutzung
im Bereich Lichtelektronik der Hella KGaA*

Autoren:

Roland Kapeller, Bernd van Vugt (Hood Group), Stefan Krause (Hella)



Roland Kapeller
Unternehmensberatung



Übersicht

Teil I) **Systematische Anforderungsanalyse: Die Einzelanforderung**

- Anforderung und Aussage
- Kategorisierung als Basis der Anforderungsformulierung
- Schema der Anforderungsgegenstände

Teil II) **Wissensrepräsentation: Der Anforderungskatalog**

- Anforderungskatalog und Systemspezifikation
- Wissen als Basis der Kommunikation von Kunde und Lieferant
- Schema der Beschreibungsebenen

Teil III) **Praktische Umsetzung: Der Workflow im Produktkontext**

- Pflichtenheft, Produktdefinition und Kundenanforderung
- Anforderungsmodellierung als Basis der Entwicklung
- Schema der Anforderungsdokumente

Fachlich-Technisches Umfeld: Lichtelektronikentwicklung Hella

- Die Lichtelektronik entwickelt innovative neue Lichtverteilungen wie

Kurvenlicht

Autobahnlicht

Stadtlicht

Abbiegelicht (ABBL)

mit hohem Potential für Alleinstellungen durch kundenspezifische Merkmale.

- Die Entwicklung setzt diese Funktionen mit einem hohem Anteil an **standardisierten Komponenten und Modulen (HW/SW/CO)** in ein mechatronisches System um.



Adaptive Frontlighting System



Adaptive
Frontlighting
System

Die Autoren



Dipl.-Ing. (FH) Dipl.-Wirt.-Ing. (FH) Roland Kapeller,
freier Unternehmensberater. Seit 10 Jahren in Systemanalyse,
Anforderungsanalyse/ -management und Prozeßanalyse/ -
design und in unterschiedlichen Branchen (Finanzdienstleister,
Automobilzulieferindustrie, Behörden) tätig.
E-Mail: Roland.Kapeller@Kapeller-UB.de



Dipl.-Inf. (FH) Bernd van Vugt, HOOD Group
Consultant im RM&E, *Schwerpunkt Automotive & Prozesse.*
E-Mail: Bernd.van.Vugt@HOOD-Group.com



Dipl.-Ing. Stefan Krause, Hella KGaA Hueck & Co.
System- & Softwareanalytiker im Bereich Entwicklung Licht-
Elektronik Systeme. *Arbeitsschwerpunkt:* Definition und
Anwendung von Methoden und Prozessen im Requirements-
und System-Engineering.
E-Mail: Stefan.Krause@hella.de

Was ist eine Anforderung?

- Eine Anforderung ist eine **Aussage** über (mind.) einen Gegenstand.
- Eine Anforderung wird stets aus einer bestimmten **Perspektive** gegeben.
- **Kommunikation** → Qualitätskriterium der **Verstehbarkeit**
⇒ Eindeutigkeit, Verständlichkeit, Einheit der einzelnen Aussage



Anforderungssteller



Anforderungsanalytiker

Was ist ein Anforderungskatalog?

- Der Anforderungskatalog ist die **Definition eines Zielsystems**.
- Es umfaßt die **Einzelperspektiven** der Anforderungssteller (Stakeholder).
- Die „Multiperspektivität“ der Anforderungen *konstituiert* den allgemeinsten Gegenstand *Zielsystem*.
- **Zielvorstellung** → Qualitätskriterium der **Konsistenz** und **Lückenlosigkeit**
⇒ Widerspruchsfreiheit und Vollständigkeit eines Anforderungskataloges



Die Anforderung als Aussage

- Eine Aussage verknüpft Gegenstände miteinander,
- macht Aussagen über die Eigenschaften eines Gegenstands,
- sein Zeitverhalten etc.

- Sprachliche Form: *Subjekt Prädikat [Objekt]*
- Beispiel: Das *Kurvenlicht* folgt dem Straßenverlauf.



- Lassen sich Aussagen *typisieren*?
- Gibt es komplexere *standardisierbare* Formen?

Die Kategorientafel des Aristoteles

- Altgriechisch „**kategoría**“ bedeutet „Aussage“.
- Eine Aussage sagt immer etwas über einen Gegenstand aus (Prädikation).
- Damit werden auch die Möglichkeiten der **Veränderung** greifbar.

1) Substantia (Wesenheit)

2) **Quantitas** (Menge):

3) **Qualitas** (Beschaffenheit)

4) **Relatio** (Beziehung)

5) **Ubi** (Ort)

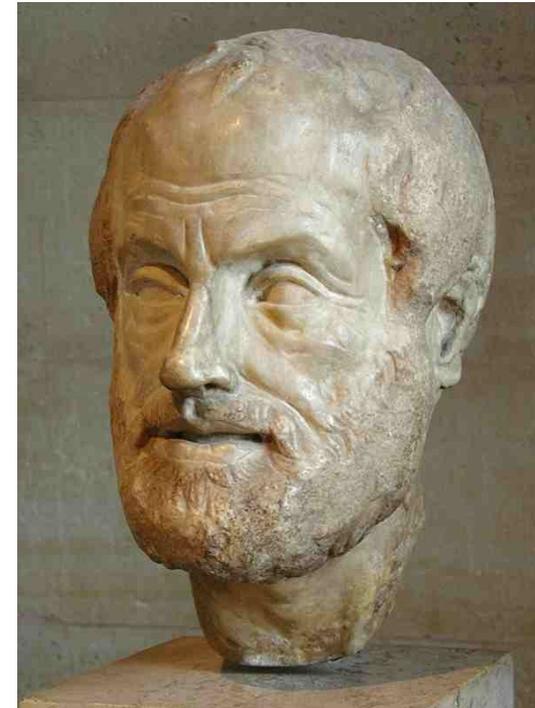
6) **Quando** (Zeitpunkt)

7) **Situs** (Lage)

8) **Habitus** (Besitz)

9) **Actio** (Wirken)

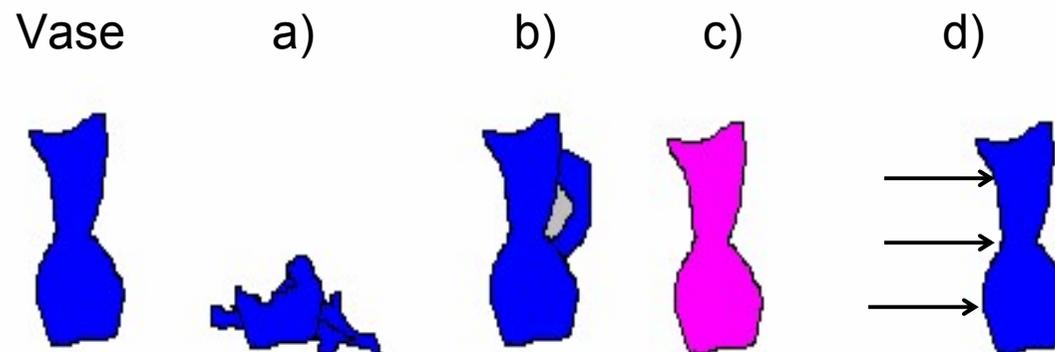
10) **Passio** (Widerfahren)



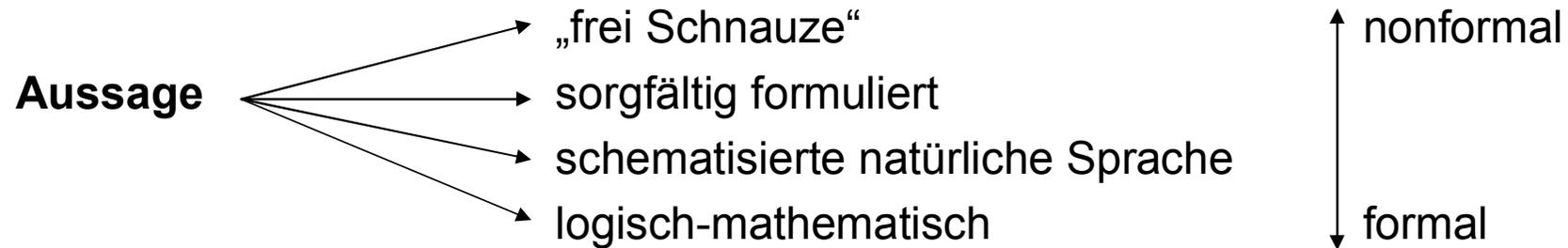
Veränderung von Gegenständen (nach Aristoteles)

- Altgriechisch „**kínesis**“ bedeutet „Bewegung“ und „Veränderung“.

- Substanzielle** Veränderung (Werden und Vergehen)
- Quantitative** Veränderung (Vermehrung und Verminderung)
- Qualitative** Veränderung (Änderung von Eigenschaften)
- Orts**veränderung



Form der Anforderung



Vorbestimmte **Aussageweisen** ↔ Vorbestimmte **Elementklassen** (Typen)

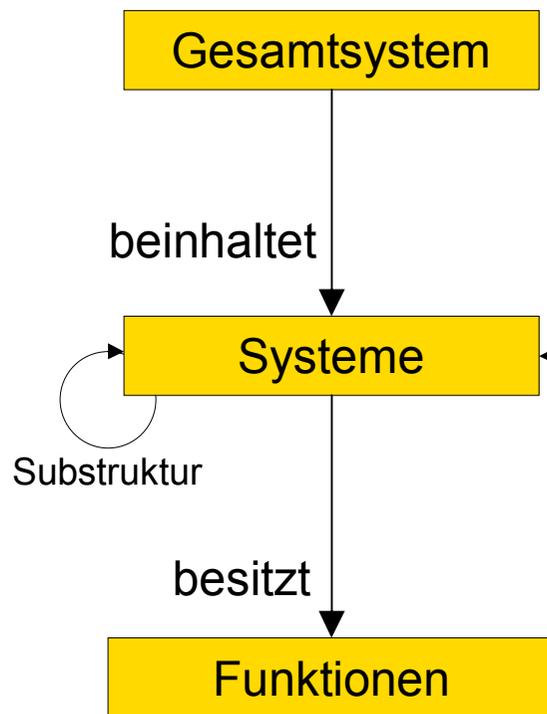
Von der Umgangssprache zur strukturierten Sprache

- „X“ wird aktiviert, sobald „y“.
- Funktion „X“ startet, wenn **Bedingung** „y“ erfüllt.
- Funktion „<Funktion>“ startet, wenn **Bedingung** „<Bedingung>“ erfüllt.
mit <Bedingung> ::= <Größe> (größer/kleiner/gleich/...) <Wert>

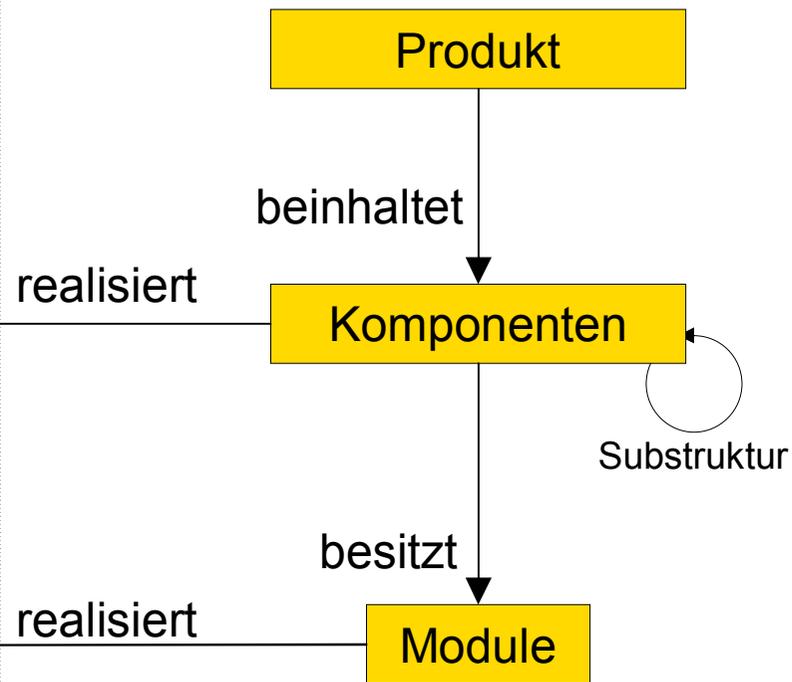
Ontologie der Anforderungsgegenstände der Systemanalyse

Ein **Metamodell** auf Basis von Elementklassen dient der Analyse, der Wissensrepräsentation, dem Systementwurf und der Verifikation.

Verhaltenssicht: Anforderungen an die Funktionalität



Struktursicht: Anforderungen an den technischen Entwurf



Anforderung als Gegenstand

- Gemäß der Kategorientafel können auch **Aussagen über Anforderungen** (als Gegenstand) gemacht werden.
- Insbesondere **Veränderungen** von Typ, Menge, Eigenschaft und Ort von Anforderungen müssen **kontrolliert** geschehen (→ **Änderungsmanagement**).

- 1) Anforderungs-**Typ** (zur Klassifikation von Anforderungen)
- 2) **Menge** von Anforderungen
- 3) **Eigenschaften** der Anforderung
- 4) **Beziehung** zu anderen Anforderungen
- 5) **Ort** der Anforderung (Speicherung)
- 6) Zeitliche **Gültigkeit** der Anforderung / History
- 7) **Form** der Anforderung (Dokumentation)
- 8) **Inhalt** der Anforderung
- 9) **Wirkung** der Anforderung
- 10) **Verwendung** der Anforderung

Konsequenzen für Anforderungsanalyse und -management

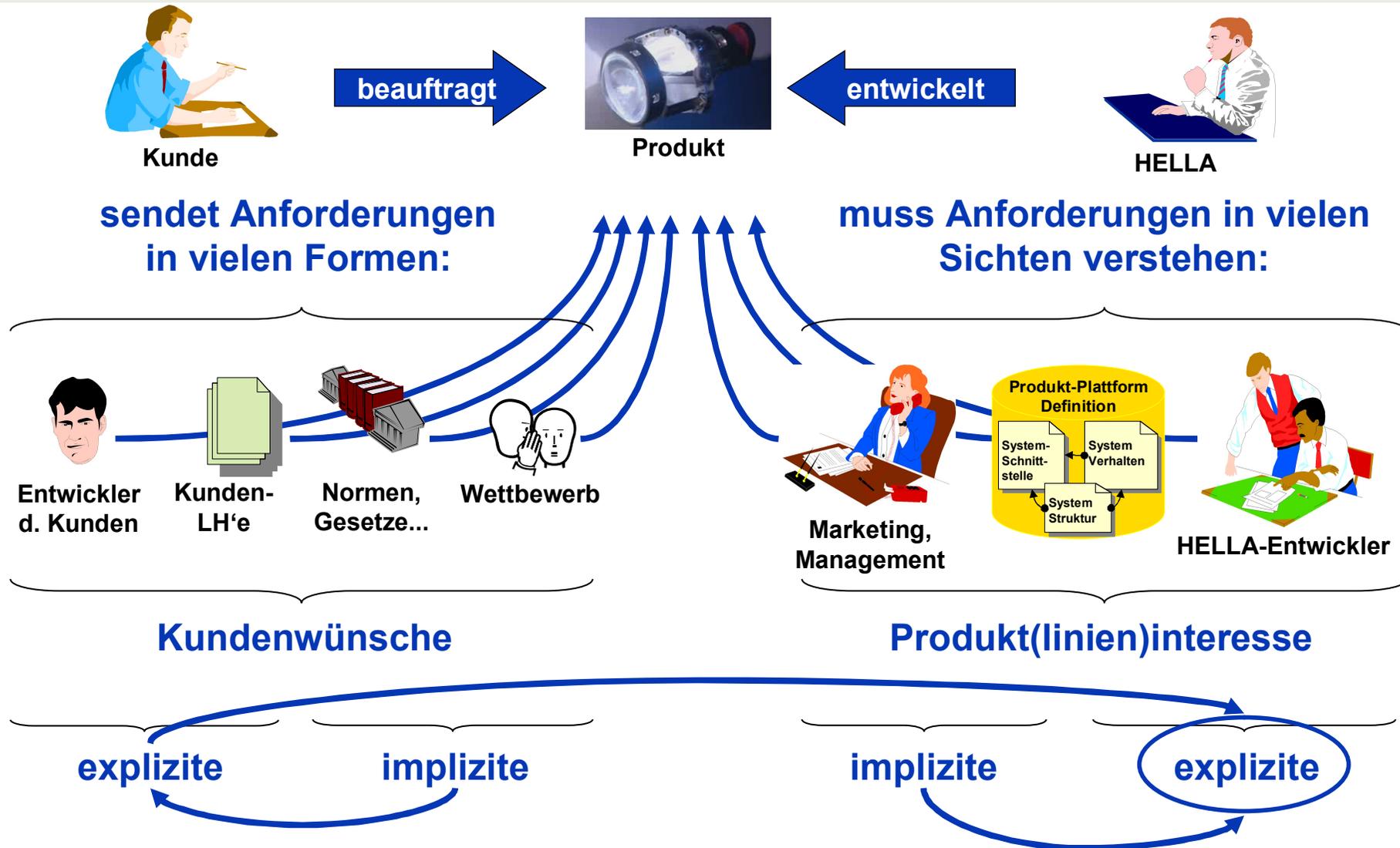
Anforderungsanalyse

- Am wichtigsten ist, eine **einheitliche Zielvorstellung** zu gewinnen.
- Diese Zielvorstellung wird nicht vorgefunden, sondern **konstituiert**.
- Ein Modell ist ein **System** von Aussagen unter Maßgabe gewisser Regeln.
- Dies ist zur Sicherung von Konsistenz und Lückenlosigkeit erforderlich.
- Die **Systemspezifikation** muß also ein **Modell** sein:
 - Beschreibung der Zielvorstellung (Projekt-Zusammenhang)
 - **Wissen** über den Zielbereich (Produkt-Zusammenhang)
- „Entwicklung“ bedeutet **Transformation** eines Begriffs-Systems in ein faktisches System.

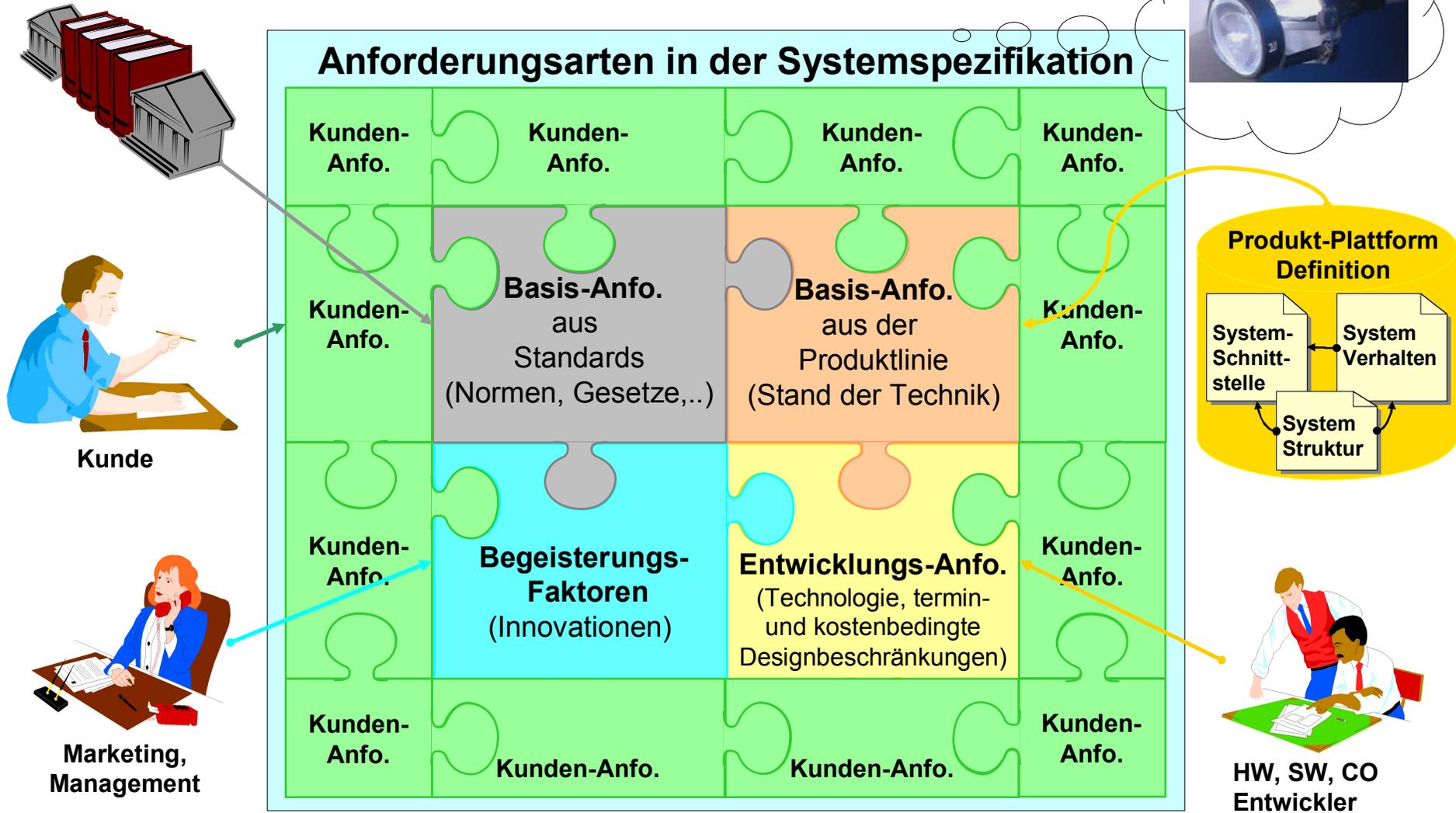
Anforderungsmanagement

- **Sichten** (Perspektiven) des PM, der Entwickler, der Stakeholder.
- Jede Anforderung ist in ihrer **Historie** zu berücksichtigen.
- Bezug auf den Gegenstand der Anforderung einschließlich seiner **Konstitution** im Rahmen der Anforderungsanalyse.

Welches Wissen ist zu „explizitieren“?



Welche Aussagen zum Produkt ~~gibt es~~ in der Systemspezifikation? benötigt man



Was ist eigentlich „Wissen“, und wie wird es repräsentierbar?

- Die **Wissensrepräsentation** bedient sich essentieller Anleihen des **systemischen Wissensmanagements**.
- **Wissensarbeit** ist das begriffliche Eingeständnis, daß Wissen nicht mittels einmaliger Übertragung o.ä. erlernbar ist, sondern...
 - (1) Kontinuierlich **revidiert** werden muss,
 - (2) permanent als **verbesserungsfähig** angesehen werden muss,
 - (3) prinzipiell nicht als Wahrheit, sondern als **Ressource** betrachtet wird und
 - (4) untrennbar mit **Nichtwissen** gekoppelt ist.
- Das, was an Wissen **gemeinschaftlich erarbeitet, formuliert** und (strukturiert) **abgelegt** werden kann, kann auch **explizit repräsentiert** werden.
- Ablage und Wiedergabe von Wissen benötigt die Kultivierung von **personalem** wie **organisationalem** Wissen.

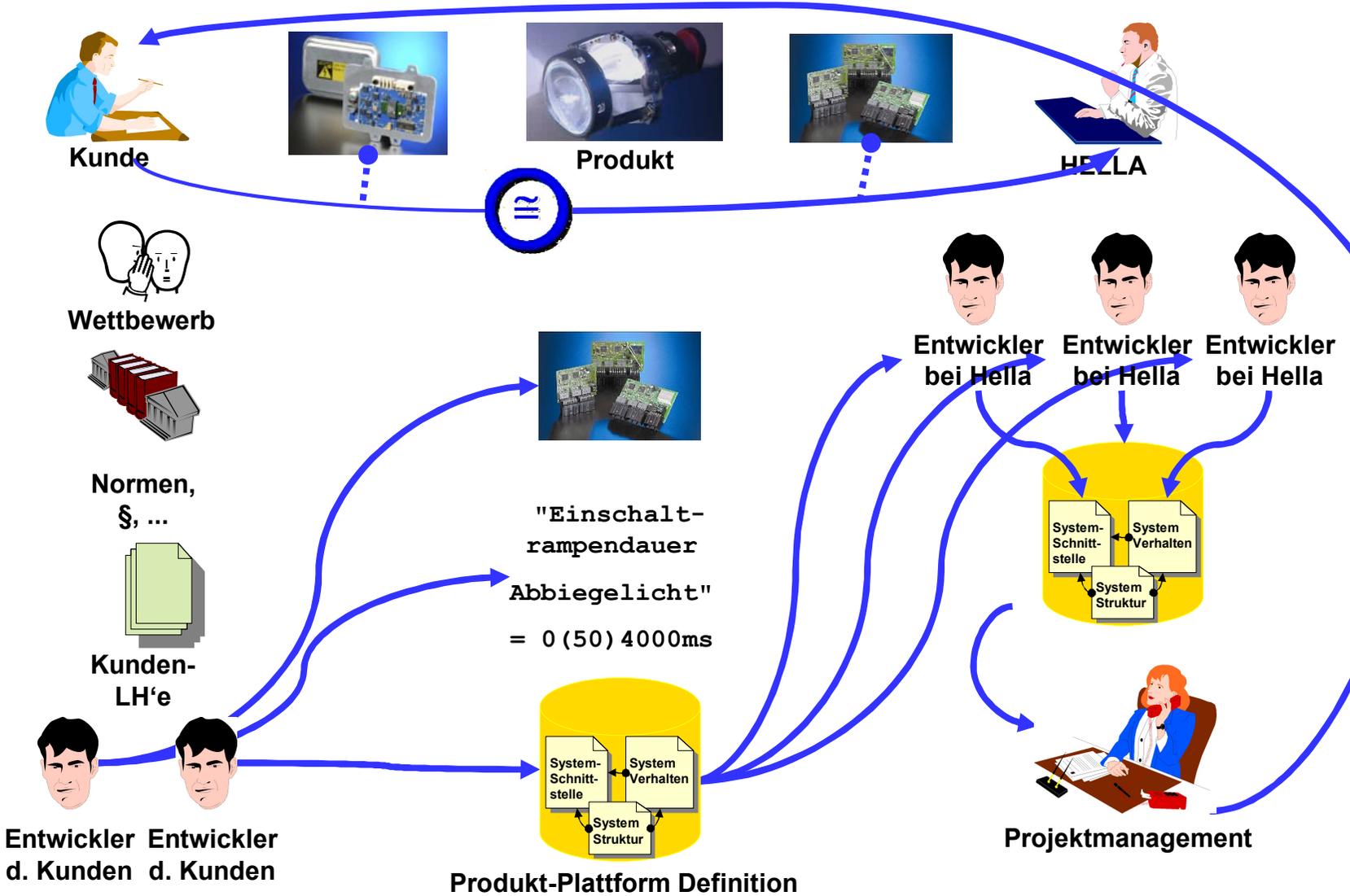
Welche Fragen sind vor der Repräsentation von Wissen zu klären?

1. Ist eine **gemeinsame Sprache**, bspw. Deutsch, verwendet worden?
2. Liegt ein möglichst großer, **gemeinsamer Wortschatz** vor?
3. Hat das verwendete Vokabular bei allen Beteiligten die **gleiche Bedeutung**?
4. Liegt eine **gemeinsame Wissensbasis** zugrunde?
5. Können alle Beteiligten aus der gemeinsamen Wissensbasis die von ihnen benötigten Informationen **bekommen**?
6. Können alle Beteiligten in die gemeinsame Wissensbasis die von ihnen erzeugten Informationen **einstellen**?

Fazit:

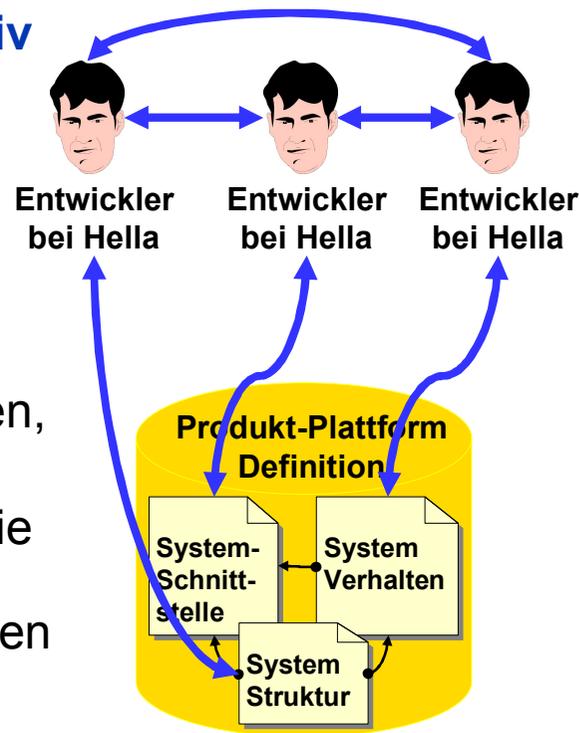
1. Damit sich alle ein gemeinsames „Bild“ vom Zielsystem machen können, ist ein **kooperatives, zielgerichtetes Vorgehen** bei allen Beteiligten nötig.
2. Dabei muß es Ziel sein, diese Wissensbasis bereits **kooperativ** zu **schaffen**.

Wissen als Basis der Kommunikation von Kunde und Lieferant



Vorbedingungen für personales Wissen und sein Wert:

- Wissensarbeit:
Die benötigten Informationen wurden im Vorfeld (a) **systemrelativ erarbeitet** und (b) **auch so abgelegt**,
- Wissensarbeit:
Alle Entwickler haben **Zugriff auf** und **Verständnis von** den abgelegten Informationen,
- Wissensarbeit:
Alle Entwickler haben die Möglichkeit, die kodifizierten Informationen zu **verändern**.
- Wissensarbeit:
Alle Entwickler arbeiten **gemeinschaftlich** – grundsätzlich!



1. Resultat:
Der Anforderungskatalog als **Definition des Zielsystems**,
2. Resultat:
Die einzelnen Sichten aller Entwickler **bilden gemeinsam** das Zielsystem,
3. Resultat:
Die Entwickler können **Art** und **Inhalt** der Wissensrepräsentation ihren Bedürfnissen anpassen (lassen).
4. Resultat:
Der Anforderungskatalog umfasst die **Einzelperspektiven aller Stakeholder**.

Repräsentation organisationalen Wissens

Produkt-Plattform Definition

The image displays three overlapping screenshots of a software tool used for product platform definition. The tool's interface includes a menu bar (File, Edit, View, Insert, Link, Analysis, Table, Tools, User, Add-Ons), a toolbar, and a main content area. The screenshots show a hierarchical tree structure on the left and detailed functional descriptions on the right.

Top-left screenshot: Shows a hierarchical tree structure under the heading "2 Steuergeräte". The tree includes sub-items like "2.1 Allgemein", "2.2 AFS-Systeme", and "2.2.1.3.1 Abbiegelicht". The "Abbiegelicht" item is selected, and its parameters are listed below it: "ABBL Einschalttrampendauer" and "ABBL Ausschalttrampendauer".

Top-right screenshot: Shows a detailed view of the "ABBL-Lampe ansteuern" function. It includes a "PROCESSING" section with sub-items like "5.1.6.2.1 Funktionalität" and "5.1.6.2.4 Parametrisierung". A text box describes the system's use of configuration parameters for the "ABBL dimmen" function, listing "ABBL Einschalttrampendauer" and "ABBL Ausschalttrampendauer".

Bottom screenshot: Shows a detailed view of the "ABBL Schreib/Lese-Speicherung" function. It includes a "Zweck" section with a text box describing the function's purpose: "Die Funktion 'ABBL Schreib/Lese-Speicherung' dient der permanenten Speicherung von Größen der Abbiegelicht-Funktionalität und läßt deren Änderung zu." Below this, it states: "Das HW-Modul 'EEPROM' realisiert die Funktion 'ABBL Schreib/Lese-Speicherung'."

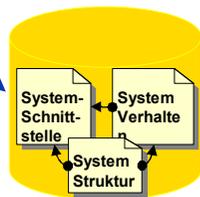
Beispiel einer Kundenanforderung (O-Ton Lastenheft)

- **Dimmung Abbiegelicht**
- ...
- Die Rampendauer ist als EEPROM-Parameter (0...4000 ms; in 50ms Schritten) vorzusehen, und für das Ein- und Ausschalten getrennt einstellbar.

Der **Kontext** der Anforderung ist die **Dimmfunktion des Abbiegelichts**:

In ihr finden sich **drei Gegenstände**, auf die sich die Anforderung richtet:

- 1) **Funktion** "ABBL dimmen"
- 2) **Parameter** "ABBL Einschaltrampendauer" = 0(50)4000ms,
Parameter "ABBL Ausschaltrampendauer" = 0(50)4000ms,
- 3) **HW-Modul** "EEPROM"



"Einschaltrampendauer Abbiegelicht" = 0(50)4000ms
„Ausschaltrampendauer Abbiegelicht" = 0(50)4000ms



Verhaltensbeschreibung

The screenshot shows a software application window with a menu bar (File, Edit, View, Insert, Link, Analysis, Table, Tools, User, Add-Ons, Innovator, kitchen, Help) and a toolbar. The main area is divided into two panes. The left pane shows a hierarchical tree structure:

- 1 Symbole, Synonyme, Abkürzungen
- 2 Gesamtsysteme
- 3 Systeme
- 4 Sensorik-Systeme
- 5 Steuerungs-Systeme
 - 5.1 Steuerung Abbiegelicht
 - 5.1.1 Funktionalität
 - 5.1.2 Systemschnittstelle
 - 5.1.3 Systemeigenschaften
 - 5.1.4 Systemzustände
 - 5.1.5 Parametrisierung
 - 5.1.6 Subsysteme
 - 5.1.7 Funktionen
 - 5.1.7.1 ABBL Schreib/Lese-Speicherung
 - 5.1.7.1.1 Zweck
 - 5.1.7.1.2 Funktionsschnittstelle
- 6 Aktorik-Systeme

The right pane displays the content for the selected node '5.1.7.1.1 Zweck':

5.1.6.2.4 Parametrisierung

Das System "ABBL-Lampe ansteuern" verwendet folgende Konfigurationsparameter zur Unterstützung der Funktion "ABBL dimmen":

- "ABBL Einschaltrampendauer",
- "ABBL Ausschaltrampendauer".

5.1.7 Funktionen

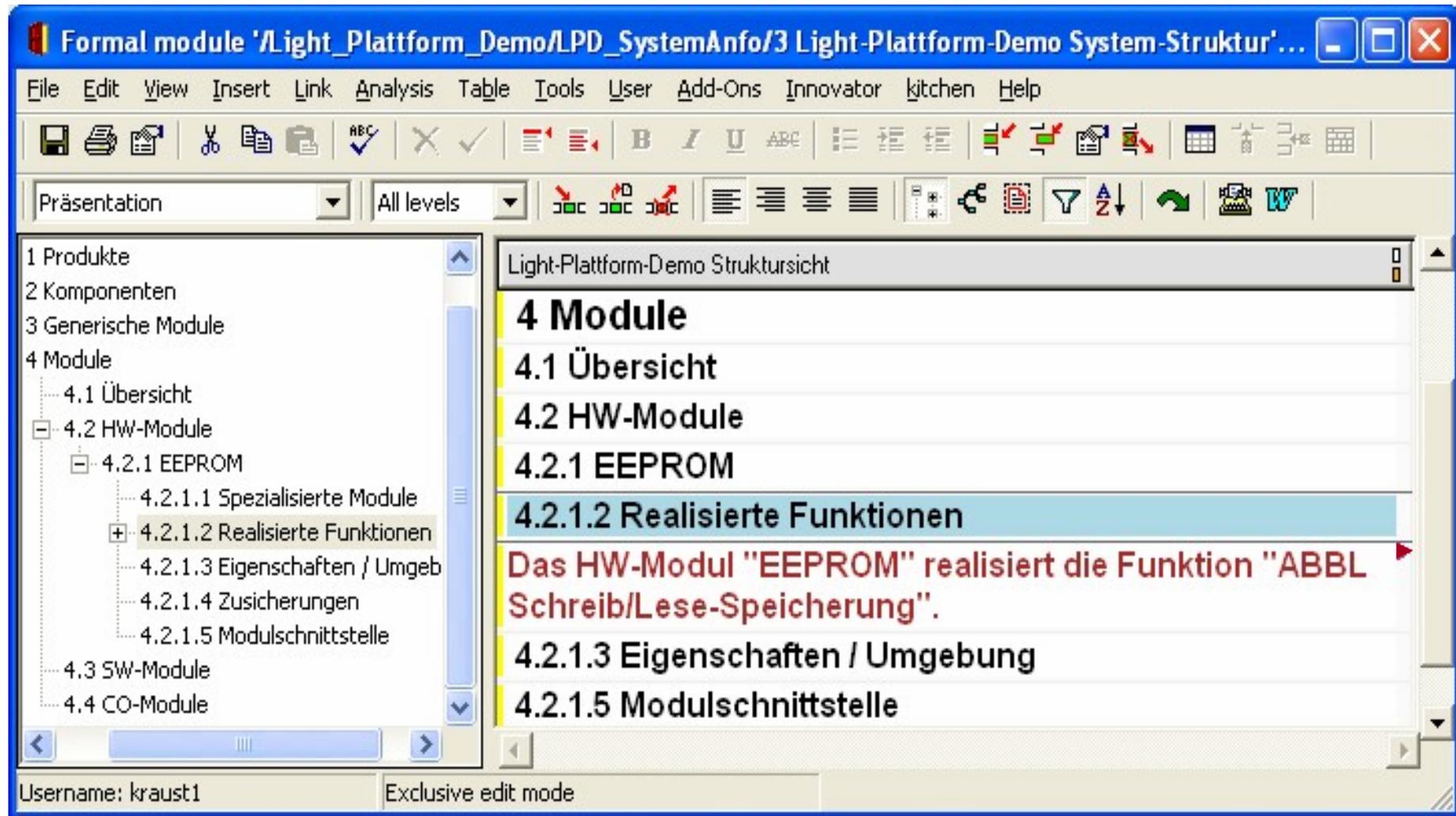
5.1.7.1 ABBL Schreib/Lese-Speicherung

5.1.7.1.1 Zweck

Die Funktion "ABBL Schreib/Lese-Speicherung" dient der permanenten Speicherung von Größen der Abbiegelicht-Funktionalität und läßt deren Änderung zu.

At the bottom of the window, a status bar reads: "Delete the currently selected object."

Strukturbeschreibung



System-Schnittstellenbeschreibung

Formal module '/Light_Plattform_Demo/LPD_SystemAnfo/1 Light-Plattform-Demo System-Schnittstellen'...

File Edit View Insert Link Analysis Table Tools User Add-Ons Innovator kitchen Help

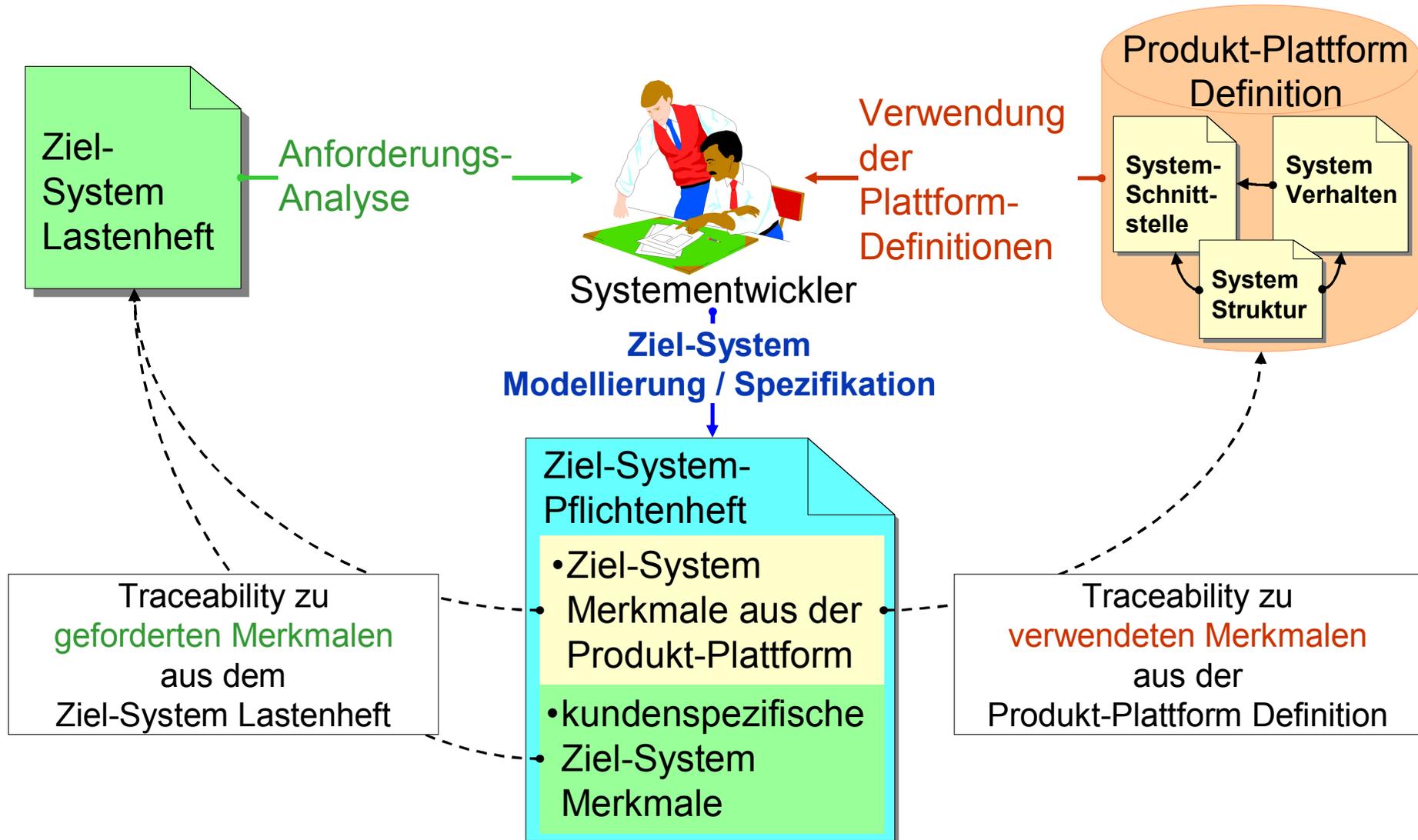
Präsentation All levels

1 Sensoren
2 Steuergeräte
 + 2.1 Allgemein
 - 2.2 AFS-Systeme
 - 2.2.1 Informationen
 + 2.2.1.1 Input
 - 2.2.1.2 Output
 - 2.2.1.3 Parameter
 - 2.2.1.3.1 Abbiegelicht
 "ABBL Einschalttrampendauer"
 "ABBL Ausschalttrampendauer"
 - 2.2.1.3.2 Lichtsystem 2
 - 2.2.1.4 Technische Limits
 + 2.2.2 Elektrische Größen
 + 2.2.3 Physikalische Größen
3 Aktoren

Light-Plattform-Demo System-Schnittstellen	Min. Wert:	Schrittweite:	Max. Wert:	Einheit
2 Steuergeräte				
2.1 Allgemein				
2.2 AFS-Systeme				
2.2.1 Informationen				
2.2.1.1 Input				
2.2.1.2 Output				
2.2.1.3 Parameter				
2.2.1.3.1 Abbiegelicht				
"ABBL <u>E</u> inschaltrampendauer"	0	50	4000	ms
"ABBL <u>A</u> usschaltrampendauer"	0	50	4000	ms

Username: kraust1 Exclusive edit mode

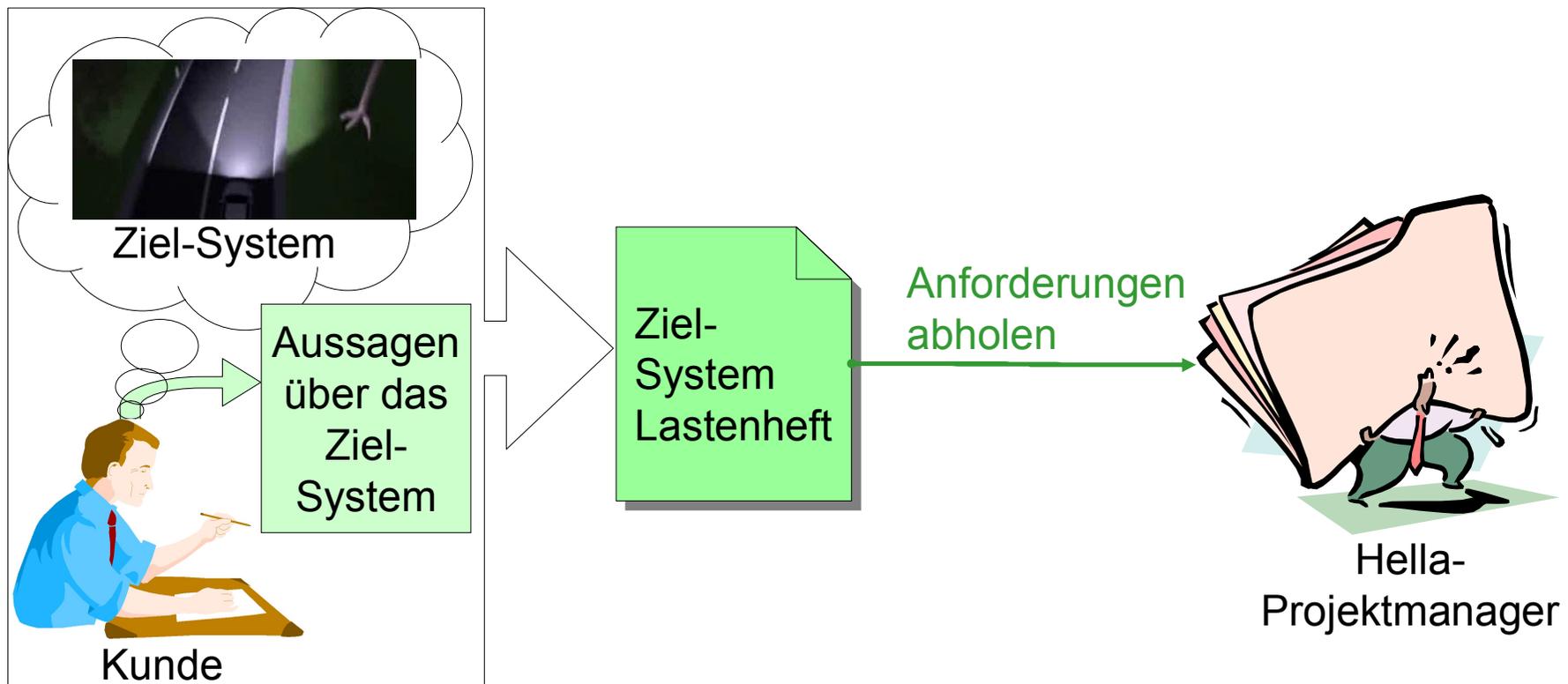
Pflichtenheft, Produktdefinition und Kundenanforderung



Anforderungsmodellierung als Basis der Entwicklung

1. Schritt: Der Hella-Projektmanager holt das LH vom Kunden ab.

- Die Entwickler seitens des Kunden repräsentieren und kommunizieren ihr Wissen über das Ziel-System durch formulierte Anforderungen.
- Dies erfolgt in der Regel in Lastenheften (LH).

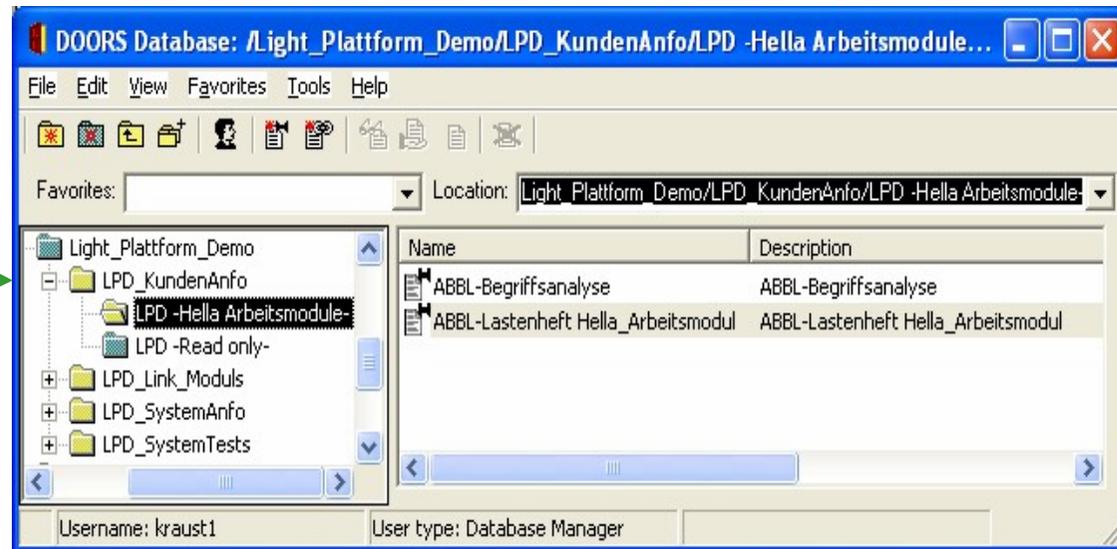


Anforderungsmodellierung als Basis der Entwicklung

2. Schritt: Der Requirements-Engineer (RE) importiert das LH in die DOORS-DB.



Import



- Der Aufwand und die Qualität des Importes hängt vom Format des LH ab:
 - DOORS-Modul (*.dma / *.dpa) : optimal
 - DOORS eXchange-Datei oder RIF : optimal
 - Word-Datei (*.doc / *.rtf) : Automatischer Import möglich
 - Adobe Acrobat-Datei (*.pdf) : praktisch nicht akzeptabel
 - Papier : nicht akzeptabel

Anforderungsmodellierung als Basis der Entwicklung

3. Schritt: Der RE **identifiziert** die zu erfüllenden Anforderungen und separiert sie bezüglich der unterschiedlichen Perspektiven in **Kategorien**:

- System-Funktion -> Verhaltenssicht
- HW, SW, Mechanik/Konstruktion -> Struktursicht
- und verschiedene sonstige -> hier nicht dargestellt

Formal module 'Light_Plattform_Demo/LPD_KundenAnfo/LPD -Hella Arbeitsmodule-/ABBL-Lastenheft Hella_Arbeitsmodul' ...

ID	UR_AnfoQuelle	ABBL-Lastenheft Hella_Arbeitsmodul	UR_AnfoKategorie	UR_AnfoStatus
UR_ABBL 1	keine Anfo	1 Dimmung Abbiegelicht	System (funktional)	-
UR_ABBL 2	Anfo Überschrift	Rampenfunktionen:	System (funktional)	-
UR_ABBL 3	Anfo	Einschalten: Sprung auf 30% ($U_{eff\ Sprung}$) weiter von 30% auf 100% ($U_{eff\ max}$) mit Rampenfunktion	System (funktional)	fertig zur Bewertung
UR_ABBL 4	Anfo	Ausschalten: von 100% ($U_{eff\ max}$) auf 0% mit Rampenfunktion	System (funktional)	fertig zur Bewertung
UR_ABBL 5	Anfo	Die Rampendauer ist als EEPROM-Parameter (0...4000 ms; in 50ms Schritten) vorzusehen, und für das Ein- und Ausschalten getrennt einstellbar.	System (funktional) HW (NF)	fertig zur Bewertung

Username: kraust1 Exclusive edit mode

Anforderungsmodellierung als Basis der Entwicklung

4. Schritt: Der System-Analytiker identifiziert in den Anforderungen Funktionen, Größen, Bedingungen, Komponenten, etc. im Attribut *Begriffe*.
5. Schritt: In einem strukturierten **Anforderungsmodell** wird die **Korrektheit**, **Konsistenz** und **Vollständigkeit** der Aussagen über die *Begriffe* verifiziert.

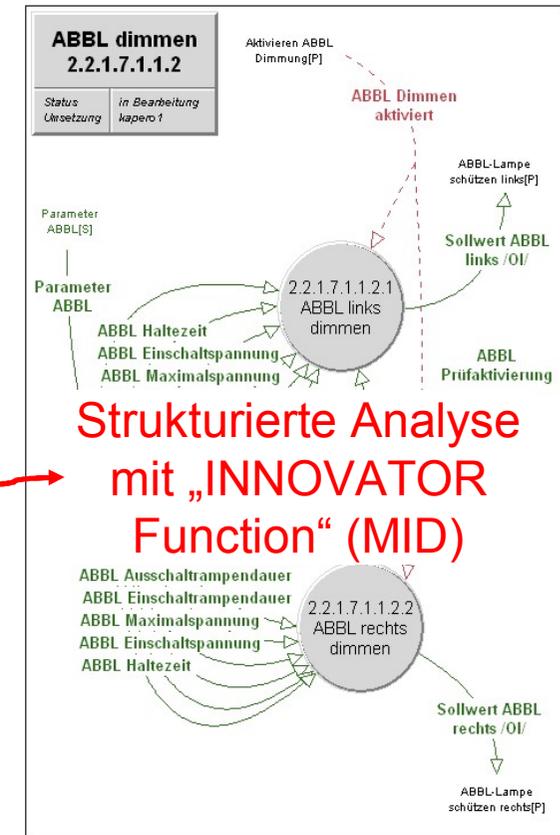
Formal module 'Light_Plattform_Demo/LPD_KundenAnfo/LPD_Hella Arbeitsmodule-/ABBL-Lastenheft...

File Edit View Insert Link Analysis Table Tools User Add-Ons Innovator kitchen Help

Verifikation_ProduktAnfo All levels

UR_AnfoKategorie	ABBL-Lastenheft Hella_Arbeitsmodul	UR_Begriffe
System (funktional)	1 Dimmung Abbiegelicht	F ABBL dimmen = Dimmung Abbiegelicht, // identifizierte Funktion und Synonymdefinition
System (funktional)	Rampenfunktionen:	
System (funktional)	Einschalten: Sprung auf 30% ($U_{eff\ sprung}$) weiter von 30% auf 100% ($U_{eff\ max}$) mit Rampenfunktion	B "ABBL Einschaltspannung" == 30% von "ABBL Maximalspannung", // identifizierte Bedingung
System (funktional)	Ausschalten: von 100% ($U_{eff\ max}$) auf 0% mit Rampenfunktion	G $U_{eff\ max_ABBL}$ = Max Sollwert ABBL, // Synonymdefinition einer Größe
System (funktional) HW (NF)	Die Rampendauer ist als EEPROM-Parameter (0...4000 ms; in 50ms Schritten) vorzusehen, und für das Ein- und Ausschalten getrennt einstellbar.	B "ABBL Einschalttrampendauer" == 0(50) 4000ms, B "ABBL Ausschalttrampendauer" == 0(50) 4000ms, K EEPROM, // identifizierte Komponente

Username: kraust1 Exclusive edit mode



Anforderungsmodellierung als Basis der Entwicklung

6. Schritt: Die Entwickler analysieren und kommentieren die Kunden-Anforderungen bezüglich ihrer Realisierbarkeit.

- Die Entwickler vergleichen dazu die Aussagen über das Ziel-System mit den entsprechenden Aussagen über das Plattform-Produkt.
- Durch den Abgleich der Anforderungen mit einer expliziten Produkt-Plattform Definition wird die Anforderungs-Bewertung nachvollziehbar und **unabhängig** vom Produkt-Wissen der beteiligten Personen!



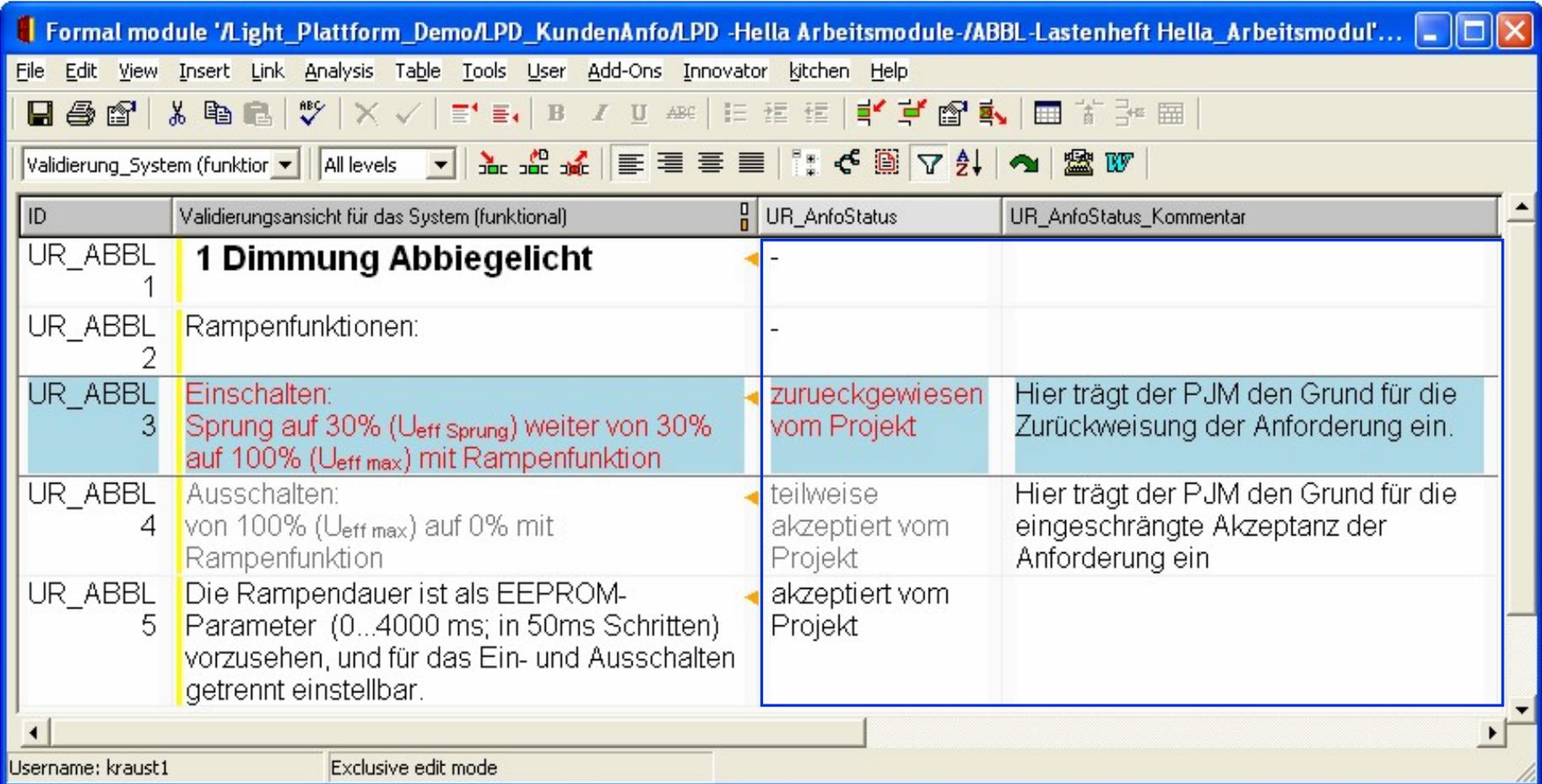
Anforderungsmodellierung als Basis der Entwicklung

6. Schritt: Die Entwickler analysieren und kommentieren die Kunden-Anforderungen bezüglich ihrer Realisierbarkeit.

ID	UR_AnfoQuelle	UR_AnfoKategorie	Analyseansicht für das System (funktional)	UR_AnfoKommentar
UR_ABBL 1	keine Anfo	System (funktional)	1 Dimmung Abbiegelicht	
UR_ABBL 2	Anfo Überschrift	System (funktional)	Rampenfunktionen:	
UR_ABBL 3	Anfo	System (funktional)	Einschalten: Sprung auf 30% ($U_{eff\ sprung}$) weiter von 30% auf 100% ($U_{eff\ max}$) mit Rampenfunktion	Hier kommentieren alle Entwickler erkannte Differenzen zur Produkt-Plattform
UR_ABBL 4	Anfo	System (funktional)	Ausschalten: von 100% ($U_{eff\ max}$) auf 0% mit Rampenfunktion	
UR_ABBL 5	Anfo	System (funktional) HW (NF)	Die Rampendauer ist als EEPROM- Parameter (0...4000 ms; in 50ms Schritten) vorzusehen, und für das Ein- und Ausschalten getrennt einstellbar.	

Anforderungsmodellierung als Basis der Entwicklung

7. Schritt: Der Projektmanager (PJM) **bewertet** (validiert) die Anforderungen anhand der Ergebnisse der Verifikation und der Analyse



ID	Validierungsansicht für das System (funktional)	UR_AnfoStatus	UR_AnfoStatus_Kommentar
UR_ABBL 1	1 Dimmung Abbiegelicht	-	
UR_ABBL 2	Rampenfunktionen:	-	
UR_ABBL 3	Einschalten: Sprung auf 30% (U_{eff} Sprung) weiter von 30% auf 100% ($U_{eff max}$) mit Rampenfunktion	zurueckgewiesen vom Projekt	Hier trägt der PJM den Grund für die Zurückweisung der Anforderung ein.
UR_ABBL 4	Ausschalten: von 100% ($U_{eff max}$) auf 0% mit Rampenfunktion	teilweise akzeptiert vom Projekt	Hier trägt der PJM den Grund für die eingeschränkte Akzeptanz der Anforderung ein
UR_ABBL 5	Die Rampendauer ist als EEPROM-Parameter (0...4000 ms; in 50ms Schritten) vorzusehen, und für das Ein- und Ausschalten getrennt einstellbar.	akzeptiert vom Projekt	

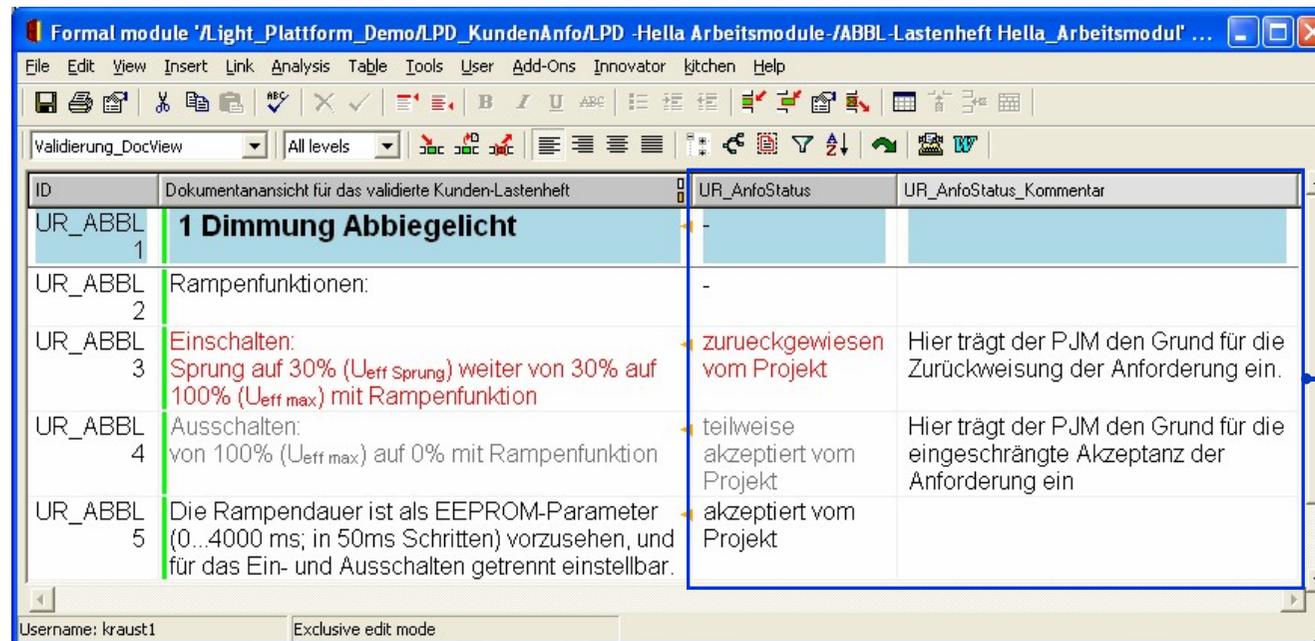
Username: kraust1 Exclusive edit mode

Anforderungsmodellierung als Basis der Entwicklung

7. *BL. Schritt:* Der RE archiviert das Ergebnis der Anforderungsanalyse in einer Baseline (BL).

8. *Schritt:* Der Projektmanager kommuniziert das Ergebnis der Anforderungsanalyse dem Kunden.

- Dazu wird das LH mit den Hella-Attributen *AnfoStatus* und *AnfoStatus_Kommentar* an den Kunden übermittelt.



The screenshot shows a software application window titled 'Formal module "/p></p>

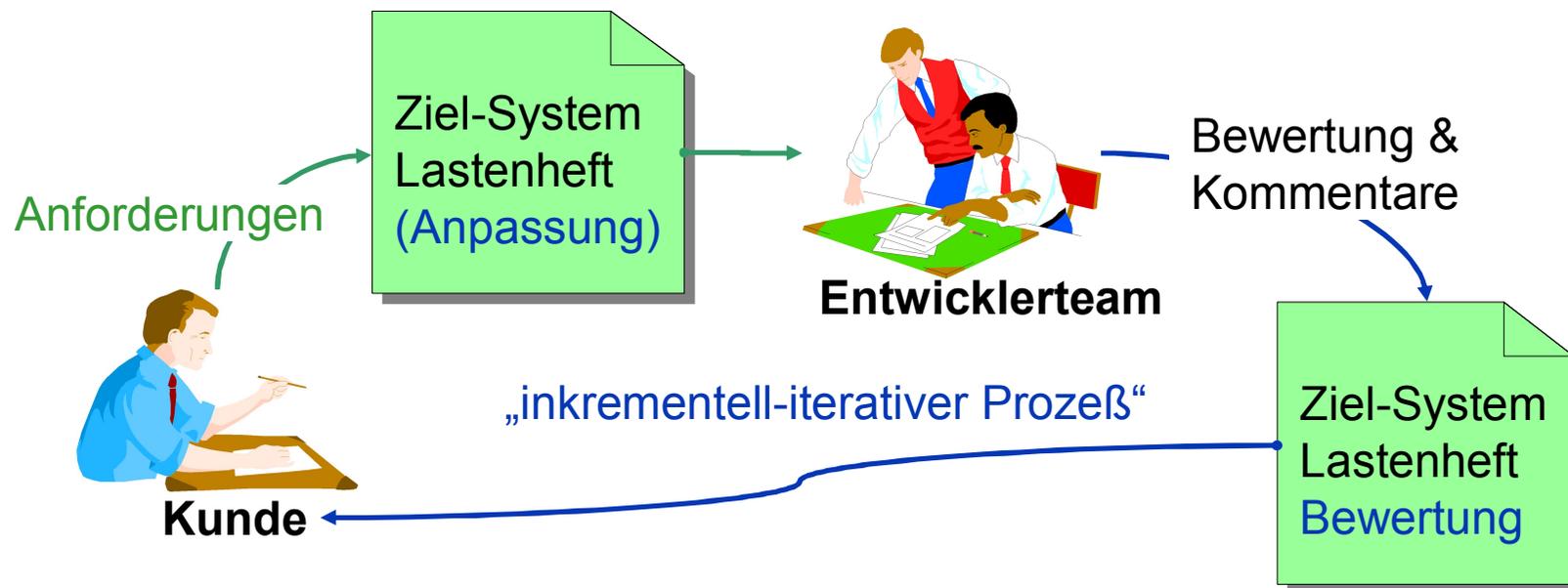


Kunde

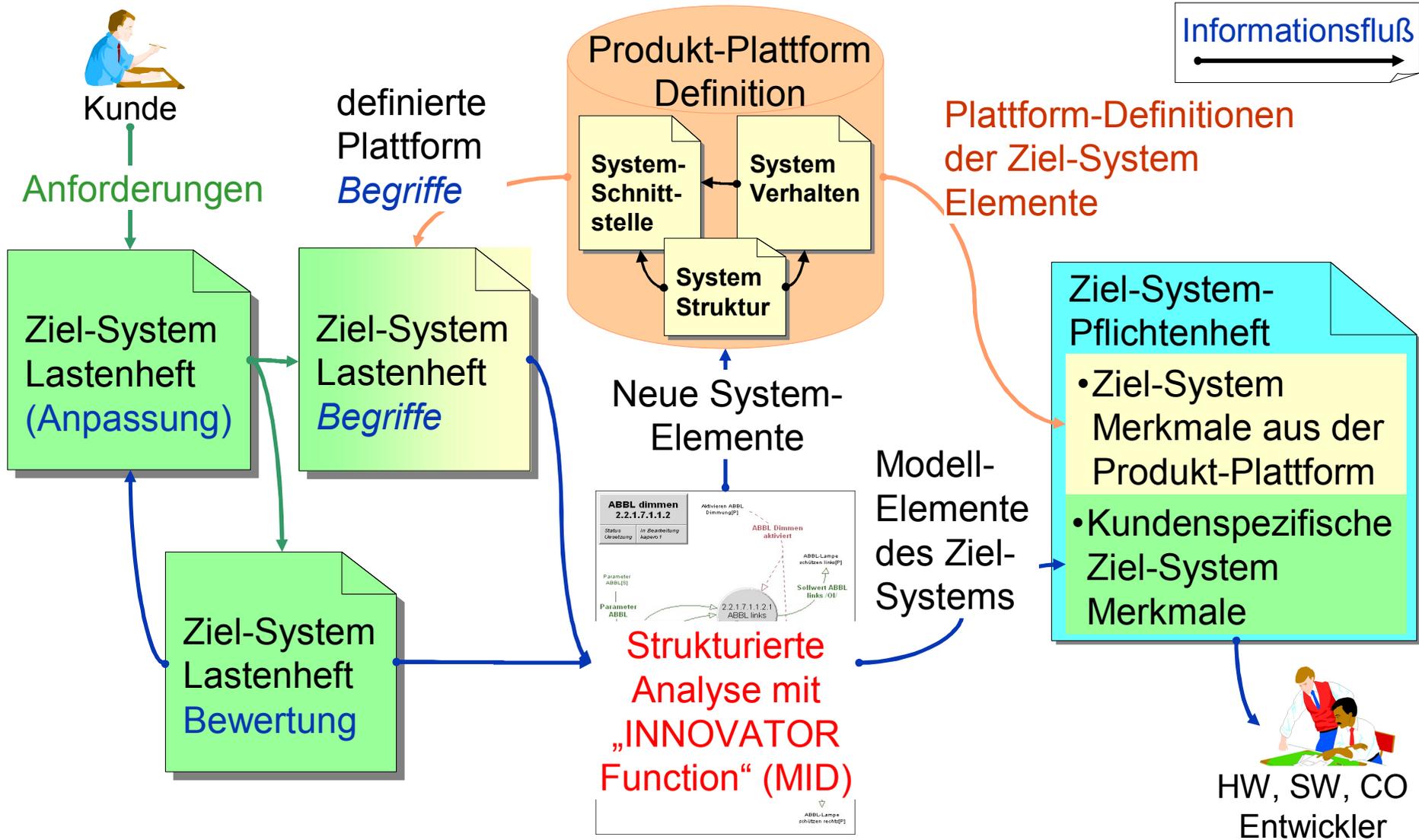
Anforderungsmodellierung als Basis der Entwicklung

9. Schritt: Der Kunde verarbeitet die Hella-Kommentare und den Hella-Status.

- Der Analyse-Workflow wird so oft durchlaufen, bis seitens des Zulieferers und des Kunden **eine gemeinsame Vorstellung** vom Ziel-System konstituiert ist.
- Dazu ist eine **geregelte Kommunikation** zwischen den System-Entwicklern seitens des Kunden und des Zulieferers zu etablieren.



Schema der Anforderungsdokumente



Fazit

- Permanente **Kommunikation** und direkte **Kooperation** zwischen Kunde und Lieferant führt zur **schnellen Konvergenz** der Zielvorstellungen.
- Eine formale **LH-Verifikation** ermöglicht **frühzeitige Problembeseitigung** und **beschleunigt** zugleich die **Erstellung der Systemspezifikation**.
- Eine **Plattformbeschreibung / Produktdefinition** als explizite **Wissensbasis** macht den Entwicklungsprozeß **personenunabhängig**, **steuerbar**, **wiederholbar** und **optimierbar** (→ SPICE).
- **Wiederverwendung** führt zur Steigerung von **Qualität** und **Effizienz** von **Produkt** und **Prozeß** (→ time-to-market).
- Notwendig ist eine **einheitliche Methode** (Workflow und Metamodell) für die **Analyse** und **Modellierung** von Anforderungen sowie für die **Produktdefinition**. Sie dient als Grundlage der **Spezifikationen**, der **formalen Prüfungen** und der Regeln für die **Traceability**.