

**IT-Symposium 2005**

**Motivation und Erfahrungsbericht bei  
der Einführung des V-Modells für die  
SW-Entwicklung**

Vortrag 3C07, 7.4.2005

Walter Kuhn  
KSG Kommunikations- und Sicherheitssysteme Ges.m.b.H.  
Computerstrasse 6, 1100 Wien  
w.kuhn@ksg.co.at

---

W.Kuhn, 7.4.2005  
1




Competence in Security and Communication

**Inhalt**

- Kurze Vorstellung der KSG
- Allgemeine Schwachstellen der Softwareentwicklung
- Möglichkeiten der Verbesserung
- Das V-Modell
- Zusammenfassung, Pros & Cons des V-Modells

---

W.Kuhn, 7.4.2005  
2



Competence in Security and Communication

## Hintergrund und Fokus der Firma KSG

1. Kommunikations- und Sicherheitssysteme GesmbH
2. Entwicklung von Hardware und Software für große Sicherheitssysteme, z.B. Gebäudeüberwachung von der Zutrittskontrolle bis zur Video-Überwachung und -Aufzeichnung  
Gebäudeüberwachung und Zutrittskontrolle unter **OpenVMS** (seit 1987)
3. Erfahrungen im EU Projekt SPIRE in den Austrian Research Centers Seibersdorf
4. Vor 2002 war die Entwicklung extern (Seibersdorf), ab dem Jahr 2002 Aufbau einer eigenen Entwicklungsabteilung in der KSG
5. Die Einführung eines Entwicklungsprozesses war notwendig
6. Heute:  
Core-Entwicklung von 20 Personen in Wien und Fürth  
Co-Entwicklung mit anderen Firmen und Forschungseinrichtungen  
Outsourcing

W.Kuhn, 7.4.2005  
3



Competence in Security and Communication

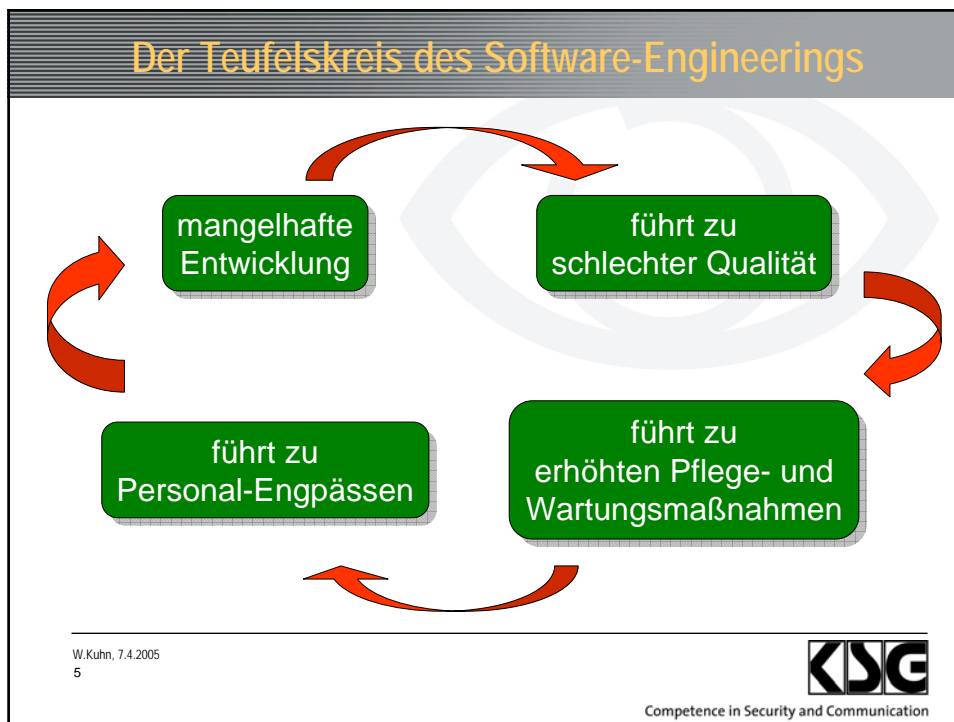
## Motivation für dieses Thema

1. Feststellung des Zustandes und der Reife sehr schwierig. Verlässliche Daten sind praktisch nicht zu erhalten. Man kann nur aus der täglichen Erfahrung versuchen, abzuschätzen oder gefühlsmäßig zu bewerten.
2. In all den Prozessen sind immer Menschen beteiligt. Programmierer haben Recht auf Qualität.

W.Kuhn, 7.4.2005  
4



Competence in Security and Communication



- ### Meist die gleichen Ausgangssituationen
1. Keine ausreichenden Anforderungsdokumente zur Spezifikation der geforderten Leistung
  2. Keine oder wenig entwicklungsbegleitende Dokumentation für die spätere Wartung
  3. Fehlende Durchgängigkeit im Entwicklungsprozess
  4. Abstimmungsprobleme
  5. Kein strukturiertes Vorgehen, viele unnötige Crash-Aktionen
  6. Arbeiten auf Zuruf
  7. Häufiges Wiederholen bereits durchgeführter Entwicklungsabschnitte
- W.Kuhn, 7.4.2005  
6
- 
- Competence in Security and Communication

## Das Ergebnis: es gibt Software...



W.Kuhn, 7.4.2005  
7

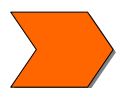


Competence in Security and Communication

## ... die wird „klassisch“ entwickelt



*"I'll go and find out what they need, and the rest of you start coding!"*

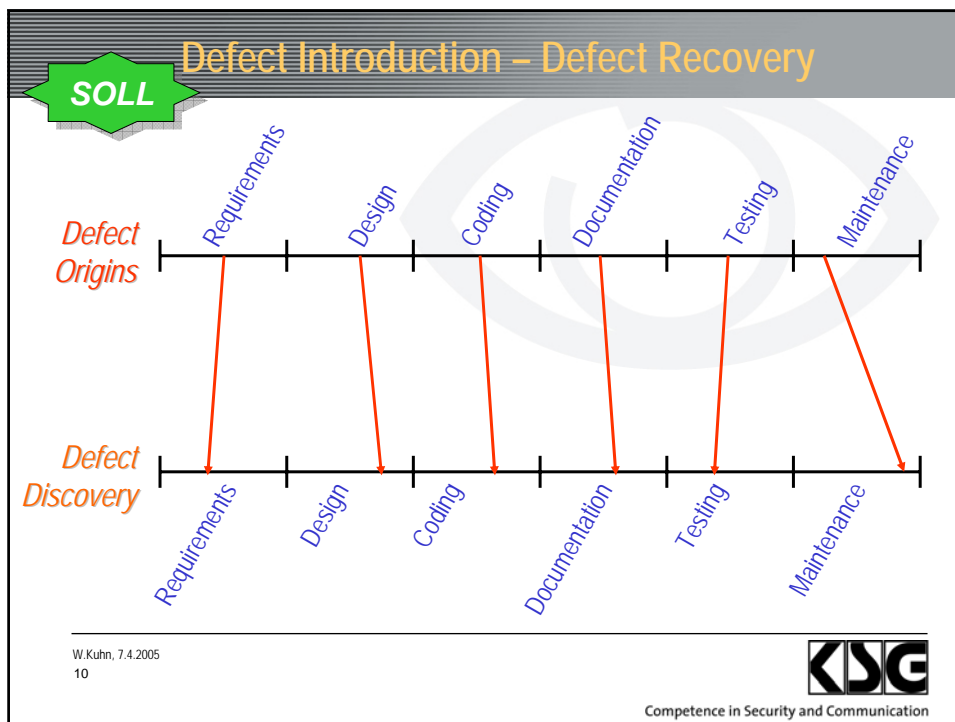
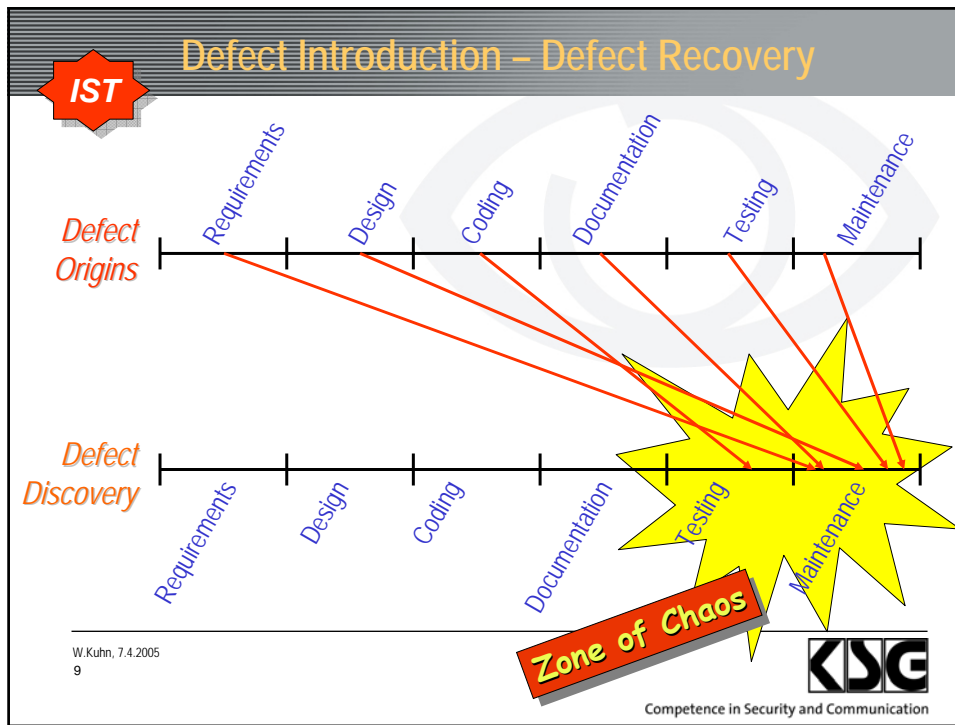


**VHIT** vom Hirn ins Terminal  
**ISW** ... is so wordn

W.Kuhn, 7.4.2005  
8



Competence in Security and Communication



## Die (falschen) Hoffnungen:

### Magern Sie ab, während Sie schlafen!

1. Es gibt einen neuen Trick, den Sie verpasst haben, und der die Produktivität immens steigert.
2. Andere erreichen Produktivitätssteigerungen von 100 Prozent oder 200 Prozent oder noch mehr.
3. Die Technologie entwickelt sich so rasch weiter, dass Sie den Anschluss verpassen.
4. Ein Wechsel der Programmiersprache wird riesige Vorteile bringen.
5. Sie automatisieren doch sonst alles; ist es nicht höchste Zeit, auch die Softwareentwicklung zu automatisieren?
6. Die Mitarbeiter arbeiten viel besser, wenn sie kräftig unter Druck stehen.

W.Kuhn, 7.4.2005  
11



Competence in Security and Communication

## Produktivitätsverbesserung

1. Es gibt keine Schnellschüsse zur Verbesserung der Produktivität.
2. Produktivitätsverbesserung erfordert langfristige Investitionen.
3. Alles, was kurzfristige Ergebnisse verspricht, ist vermutlich Scharlatanerie.

W.Kuhn, 7.4.2005  
12



Competence in Security and Communication

## Die Herangehensweise verändern

1. Die einzige Möglichkeit, die Leistung innerhalb eines Projekt substantiell zu verbessern, besteht darin, die Zeit für die Fehlersuche stark zu reduzieren.
2. Hochproduktive Projekte wenden anteilmäßig sehr viel weniger Zeit für das Debuggen auf.
3. Hochproduktive Projekte wenden anteilmäßig sehr viel mehr Zeit für den Entwurf auf.

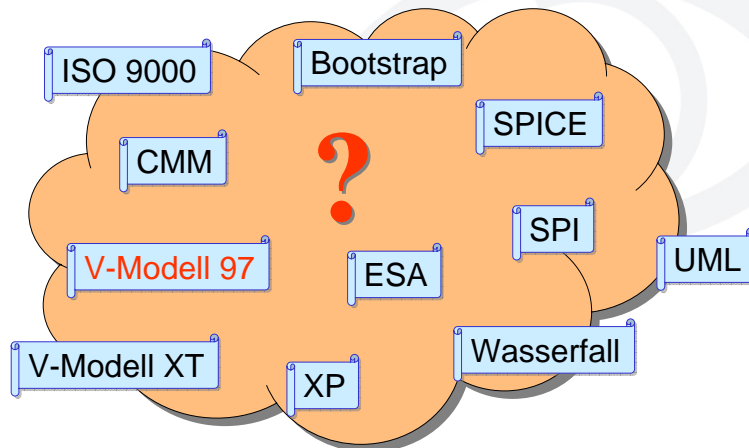
„Man kann nichts verbessern,  
wenn man nichts verändert.“

W.Kuhn, 7.4.2005  
13



Competence in Security and Communication

## Prozesse, Modellierung, Bewertung

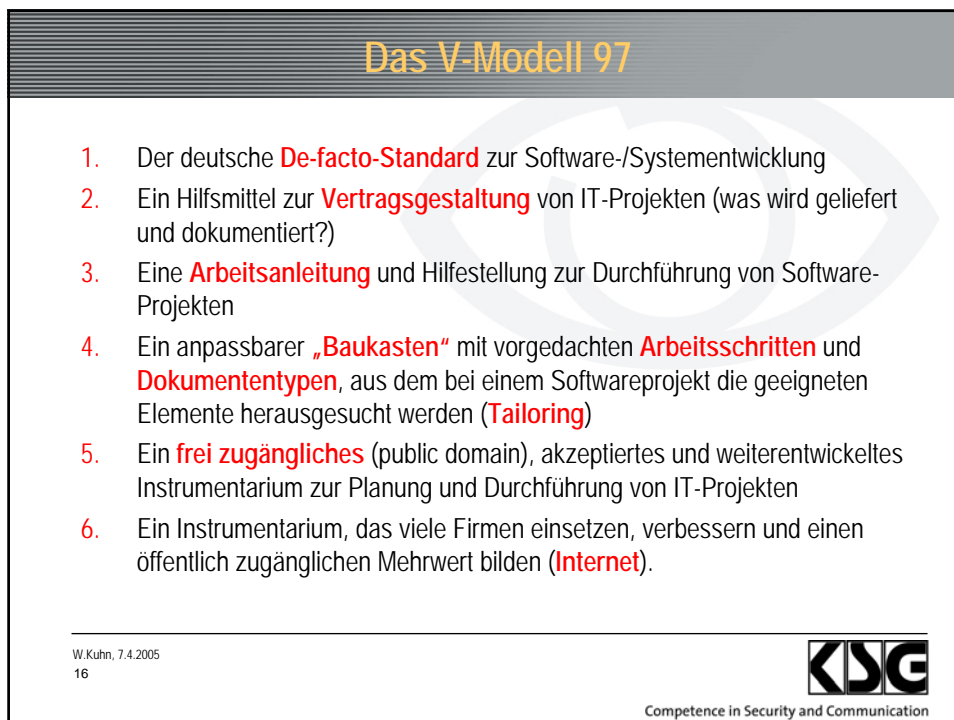
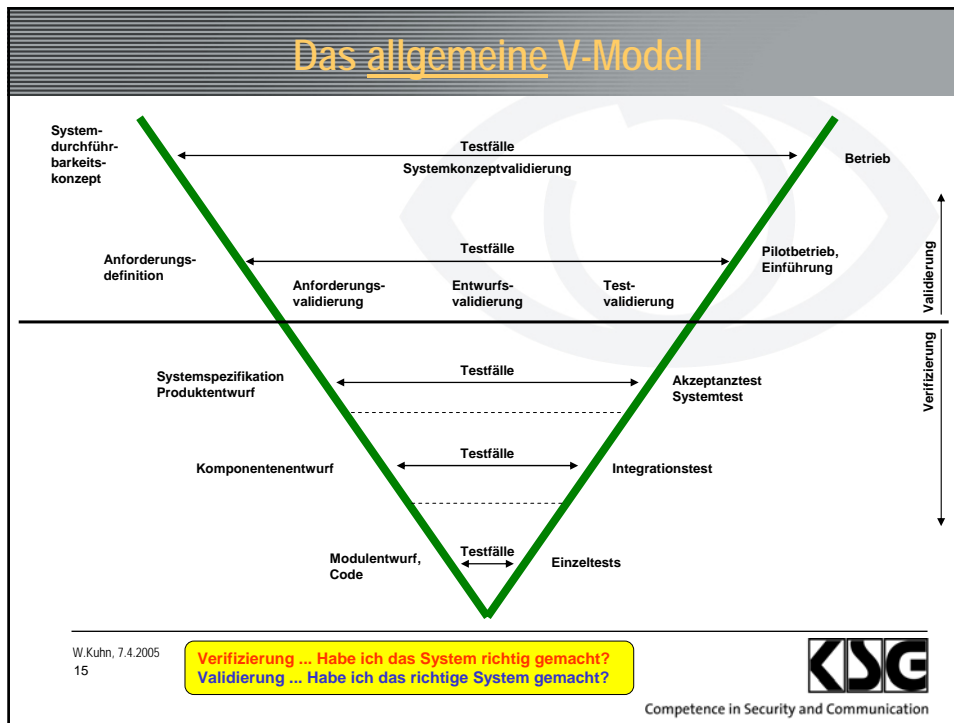


z.B. UML (Modelliersprache) + V-Modell (Entwicklungsprozess)

W.Kuhn, 7.4.2005  
14




Competence in Security and Communication






## Die zwei Aspekte des V-Modells


**1. Die IDEE**



**2. Der FORMALISMUS**



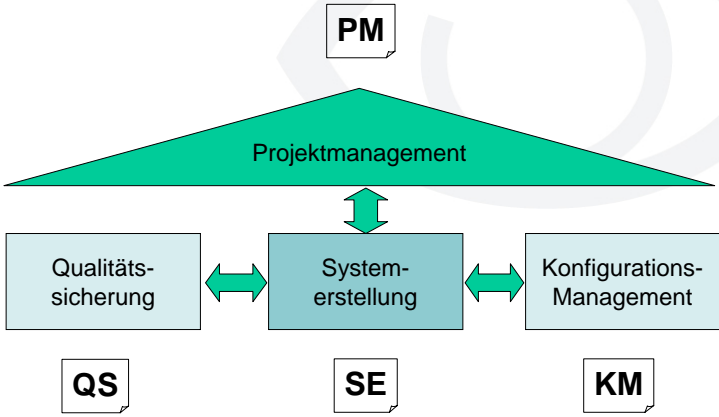
W.Kuhn, 7.4.2005  
17



Competence in Security and Communication


## Überblick: die vier Submodelle

<trockener einschub>

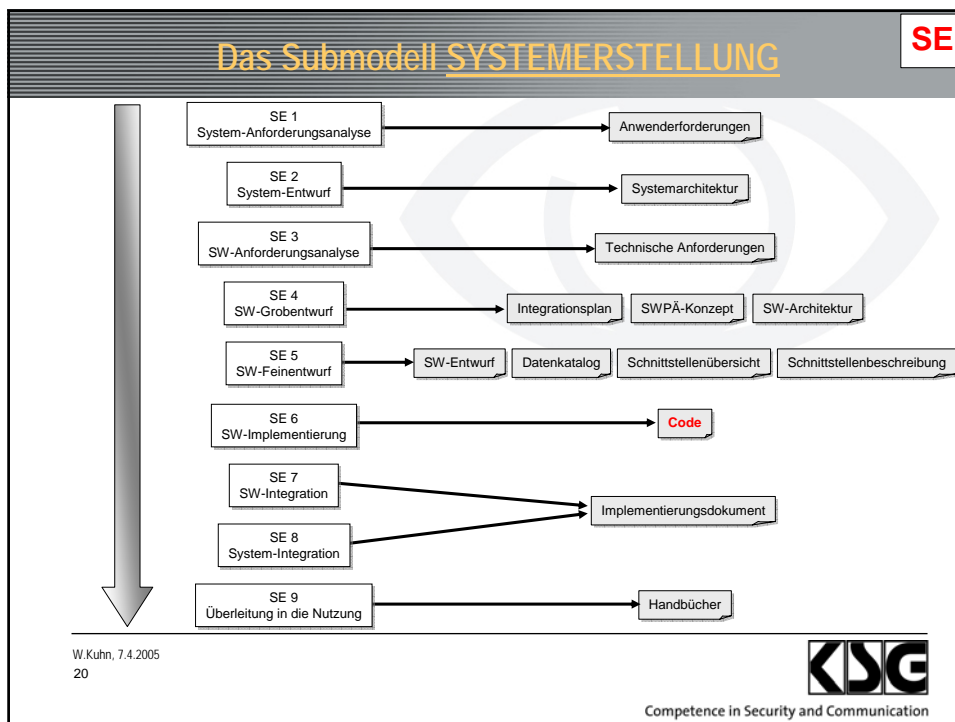
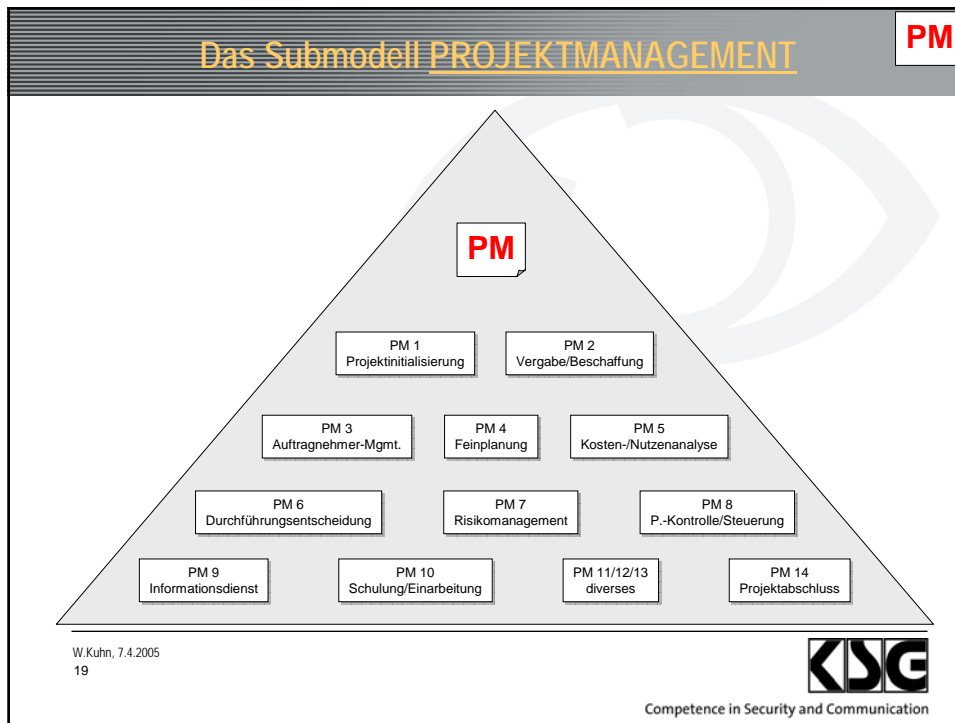


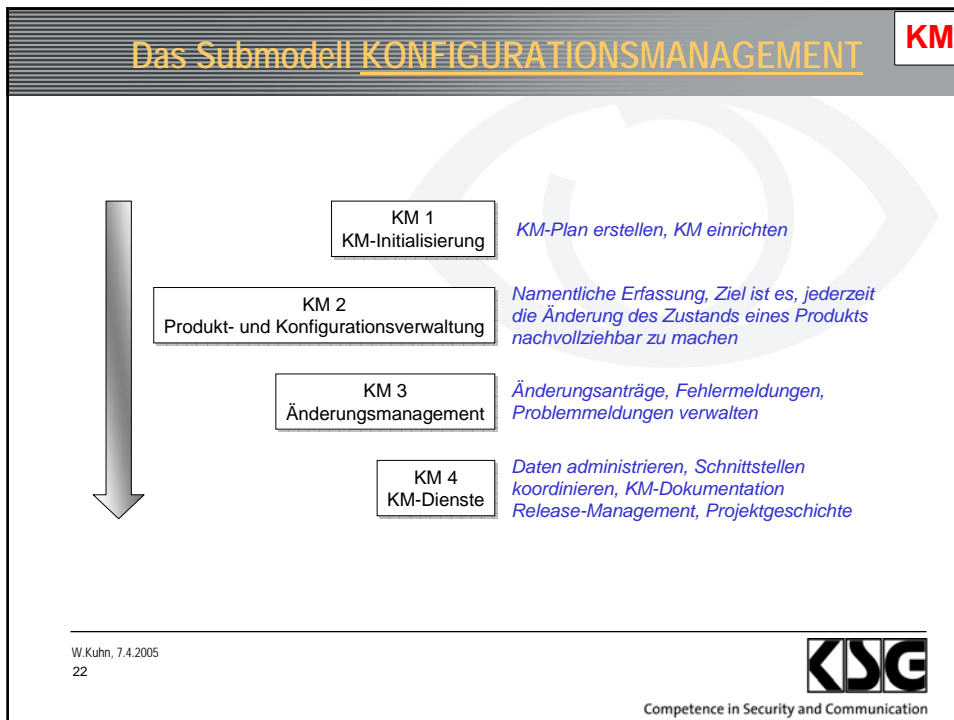
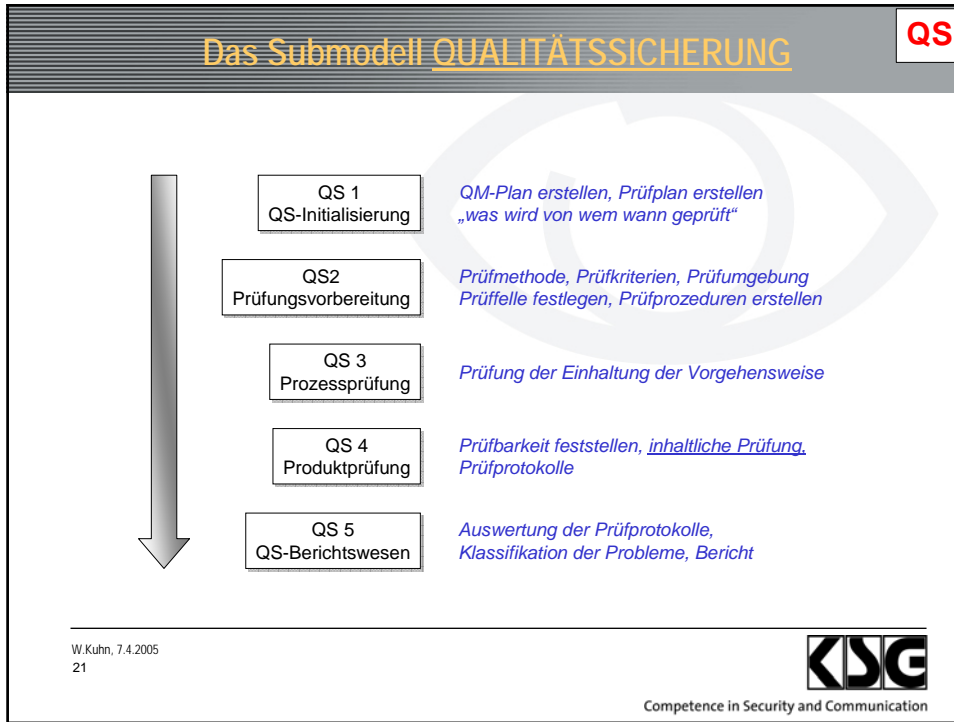
</trockener einschub>

W.Kuhn, 7.4.2005  
18



Competence in Security and Communication





### Beispiel: Inhaltsverzeichnis der Anwenderforderungen

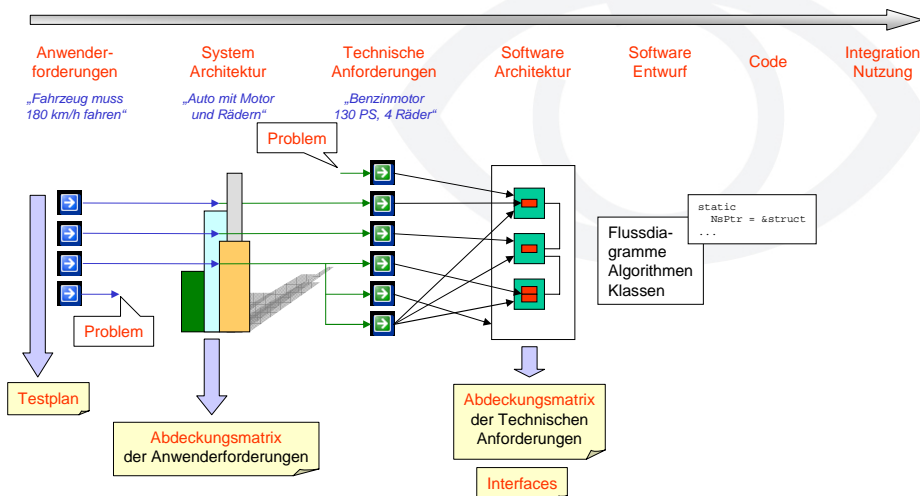
1. Einleitung
2. Ist-Aufnahme und Ist-Analyse
3. IT-Sicherheit
4. Bedrohungs- und Risikoanalyse
5. Fachliche Anforderungen
  - 5.1 Grobe Systembeschreibung
  - 5.2 Organisatorische Einbettung
  - 5.3 Nutzung
  - 5.4 Kritikalität des Systems
  - 5.5 Externe Schnittstellen
  - 5.6 Beschreibung der Funktionalität
  - 5.7 Qualitätsanforderungen
6. Randbedingungen
  - 6.1 Technische Randbedingungen
  - 6.2 Organisatorische Randbedingungen
  - 6.3 Sonstige Randbedingungen

W.Kuhn, 7.4.2005  
23



Competence in Security and Communication

### Die IDEE des V-Modells




The diagram illustrates the V-model of software development. The left side (development) descends from requirements to implementation, while the right side (testing) ascends from implementation back to requirements. Key components include:

- Anwenderforderungen:** „Fahrzeug muss 180 km/h fahren“
- System Architektur:** „Auto mit Motor und Rädern“
- Technische Anforderungen:** „Benzinmotor 130 PS, 4 Räder“
- Software Architektur:** (represented by a box with three red indicators)
- Software Entwurf:** (represented by a box with three red indicators)
- Code:** static MapPtr = &struct ...
- Integration Nutzung:** (represented by a box with three red indicators)

Supporting elements include:

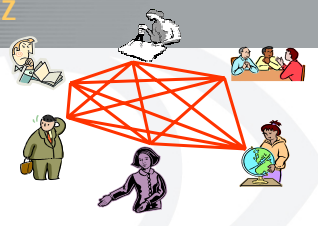
- Problem:** A central box at the top with arrows pointing to the requirements and architecture levels.
- Testplan:** A box at the bottom left, connected to the requirements level.
- Abdeckungsmatrix der Anwenderforderungen:** A box at the bottom center, connected to the system architecture level.
- Abdeckungsmatrix der Technischen Anforderungen:** A box at the bottom right, connected to the technical requirements level.
- Interfaces:** A box at the bottom right, connected to the software architecture level.
- Flussdiagramme, Algorithmen, Klassen:** A box at the bottom right, connected to the software design level.

W.Kuhn, 7.4.2005  
24




Competence in Security and Communication

### Sinnvoller Einsatz



1. Große Teams
2. AG/AN im Team
3. Standorte, Internationale Zusammenarbeit, mehrere Unternehmen
4. Dauer > 3 Monate
5. (tw.) kritische Anforderungen
6. Hohe erwartete Lebensdauer des Produktes

W.Kuhn, 7.4.2005  
25




Competence in Security and Communication

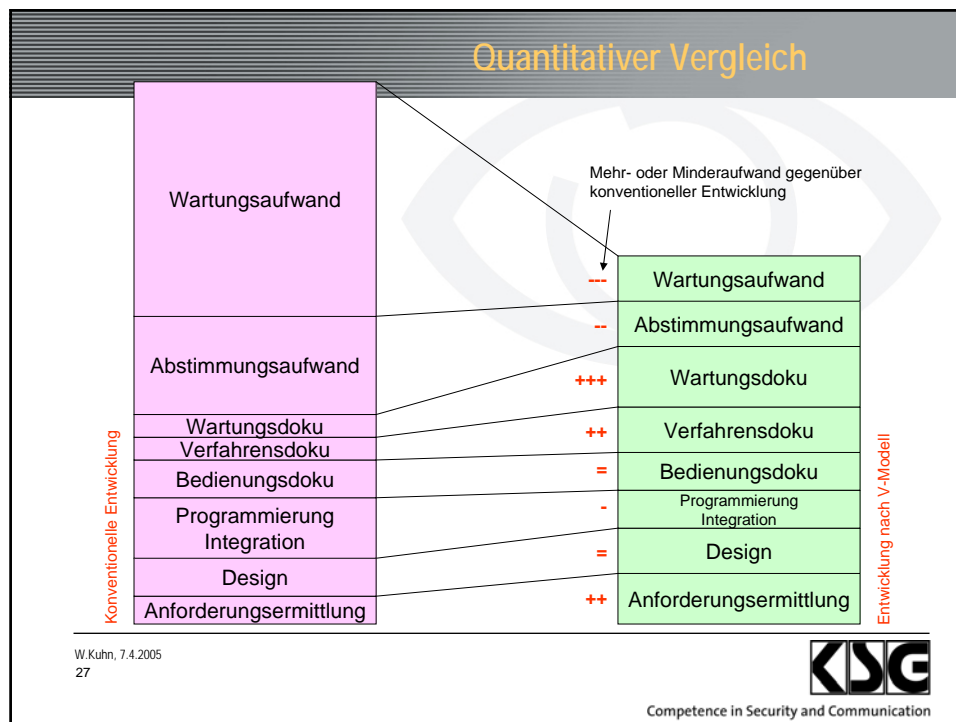
### Die praktischen Erfahrungen ... allgemein

1. Der „Sense of Urgency“ muss erkannt werden, Umsetzung ähnlich wie beim „Change Management“  
*„Menschen erzeugen Qualität, nicht Vorgehensmodelle“*
2. Je *größer* die Arbeitsgruppen, desto mehr Formalismus ist notwendig
3. Je *verteilter* die Arbeitsgruppen, desto mehr Formalismus ist notwendig
4. Beim Tyloring Beratung zukaufen

W.Kuhn, 7.4.2005  
26



Competence in Security and Communication



### Die praktischen Erfahrungen ... 😊

1. Das V-Modell ist „da“, man braucht sich nichts neues zu überlegen
2. Unabhängig von den eigentlichen *Methoden* der Software-Entwicklung
3. Durch den Formalismus wird man gezwungen, immer an (möglichst) alles zu denken
4. Erziehung zum strukturierten Denken
5. Gleiches „Wording“ in den einzelnen Teams
6. Vollständige Verfolgung der Anwenderforderungen
7. Dokumentfreigabe

W.Kuhn, 7.4.2005  
28

**KSG**  
Competence in Security and Communication

### Die praktischen Erfahrungen ... Caveats ...

1. Das V-Modell alleine garantiert noch keinen Erfolg  
*„Menschen erzeugen Qualität, nicht Vorgehensmodelle“*
2. Falsche oder unvollständige Anforderungen werden durch das V-Modell nicht richtiger oder gar vervollständigt (GIGO – Garbage in, Garbage out)
3. Sich nicht versklaven lassen; sich die Freiheit nehmen, Dinge für das eigene Umfeld zu interpretieren und abzuändern
4. Erfolg in kurzfristig leider schwer messbar ... € ?

W.Kuhn, 7.4.2005  
29



Competence in Security and Communication

- „Man kann nichts verbessern, wenn man nichts verändert“  
*Tom DeMarco*
- „Alles sollte so einfach wie möglich gemacht werden, aber auch nicht einfacher.“  
*Albert Einstein*

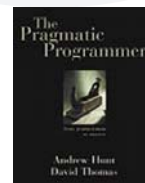
W.Kuhn, 7.4.2005  
30



Competence in Security and Communication

## Literatur

1. Tom DeMarco  
*The Deadline: A Novel about Project Management*
2. Capers Jones  
*Applied Software Measurement*
3. Andrew Hunt, David Thomas  
*The Pragmatic Programmer*
4. Gerhard Versteegen (Hrsg.)  
*Das V-Modell in der Praxis*
5. <http://www.v-modell.iabg.de>
6. <http://www.informatik.uni-bremen.de/~uniform/gdpa>



W.Kuhn, 7.4.2005  
31



Competence in Security and Communication