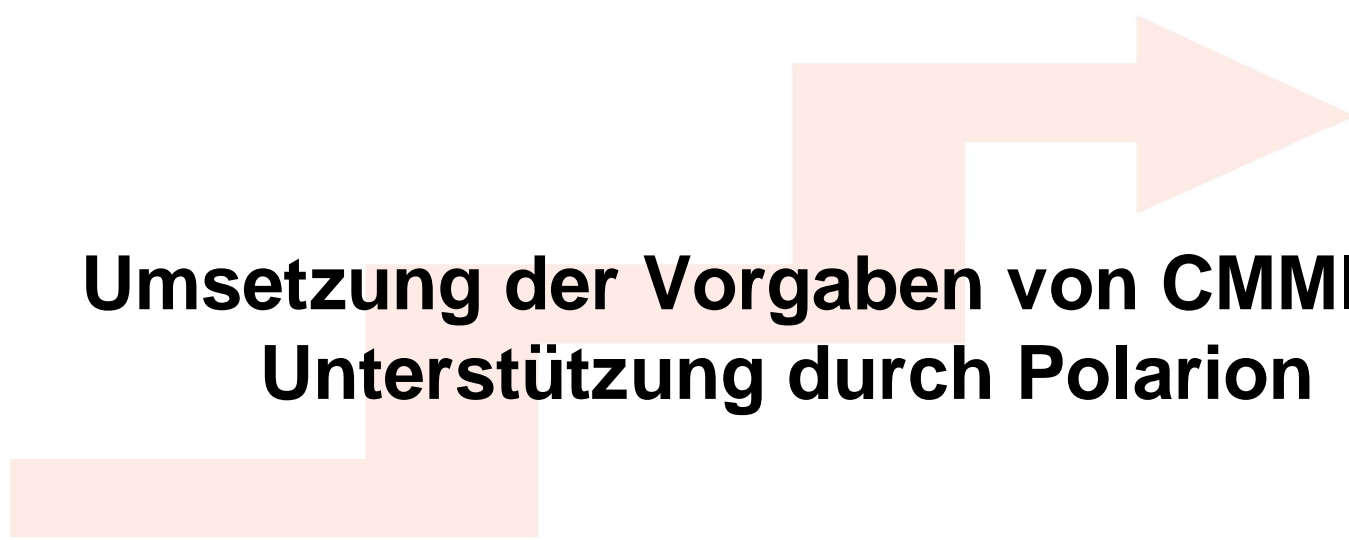


---



# Umsetzung der Vorgaben von CMMI mit Unterstützung durch Polarion

Dr. Ralf Kneuper

Beratung für Softwarequalitätsmanagement  
und Prozessverbesserung

Dr. KNEUPER

- Vorstellung des Verbesserungs- und Reifegradmodells CMMI
- Möglichkeiten und Grenzen der Werkzeugunterstützung
- Anforderungsmanagement - CMMI-Forderungen und Werkzeugunterstützung
- Test/Validierung der Anforderungen, Testmanagement – CMMI-Forderungen und Werkzeugunterstützung
- Projektplanung/-steuerung - CMMI-Forderungen und Werkzeugunterstützung
- Messung und Analyse - CMMI-Forderungen und Werkzeugunterstützung
- Konfigurationsmanagement - CMMI-Forderungen und Werkzeugunterstützung

Dr. KNEUPER

**Dipl.-Mathematiker, Univ. of Bonn**

**PhD Computing Science, Univ. of Manchester**

**1989-1995: Software AG**

- Quality assurance, quality management, ISO 9000

**1995-2005: Deutsche Bahn/TLC/DB Systems**

- Senior consultant, project lead
- Quality management, internal CMM(I) consultant, development processes, project management

**Since 2003: Independent consultant on CMMI**

**Former Speaker of GI SIG on software processes**

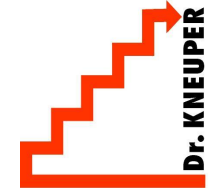
**SEI-Authorized CMM Lead Assessor, CMMI Lead Appraiser**

**Coordinator of the German CMM(I) Lead Appraiser and Instructor Board (CLIB)**



**Contact: [ralf@kneuper.de](mailto:ralf@kneuper.de)**

# Underlying Premise of Process Improvement

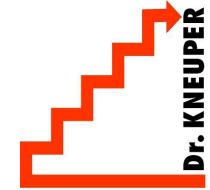


**“The quality of a product is largely determined by the quality of the process that is used to develop and maintain it.”**

Based on TQM principles as taught by Shewhart, Juran, Deming and Humphrey.

Dr. KNEUPER

# Capability Maturity Model Integration for Development (CMMI-DEV)



## Maturity model for the development of software and systems

- Used both for internal process improvement and as external proof of process maturity
- Defined improvement path

## First published in 2002 as successor to CMM

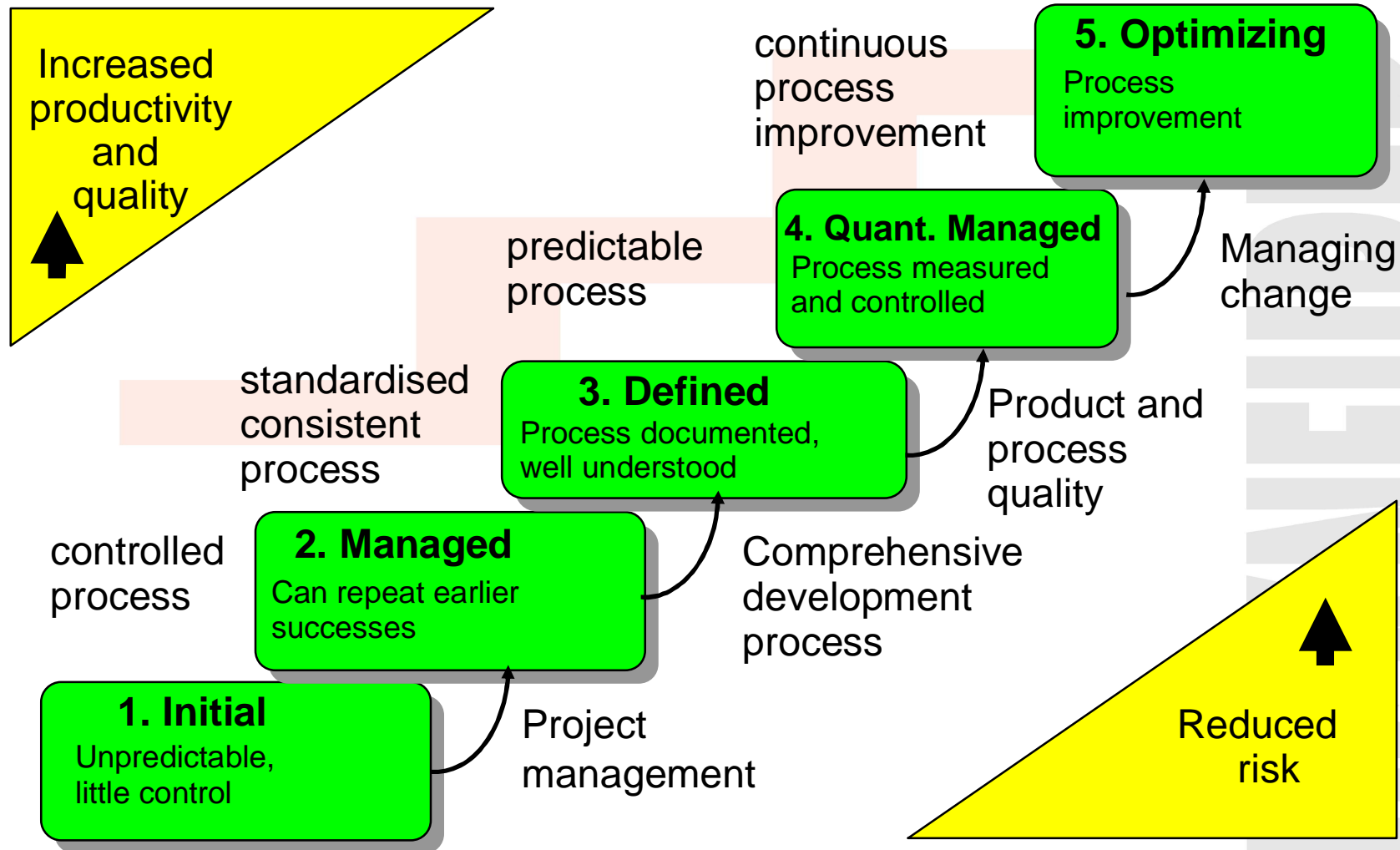
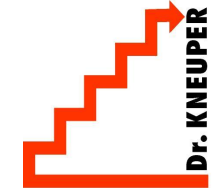
- Author: Software Engineering Institute (SEI), Carnegie Mellon University in Pittsburgh, USA
- Current version v1.2 published in August 2006

## Other „constellations“ of CMMI

- CMMI for Acquisition (CMMI-ACQ)
- CMMI for Services (CMMI-SVC)

Dr. KNEUPER

# The five maturity levels of CMMI



# CMMI-DEV v1.2

## Process Areas by Category and Maturity Level



	Process Mgmt.	Project Mgmt.	Engineering	Support
2		Project Planning (PP) Project Monitoring and Control (PMC) Supplier Agreement Management (SAM)	Requirements Management (REQM)	Configuration Management (CM) Process & Product Quality Assurance Measurement and Analysis (MA)
3	Organizational Process Focus (OPF) Organizational Process Definition (OPD) Organizational Training (OT)	Integrated Project Management (IPM) Risk Management (RSKM)	Requirements Development (RD) Technical Solution (TS) Product Integration (PI) Verification (VER) Validation (VAL)	Decision Analysis and Resolution (DAR)
4	Organizational Process Performance (OPP)	Quantitative Project Management (QPM)		
5	Organizational Innovation and Deployment (OID)			Causal Analysis and Resolution (CAR)

# Tool Support for CMMI

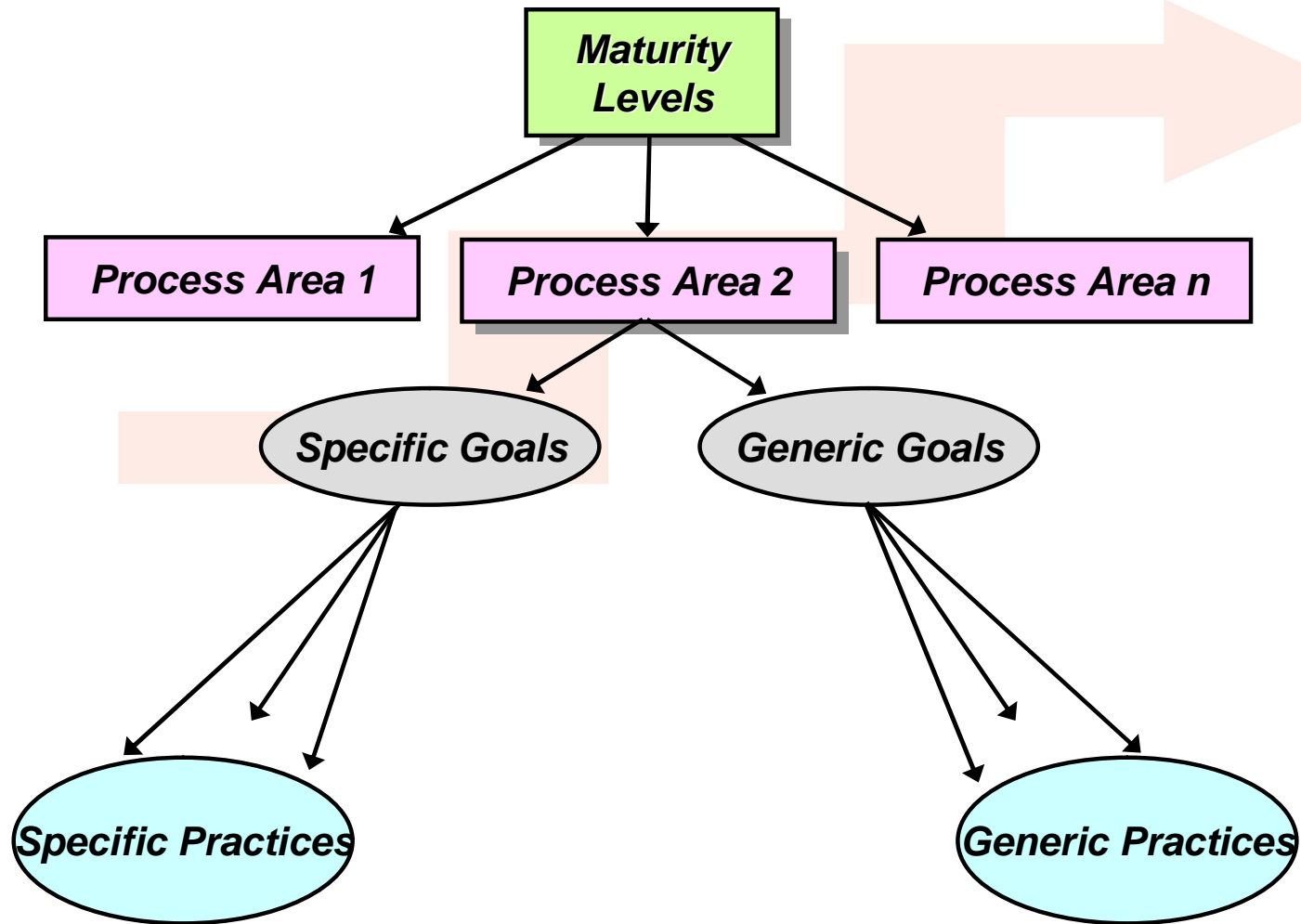


- CMMI does not require any tools
  - It is difficult but not impossible to satisfy the CMMI requirements without any SW tools
- Tools can support the implementation and help to satisfy the CMMI requirements
- Tools cannot in themselves satisfy the CMMI requirements
  - If a tool is not used in the defined way (or perhaps not at all) it will not help with satisfying CMMI requirements
- To be truly useful, processes must be „institutionalized“ in order to ensure that they are used consistently
  - See generic goals and practices

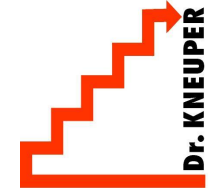
Dr. KNEUPER



# Structure of CMMI V. 1.2 Staged Representation



## Generic Goal Level 2 Managed Process



### Institutionalize a Managed Process (GG 2)

- Establish an Organizational Policy (GP 2.1)
- Plan the Process (GP 2.2)
- Provide Resources (GP 2.3)
- Assign Responsibility (GP 2.4)
- Train People (GP 2.5)
- Manage Configurations (GP 2.6)
- Identify and Involve Relevant Stakeholders (GP 2.7)
- Monitor and Control the Process (GP 2.8)
- Objectively Evaluate Adherence (GP 2.9)
- Review Status with Higher Level Management (GP 2.10)

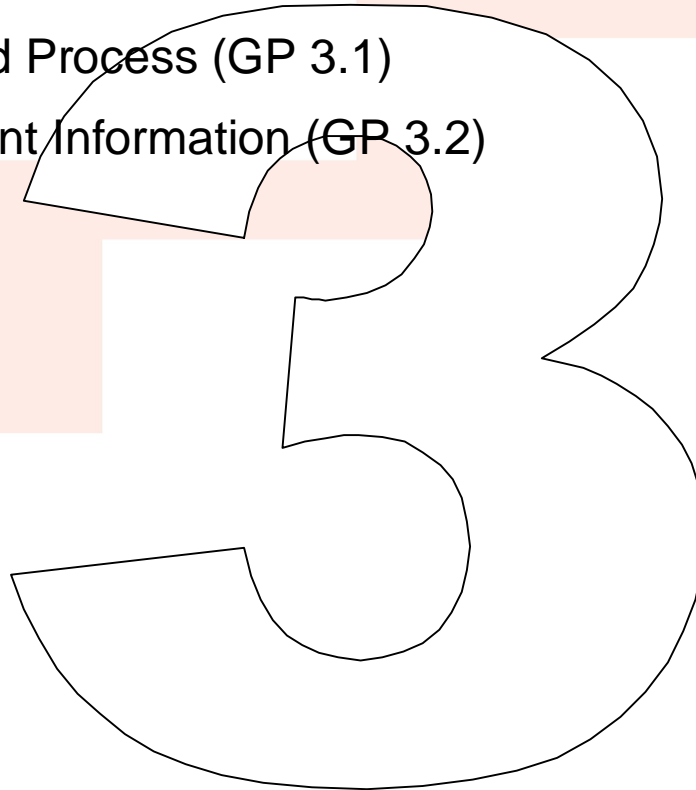
Dr. KNEUPER

## Generic Goal Level 3 Defined Process



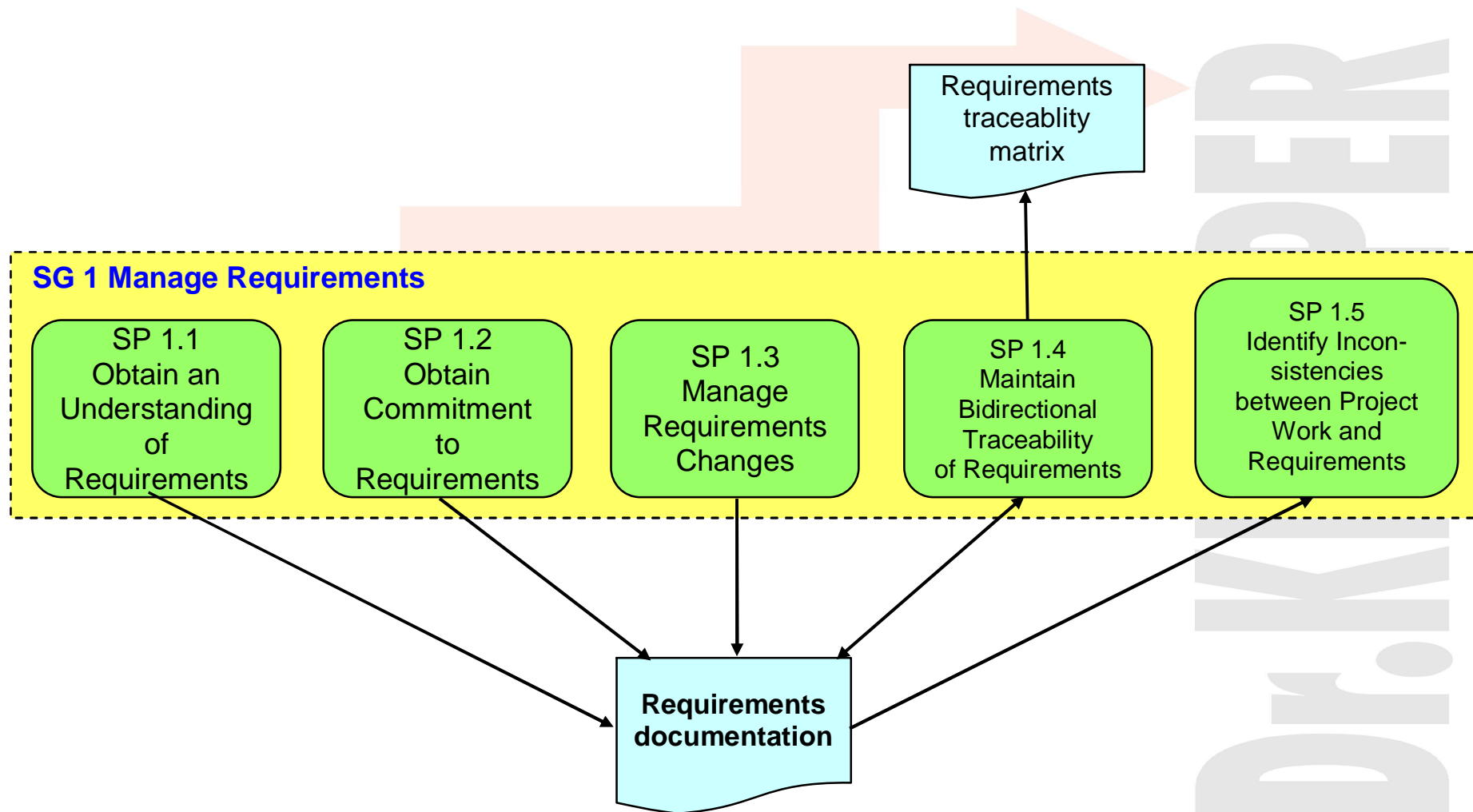
### Institutionalize a Defined Process (GG 3)

- Establish a Defined Process (GP 3.1)
- Collect Improvement Information (GP 3.2)

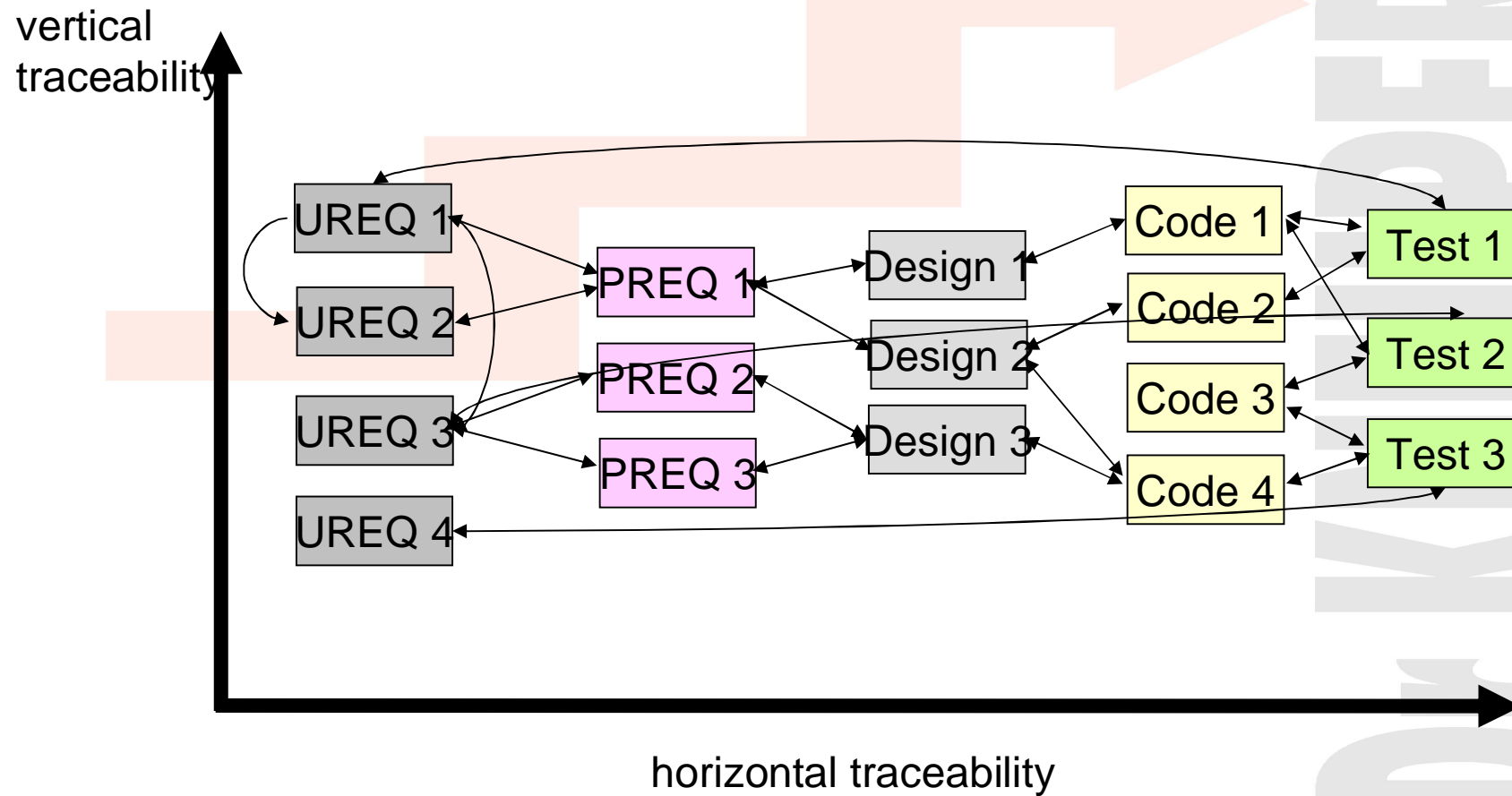
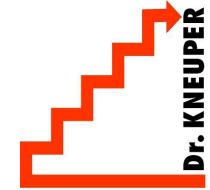


Dr. KNEUPER

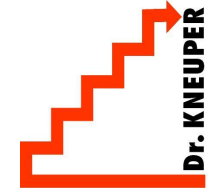
# Requirements Management



# Traceability of Requirements



# Why Traceability of Requirements?



## Forward traceability:

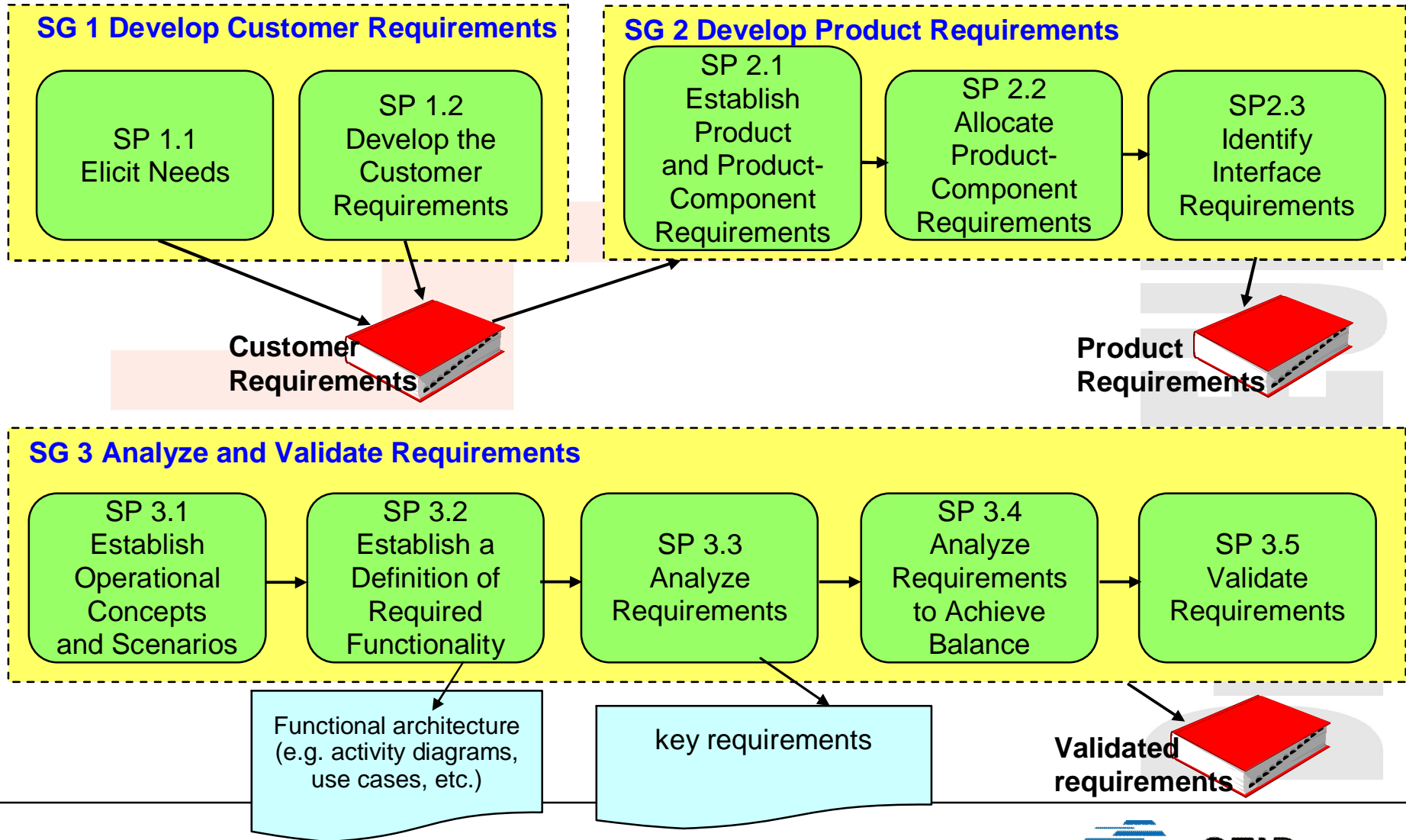
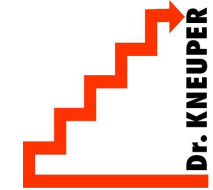
- Ensure that all requirements are implemented in the work products building on them (design, code, test, etc)
- Helps to plan and track work
- Quickly identify impact of a change request

## Backward traceability:

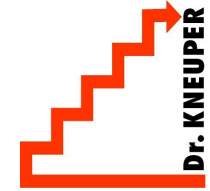
- Prevent „requirements creep“
- „Why do I create this product?“
- „What is affected if I leave out this test?“

Dr. KNEUPER

# Requirement Development



# Supporting Requirements Management and Requirements Development Using the Polarion Toolset





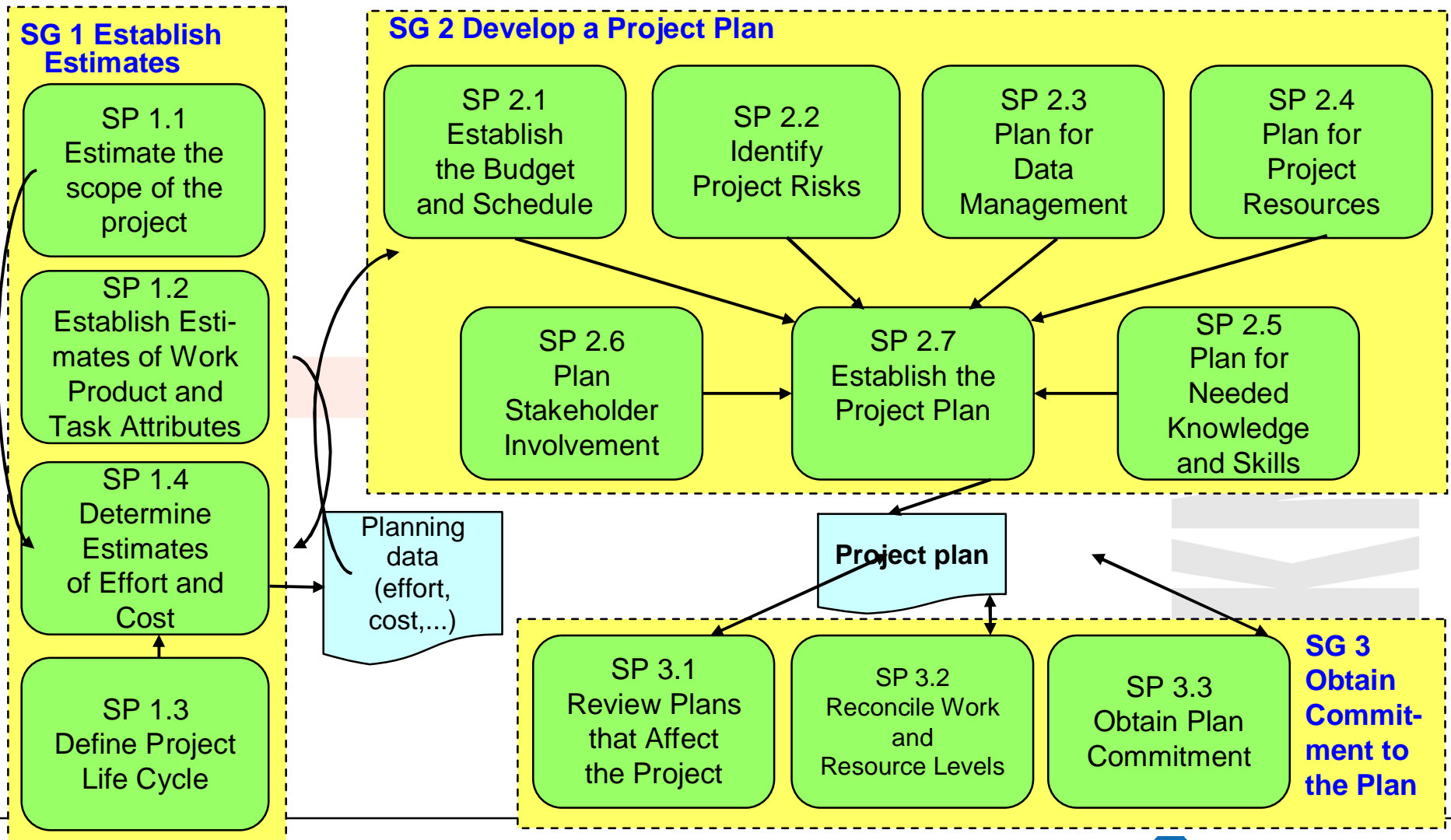
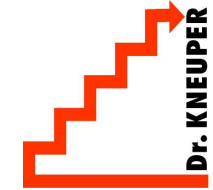
## Issues to Consider



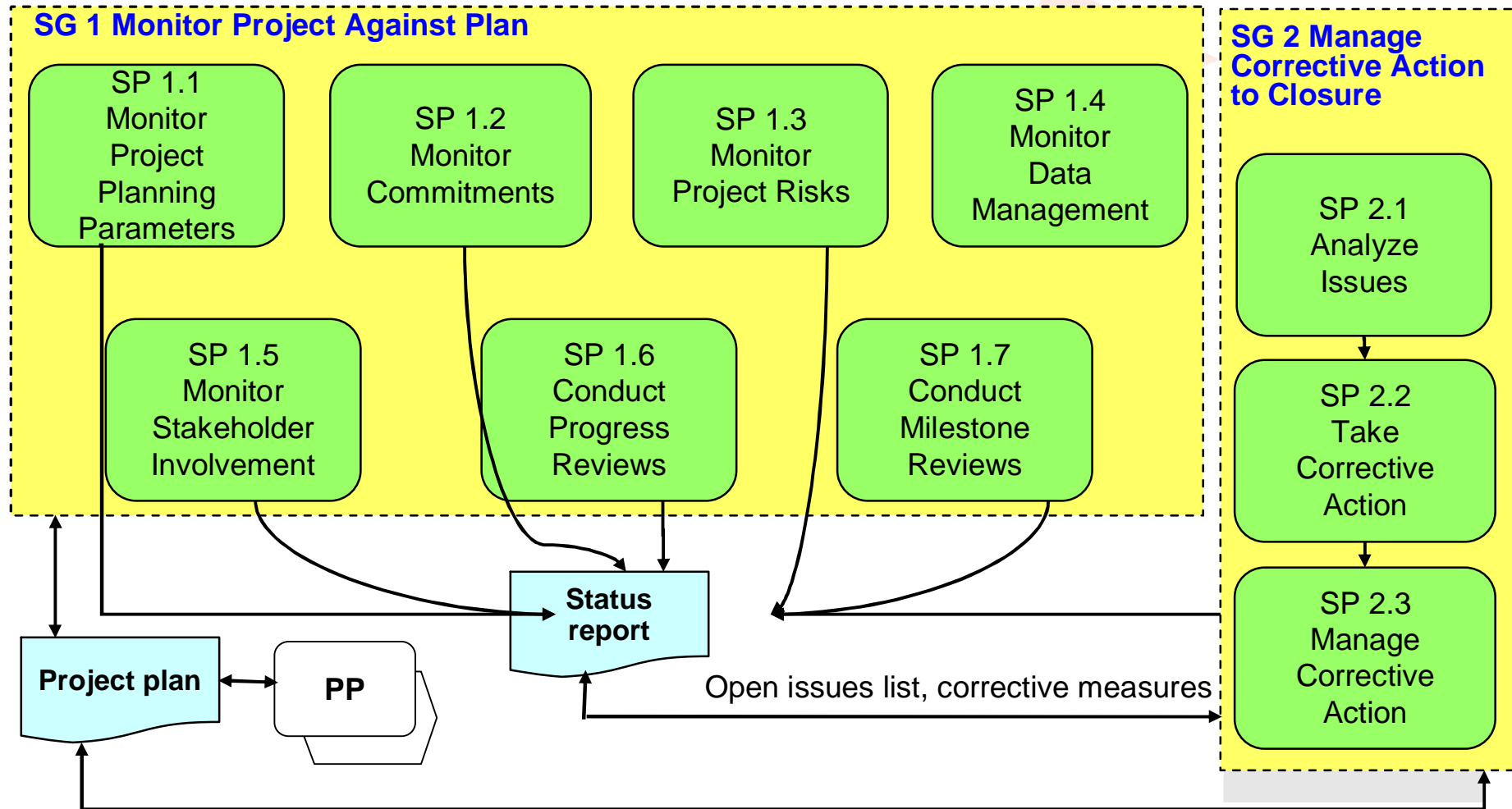
- The Polarion toolset helps to manage the requirements and their traceability
- It cannot identify and understand the requirements for you
- It cannot know the relationships between requirements and project work. You have to identify and enter this information yourself so that the Polarion tools can then help you to manage it
- Maintaining traceability only becomes useful if this information is used, e.g. for identifying inconsistencies

Dr. KNEUPER

# Project Planning



# Project Monitoring and Control



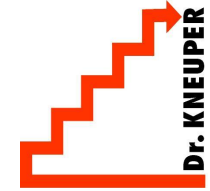
# Project Planning and Project Monitoring and Control in an Agile Environment



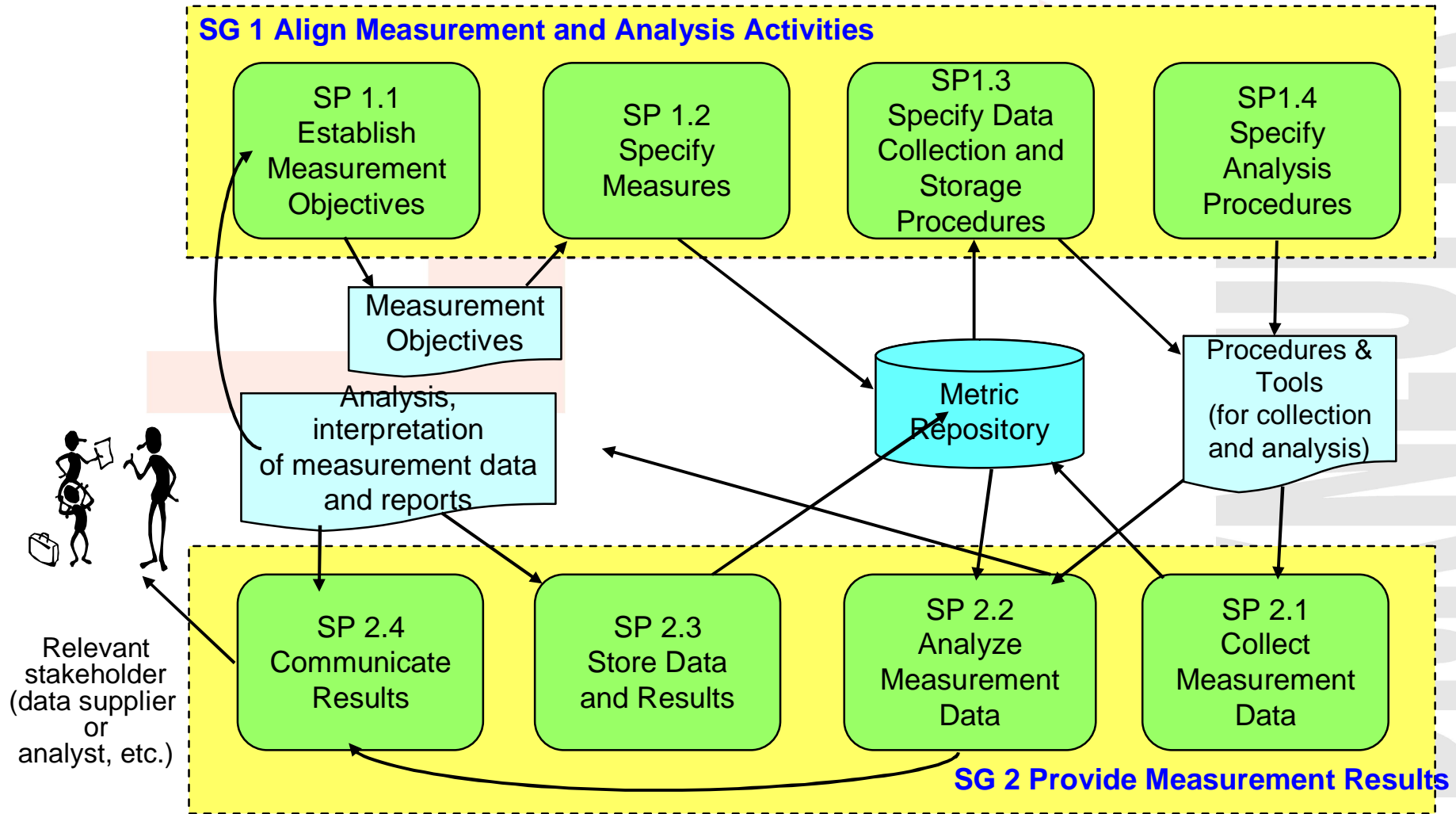
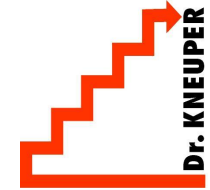
- Create a plan but adapt it when necessary
- Have a detailed short-term plan (e.g. covering one monthly Scrum sprint) plus a rough long-term plan
- Estimation is needed to plan a sprint / increment
- Plan and monitor in terms of user stories etc.
- Agile development usually monitors progress closely but with little defined measurements and little documentation
- CMMI requires more use of measurements for project tracking than usually used in agile development, and more documentation of the results
- Summary: agile development plus some add-ons can satisfy the CMMI requirements on PP and PMC

Dr. KNEUPER

# Supporting Project Planning and Monitoring Using the Polarion Toolset



# Measurement and Analysis



## Project level

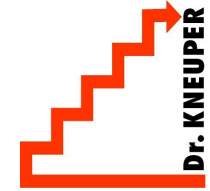
- Mainly used to keep project on track (-> Project Monitoring and Control)
- Typically measurements reported regularly in a status report, e.g. budget spent, earned value, etc
- Important to include measurements on ALL project goals, not just time and budget:
  - staff development
  - process improvement
  - customer satisfaction
  - ...

## Organizational level

- Look at longer-term development of organization
- Typical example: Balanced Score Card
  - Financial view
  - Process view
  - Product view
  - Employee view
  - Customer view

Dr. KNEUPER

# Supporting Measurement and Analysis Using the Polarion Toolset





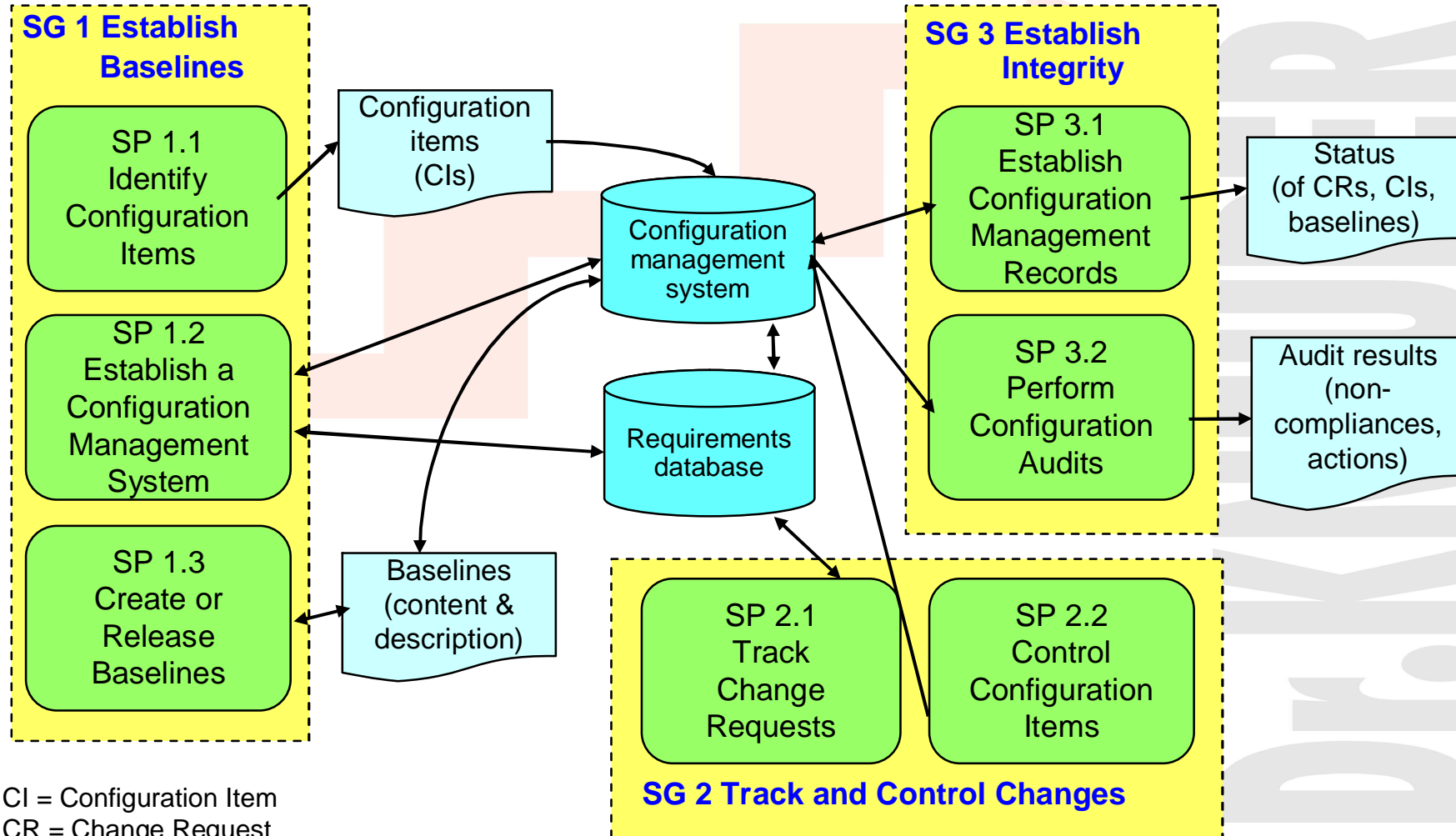
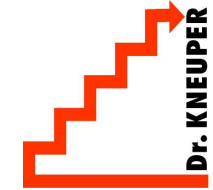
## Issues to Consider



- By definition, a tool such as Polarion can strongly support the second specific goal for Measurement and Analysis, but provide very little support for the first specific goal.
- Remember to start from information needs, and then derive the relevant reports and measurements in Polarion, and not to start with the reports and measurements that are available but may not always provide the information you need.

Dr. KNEUPER

# Configuration Management



## A baseline

- is a consistent collection of work products
- forms the basis for further activities

## This implies that a baseline

- is planned
- is reviewed for consistency, correctness and completeness
- is put under change control and must be kept consistent
  - therefore files that are not maintained should not be part of the baseline
- is more than a backup of the current project directory at a certain point in time

**Typically, a baseline is created at major milestones such as the end of a development phase**

Dr. KNEUPER

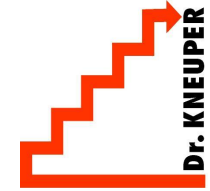
## Change Management can mean very different things

### Manage changes at different levels:

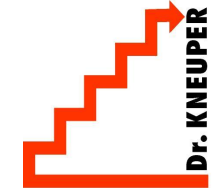
- Change of requirements
  - in CMMI covered by REQM, SP 1.3
  - typically leads to changes of multiple work products
- Change of work products
  - in CMMI covered by CM, SG 2
- Change of own organization
  - training, rollout, etc.
  - in CMMI covered by generic goals and practices, OPF, OT
- Change of customer organization
  - training, rollout, etc.
  - not covered by CMMI but e.g. by ITIL

Dr. KNEUPER

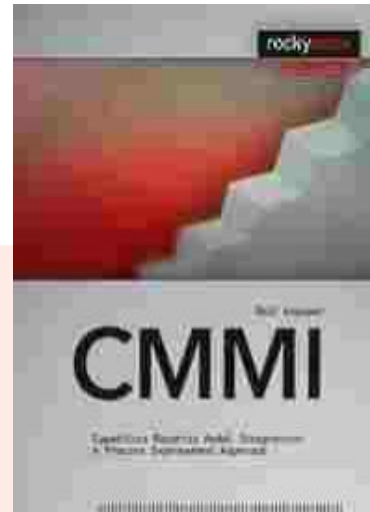
# Supporting Configuration Management Using the Polarion Toolset



## Further reading



Kneuper:  
CMMI.  
Verbesserung von Software- und Systementwicklungsprozessen mit Capability Maturity Model Integration (CMMI-DEV).  
dpunkt.verlag 2007



Kneuper:  
CMMI.  
Improving Software and Systems Development Processes.  
Rocky Nook 2009



Kneuper, Walmüller:  
CMMI in der Praxis.  
Fallstudien zur Verbesserung der Entwicklungsprozesse mit CMMI.  
dpunkt.verlag 2009

