



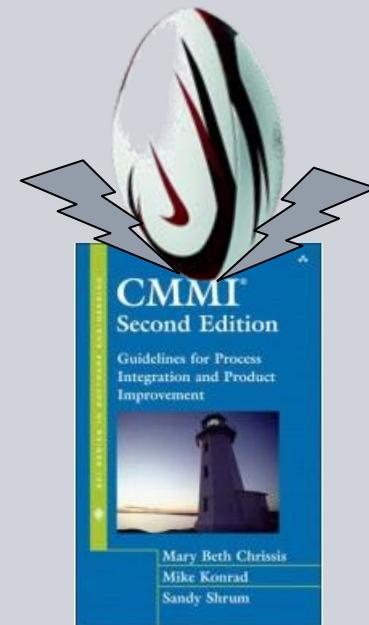
Hidden Treasure: The Implementation of CMMI Practices by Agile Methods

Dr. Winfried Russwurm

Siemens AG, Corporate Technology, Munich, Germany

Presentation/Conference ID: # 2201
presented at SEPG Europe 2010, Porto, Portugal

Siemens AG
Corporate Technology
Processes & Production
Enterprise Processes
Consulting



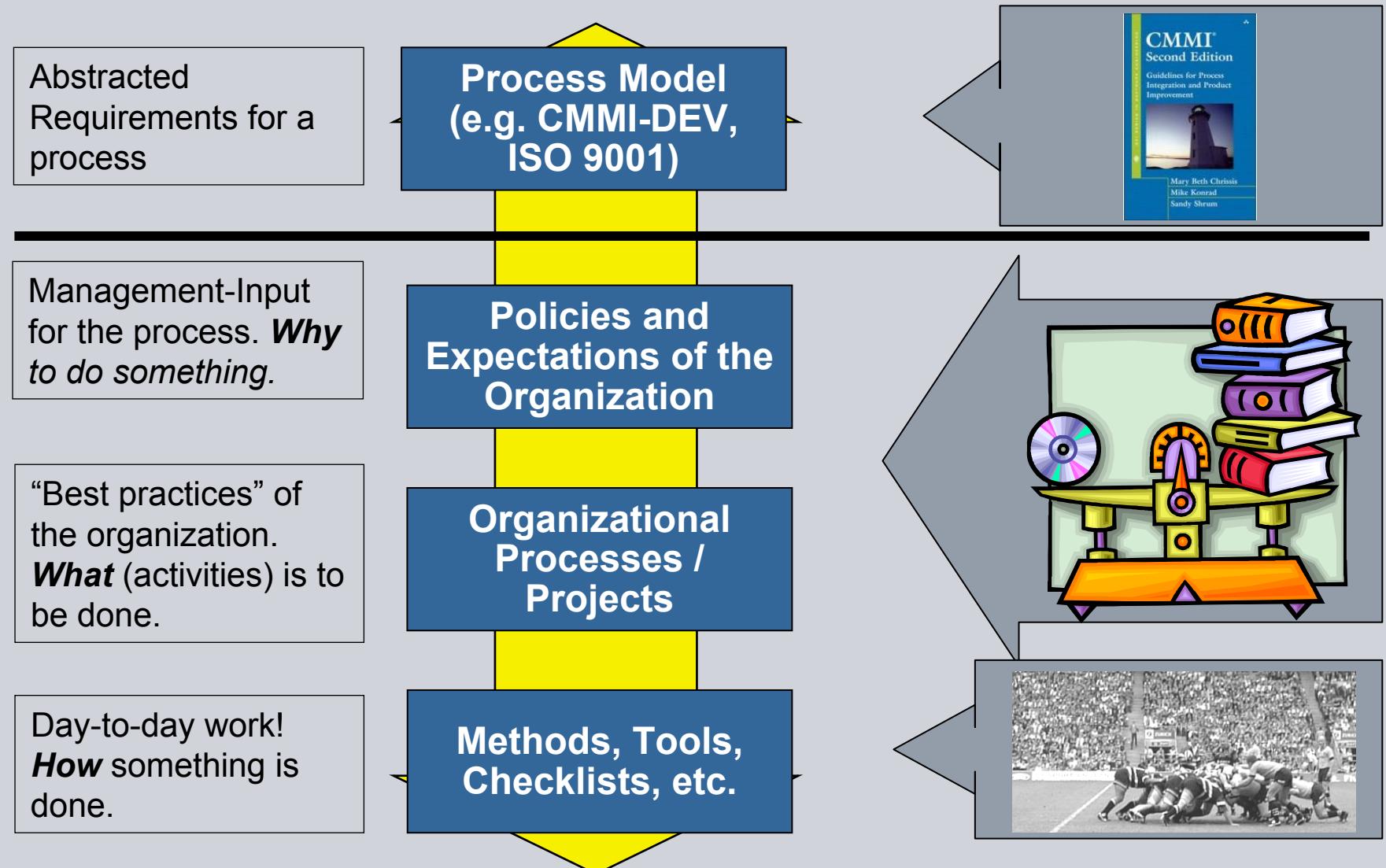
Motivation

- CMMI-DEV is now here for numerous years as the best renowned model for development process improvement.
- Simultaneously, Scrum and other Agile methods (particularly for software development) are becoming more popular.
- Both have the same intention – less defects, better quality, faster results etc.
- Both are based on positive experience with certain approaches, but originating from different historical background and boundary conditions.
- Both have also been used inappropriately, which in consequence led to frustration and undesirable results.
- Therefore both have polarized the community quite rapidly.
- Are both approaches actually so far off from each other or is there a hidden treasure called synergy?
- Can both, CMMI and Agile methods, mutually benefit from each other?
- Can evidence from Agile methods be used for a SCAMPI appraisal?

Let's look deeper: What is the difference?

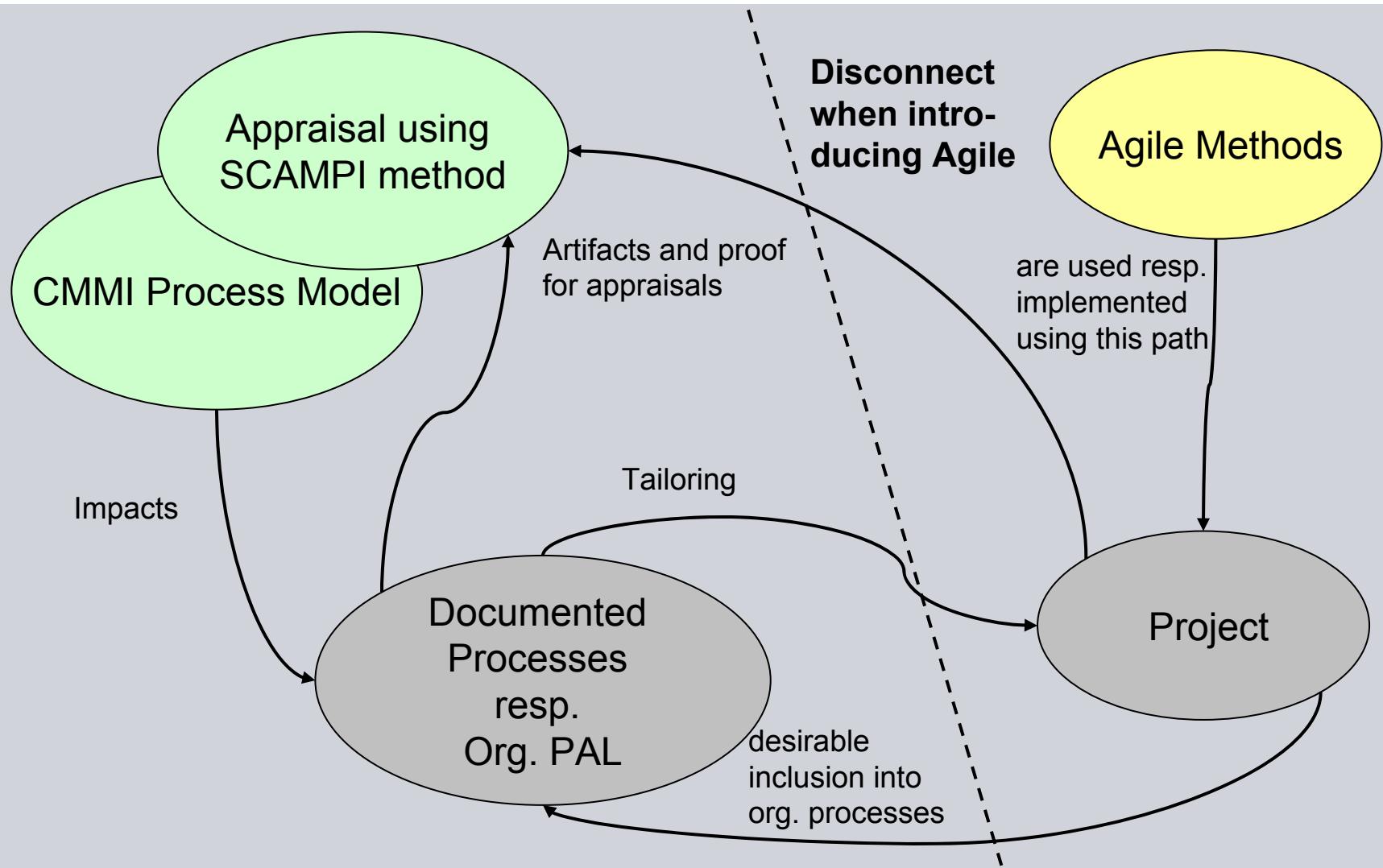
- It is the “use case” of CMMI and Agile methods (here Scrum as an example).
- CMMI suggests “**why to do**” and “**what to do**” needing implementation using a “how to”. Usually the “how to” is part of the organizational process library and used by projects.
- Scrum (as an example) directly addresses some “**how to’s**” (without referring to a process model or an organizational process) to be used by projects originating directly from books, discussions and ScrumMaster’s knowledge.
- Agile methods / Scrum are not claiming to implement a process model (e.g. CMMI-DEV). For example, Scrum focuses mainly on some “software management” methods.
- Topics, such as supplier management, organizational process improvement and others are not covered by Agile methods.
- An organization has interfaces to subcontractors, customers, business, shareholders etc., therefore these topics need to be implemented as well (in addition to Agile methods).

The “Use Cases” of CMMI-DEV and Agile Methods



Relationship between CMMI, PAL, Agile, Project and SCAMPI

SIEMENS



How to interpret CMMI in an Agile context?

- Agile methods (particularly Scrum) comprise methods and method collections. They are not a “process model”.
- Can these methods be interpreted as “sufficient” implementations of CMMI?
- For an answer, it needs to be understood:
 - How words / terms / phrases in CMMI are used
 - What is the purpose of the practice?
 - What risks or problems does the practice address?
 - In various contexts, artefacts will vary greatly (also for Agile methods).
 - How work is done in an Agile Environment
 - What represents the work done?
 - What is the organization or project doing to achieve the purpose
- If the efforts made by Agile methods accomplish what CMMI expects to accomplish, the intention is met, and there is no reason, why Agile methods cannot be considered.

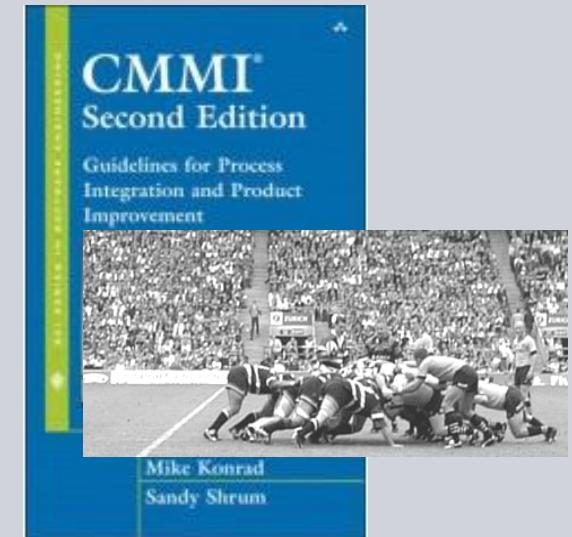
The “footprints” of actual work: Artifacts and the SCAMPI method

SIEMENS

- To show that an organization has implemented CMMI practices, frequently a SCAMPI appraisal (Standard CMMI Appraisal Method for Process Improvement) is used.
- The SCAMPI method relies heavily on evidence, i.e. proof or “footprints” that an activity has been implemented (i.e. performed).
- Evidence may consist e.g. of documents or other suitable artefacts and interviews (oral affirmation).
- It is a misunderstanding, that SCAMPI (and CMMI) needs an overkill of documents.
- It is the proof that counts (that a practice is implemented), not the amount of documents. SCAMPI does not ask for documents, it asks for evidence (which comprises much more than just documents).
- The SCAMPI lead appraiser needs to know about Agile methods, work results from these methods and how evidence is created (and saved/stored/document etc. for further use in the project and for an appraisal).

Caveat: Inappropriate Use of CMMI and Agile Methods

- But there are caveats: Doing Agile methods just because it is “cool” and not knowing about the typical application scenarios, side effects and impacts, will lead to undesirable results (e.g. some companies already reporting loss of quality due to inappropriate use). Agile methods have to improve the organization’s business, but not impair it.
- Confusing the CMMI with a documented process or other misunderstandings (e.g. everything needs to be in a document and have to named the same as “typical work products”) leads to frustration and bureau-“CRAZY”.
- The magic word here is “adequacy”: Is there proof that it helps the organization and the projects to do the work better and to create better results?
- If the lead appraiser is not ready to appraise in an Agile environment, the result may be undesirable and create “scorched earth”.
- To illustrate that Agile can implement some CMMI practices, let’s analyze some Agile practices. The selection is incomplete, but typical.



Example 1: Planning Poker

- Planning poker (as used for product backlog/user stories) may be performed by using special “card decks” with tiered “values”.
- Some online-applications are already available.
- This can be seen as a size estimation method (not effort estimation).
- It systematically estimates relative size of an object (e.g. a requirement) compared to a (small) initial size.
- Results need to be collected and made visible to serve as evidence, e.g. by creating a hand-written record of all estimates and a scan to serve as “electronic” evidence.
- Size estimation is in CMMI in the project planning process area.
- Appraisals in a non-agile environment frequently identify size estimation as a weakness.



Example 2: Self Organizing Teams

- Self-organizing teams: Compared to non-agile projects, this employs a significantly closer collaboration of stakeholders / team members with dynamic task self-assignment.
- CMMI as a model describes a similar concept in the (optional) addition IPPD (Integrated Product and Process Development). IPPD uses “integrated teams” resembling Agile practices in many instances (see IPM process area). CMMI as a model asks for “roles” and “tasks”, which may be implemented by the ScrumMaster, team and product owner.
- In addition, CMMI encourages the organization to be prepared for projects with integrated or self-organizing teams, in order to avoid an “alien” status. See process area OPD.
- This organizational aspect is not clearly addressed e.g. in Scrum, but is indispensable, since working in contradiction to an organization’s objectives and policies does not provide sustainability.
- In the future CMMI-DEV v1.3 it is intended to move 2 IPPD practices to the core model. Unfortunately, the IPPD addition as such will go.



Example 3: Iterations and Increments

- Here one of the most “prominent” misunderstandings is encountered.
- CMMI is not a process, it is a model.
- Therefore it does not prescribe nor represent nor imply a waterfall model.
- Iterations and increments: CMMI does not align to a certain paradigm.
- CMMI asks to provide useful paradigms (that help the organization to do their business)
- CMMI gives several examples (e.g. in process area OPD),
 - Spiral
 - Evolutionary
 - Incremental
 - Iterative
 - and Waterfall
- If time-boxing and testing/regression at the end of each iteration is effective and helps to develop a successful product within the goals of the project, it is fully within the intentions of CMMI.



Example 4: Sprint Backlogs

- Sprint backlogs: This is a planning method for each sprint and therefore seen as a schedule (and work breakdown), or at least for a detailed planning for the respective sprint.
- Makes only sense, if all needed and performed activities (e.g. testing, documentation) are included. There must be no unplanned activities (since they would blow the sprint, as they would blow a conventional project).
- A digital picture of the task board (with attached cards) may serve as evidence.
- Some agile teams are already using spreadsheet tools, the file can be used as evidence as well.



Example 5: Sustainable Pace

- Sustainable pace is used for a positive work-life-balance and needs to be kept for a longer period without compromising quality of the product.
- This should be an objective of all projects, Agile or not.
- CMMI does not use the same terms / wording, but elaborately addresses realistic planning (e.g. by obtaining commitments), checking resource situation and taking countermeasures in case of deviations.
- Sustainable pace is therefore one of the key pillars of CMMI.
- Scrum makes accumulated delays (“snow-plough effect”) quite rapidly visible. However, “descoping” may remove important requirements to get back on track. This is a potential pitfall for quality, which in turn is not desired in both Agile methods and CMMI. Needs also to be seen in the context of the business case of the project.
- It frequently occurs that a project (agile or not), is exposed to pressure from upper management: “Do something, but needs to be completed by Christmas”.
- This is not according to Agile and well as to CMMI.
The policies and management commitment (as suggested by CMMI) should avoid such situations in general.



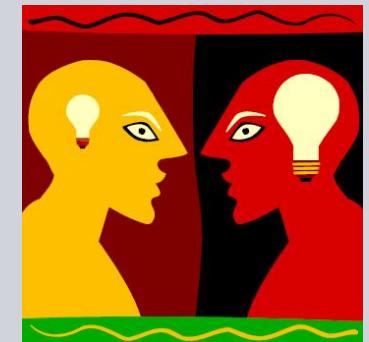
Example 6: Sprint Retrospectives

- Sprint retrospectives can be seen as team's lessons learned at the end of a sprint. Results to be immediately used for the next sprint.
- CMMI suggests this in several practices (e.g. in OPF, GP3.2, IPM). A big benefit of Agile methods here is, that improvements are incorporated in a timely manner.
- However, since Agile projects frequently have a disconnect to the organizational processes, experiences are often not available for other projects.
- Experiences are collected mostly on how the Agile methods are performed, other things, such as supplier management are frequently neglected.
- Here, non-agile projects (in appraisals) frequently exhibit gaps and miss opportunities for continuous improvement. If done, it happens mostly at the end of the project only and misses important aspects (after the fact).
- Minutes or other stored records or digital pictures can serve as evidence for the retrospective.
- Some larger organizations have introduced "joint retrospectives" with members from various teams participating, which is even more in the sense of CMMI.



Example 7: Product Owner

- Product Owner: As a concept (not by a particular name) it is addressed by CMMI in various places (e.g. REQM, RD, GP2.7).
- For being successful, the product owner needs to have the experience, competence and particularly resources (e.g. time, effort) to contribute appropriately.
- In non-agile projects, the comparable function (e.g. product manager) frequently does not have sufficient resources and does not attend the respective activities in an appropriate way.
- This is also a major risk in Agile projects (for the product owner), and would lead to the same issues and shortfalls as in non-agile projects (just the naming of a product owner is not sufficient).
- CMMI addresses this issue in all process areas using the concept of institutionalization (GP2.7, GP2.3, GP2.4, etc.).



Other Examples

- As mentioned, this was not a complete collection.....
- Sorry for missing your “favorite practice”.....
- Even some more CMMI practices may be seen as implemented by Agile methods, e.g. some PPQA practices by the ScrumMaster’s activities.
- As mentioned above, CMMI has many more practices than are usually covered by Agile methods.
- If you are targeting at “model-based process improvement”, Agile methods by themselves are not sufficient to achieve major process improvement goals.
- Agile Methods do not claim to implement complete (or dedicated parts of) process models (and are not doing in practice).
- In order to cover important aspects of the projects’ and organization’s work, a comprehensive organizational process improvement remains indispensable.
- The notorious question: „If we are doing Scrum, does this automatically give us a CMMI ML3?“ reveals basic misunderstandings.
- The fundamental question is always: What do we need to create business success?

Summary and Outlook

- Some Agile methods implement easily CMMI practices, that are issues in traditional settings (e.g. size estimation, lessons learned).
- There are indications, that the puristic Agile hype is moving towards a “peaceful coexistence” with CMMI for a better business value.
- Companies return to the more comprehensive “model-based” process improvement, but using proven Agile practices.
- CMMI v1.3 should have more informative material on putting Agile methods in context.
- People working in Scrum projects frequently report that they have lots of fun, created by leadership implementation in contrast to “command-and-control” management.
- CMMI does not demand “command and control”, but uses agreements, commitments and integrated teams (in IPPD and in the future also in the core) with a strong leadership focus.
- So you can have lots of fun with CMMI as well!
- And even more fun with having implemented useful Agile methods and getting “credits“ for this in a CMMI and SCAMPI environment!



Thank you for your Attention!



Dr. Winfried Russwurm

russwurm@siemens.com

Phone: +49-89-636-42627

Siemens AG, CT T DE TC 1, GTF PRO EPC
Otto-Hahn-Ring 6
D-81739 Munich, Germany

® CMM, CMMI, Capability Maturity Modeling, CERT, and CERT Coordination Centre are trademarks copyrighted of the Carnegie Mellon University, Pittsburgh, Pennsylvania (USA), registered in the U.S. Patent and Trademark Office

SM CMM Integration, SCAMPI, SCAMPI Lead Assessor, SCAMPI Lead Appraiser, IDEAL, SEI, SEPG, Interim Profile, Personal Software Process, PSP, SCE, Team Software Process, and TSP are service marks of the Carnegie Mellon University

TM Carnegie Mellon Software Engineering Institute design and stylized hexagon are trademarks of the Carnegie Mellon University