

10 Agile Years Later

Lessons Learned

Martin Lippert (VMware)
mlippert@vmware.com

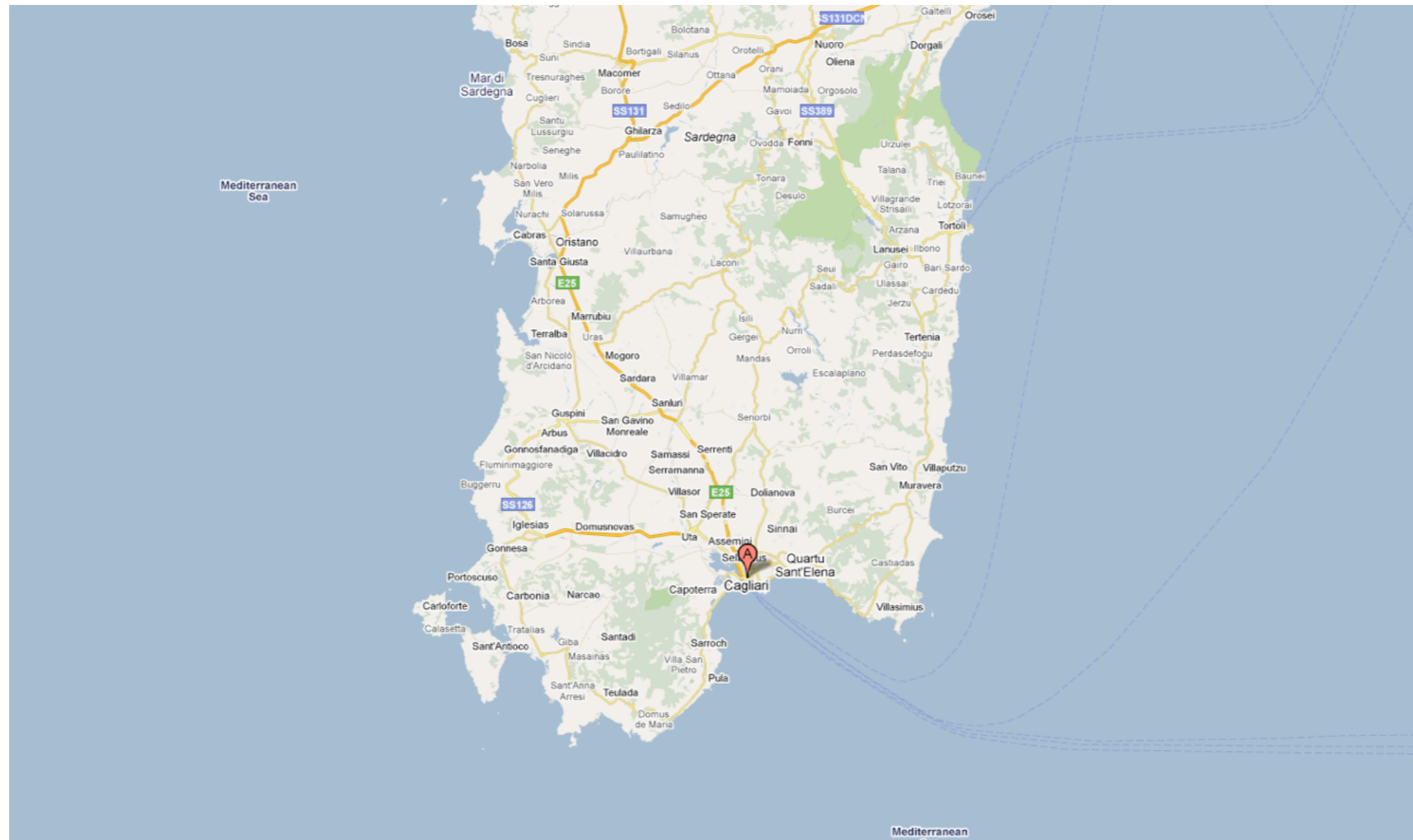
Kurze Vorstellung



Schauen wir zurück



XP 200 I - Cagliari, Sardinien



Revolutionäre Ansätze

Pair Programming
Kurze Release-Zyklen
Erst die Tests, dann...
usw.

Wasserfall
Rational Unified Process
Chaos

Heute?
Agil ist Mainstream

Aber:

Wasserfall ist nicht totzukriegen

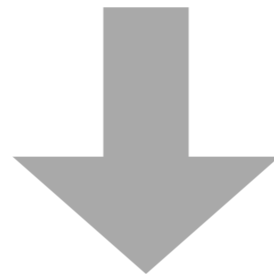
Heute:

Viele meinen, Sie wären agil
Wenige sind es wirklich

Schwierige Veränderungen

Scrum nach dem Buch

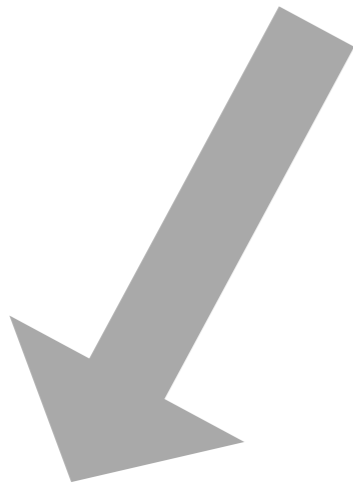
Geht nicht immer, ok...



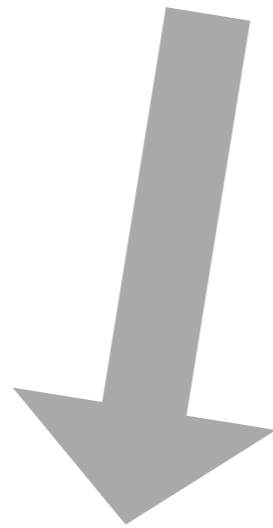
Sit back and relax?

Grenzen in Frage stellen!!

Grenzen in Frage stellen!!!



Mut



**Durchhalte-
vermögen**



Kreativität



Hilfe

Fokus

Ich erinnere mich...

Agile Werte
Agile Prinzipien

Laufende Software **&** **Shipping**

**„What you don't ship,
doesn't exist“**

**„Nur laufende Software
liefert echtes Feedback“**

**Laufende Software hilft,
Risiko zu reduzieren**

Ein wichtiges Ergebnis agiler Softwareentwicklung:
**„Wir konzentrieren uns wieder darauf,
laufende Software auszuliefern, die
Bedürfnisse befriedigt und echten
Mehrwert liefert“**

Frequent Shipping Short Releases

Eine sehr nützliche Frage:
**„Was ist nötig, damit wir
täglich ausliefern können?“**

Separate QA?
Keine Chance!

Separate Operator?
Separate Betriebsabteilung?
Keine Chance!

Kein kleinschrittiges Vorgehen?
Keine Chance!

Keine Continuous Integration?
Keine Chance!

Die nächsten wichtigen Fragen?

Wie lang ist mein Release-Zyklus heute?
Was hindert mich daran, ihn zu zehnteln?

Klingt abstrus?
Stimmt...

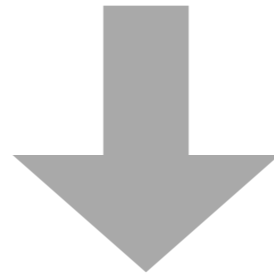
Klingt abstrus?

Stimmt...



**Aber es macht klar,
woran ich arbeiten muss**

**Aber es macht klar,
woran ich arbeiten muss**



**Alles weitere ergibt
sich aus dieser Frage**

**Es ist die Software,
die im Vordergrund stehen sollte**

nicht Dokumente, Vorschriften, Blockaden, Vorgesetzte,
Karrieren, existierende Strukturen, ...



Q&A

Martin Lippert
mlippert@vmware.com