XP 2013 VIENNA

MULTI-PROJECT MANAGEMENT WITH DISPERSED AGILE TEAMS

Delivering safety-critical applications for air traffic control to multiple parallel customers out of a distributed workforce

2013-06-05 | Maximilian Hantsch-Köller

FREQUENTIS

→ Multi-project management with dispersed agile teams

How do you deliver safety-critical applications for air traffic control to multiple parallel customer projects out of a distributed workforce?

Where can agile methodology help?

Two years ago, Frequentis introduced agile development methods to coordinate projects and product development for air traffic management safety-critical applications out of dispersed and distributed teams in two countries. This talk presents the approach used, questions that arose, decisions taken, the observed results, impact on team motivation, and lessons learned throughout the process. Witness the transformation of a waterfall-driven business into an increasingly agile organization!

→ Agenda

- → The Company
- → The Environment (Safety first!)
  - regulations, standards, waterfall processes…
- → Challenges
- → Possible Approaches
- → Frequentis‘ Choice
- → Daily Routine
- → Observed Results & Lessons Learned

COMMUNICATION AND
INFORMATION SOLUTIONS
FOR A SAFER WORLD

The Company

FREQUENTIS

More than 65 years of innovation & expertise in mission critical applications

We develop and market high reliable communication and information systems for mission critical applications in the fields of Air Traffic Management and Public Safety & Transport.

ATM
Air Traffic Management

PST
Public Safety & Transport





ATM Civil | **Defence** | **Public Safety** | **Public Transport** | **Maritime**

Worldwide Control Centres develop towards the same standards.

→ Company Overview

- → Established in 1947
- → 183 Mio. EUR Turnover 2012
- → Corporate headquarters in Vienna
  - – Subsidiaries and regional offices in over 50 countries
- → about 1,100 Employees
- → Outstanding Engineering Capacity
  - – more than 600 highly-qualified engineers (HW/SW/PM) at FREQUENTIS headquarter and subsidiaries
- → Export Quota > 90%
- → R&D Quota > 12%

First Air Traffic Control System in Austria, Vienna / Schwechat, 1955

Breakthrough in the US: FAA Command Centre / Herndon, Virginia, 2003

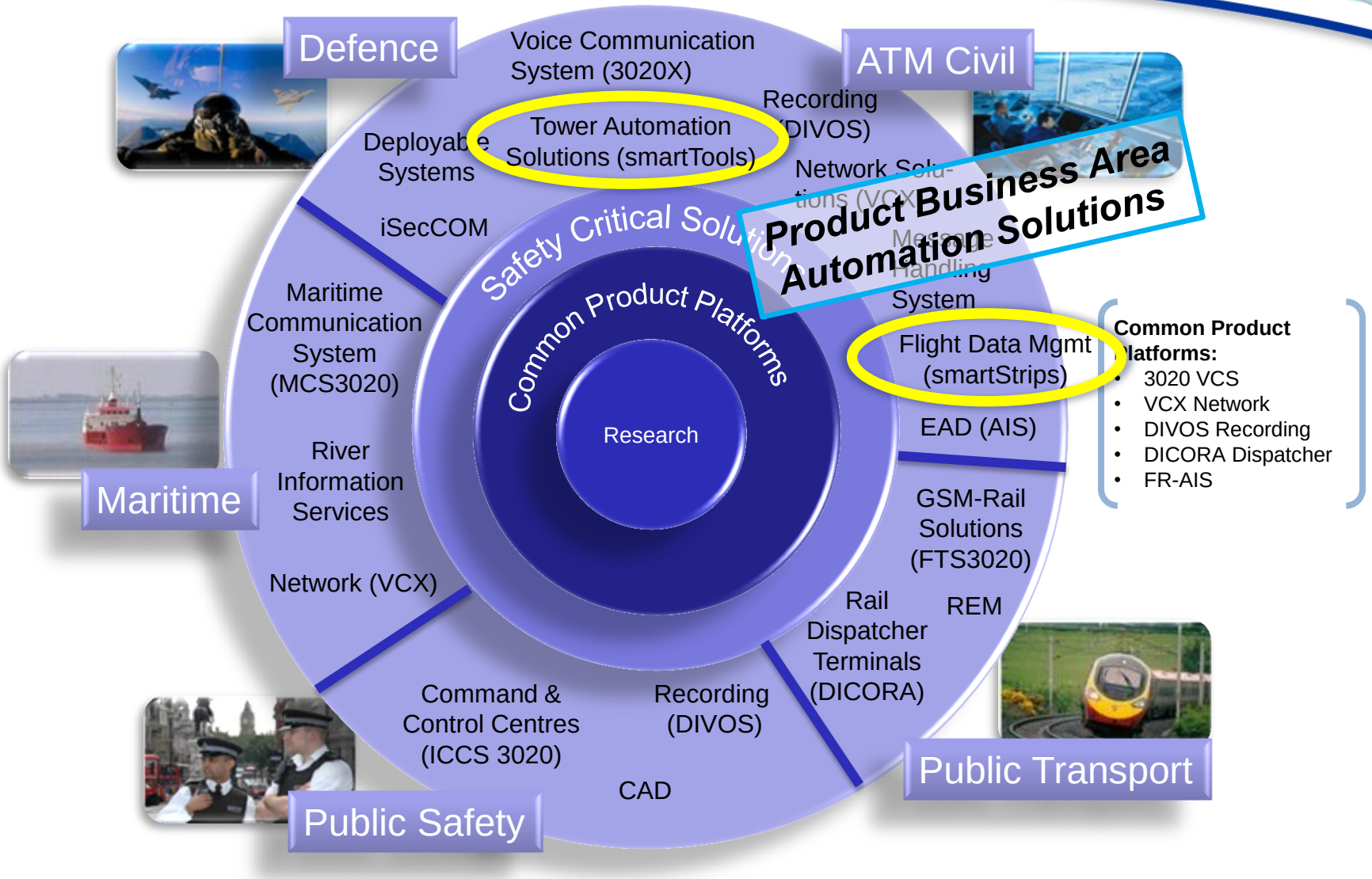
Company Headquarters on Wienerberg, relocation in 2006

Global Market Leader in ATC Voice Communication Systems

→ Worldwide References

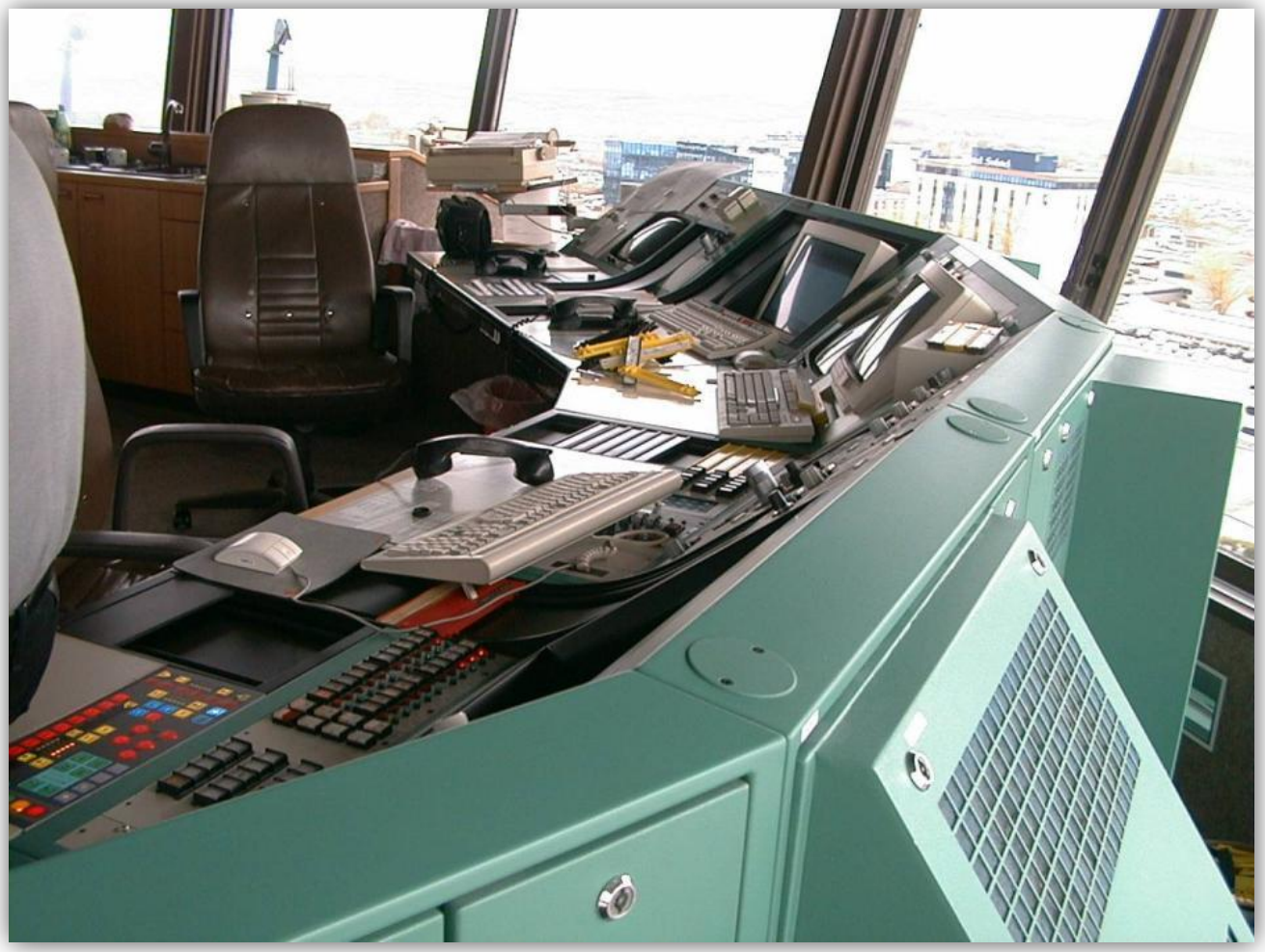
[Excerpt 05/2013]

→ Frequentis Product Portfolio

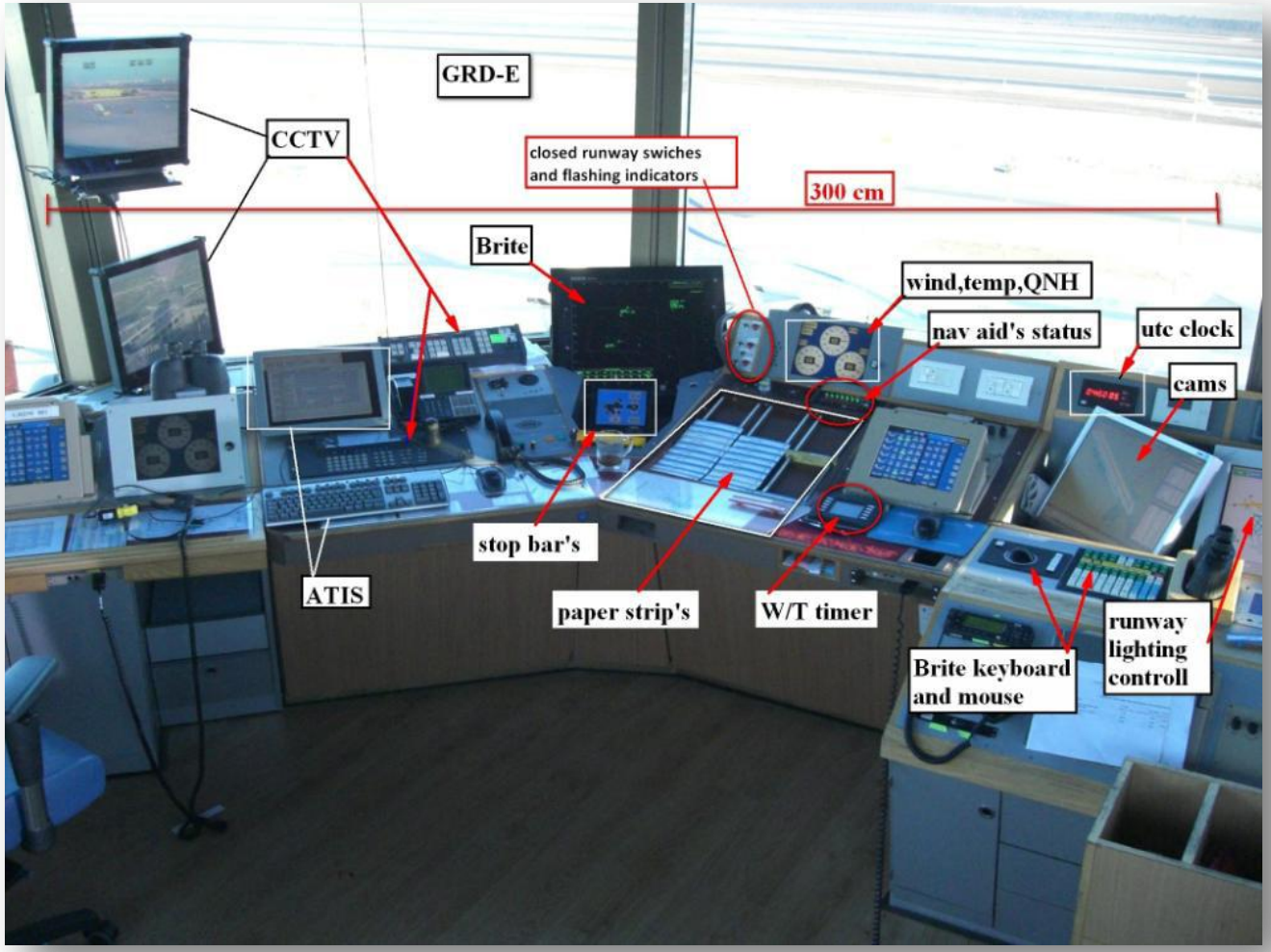


→ What do these systems look like?

→ Legacy Tower



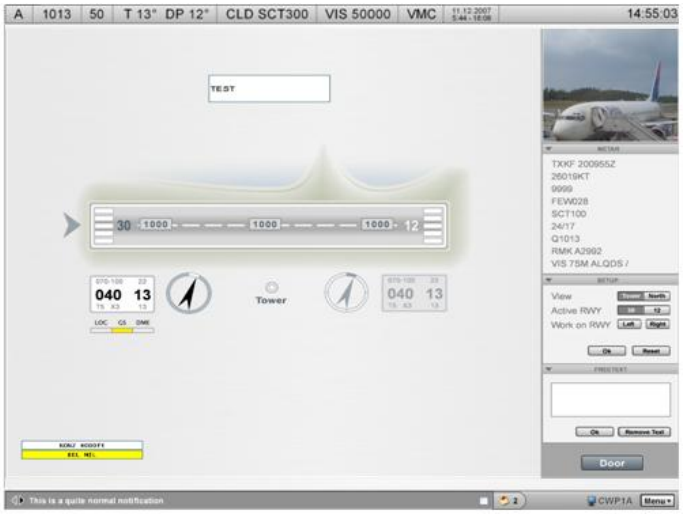
→ Tower Working Position today

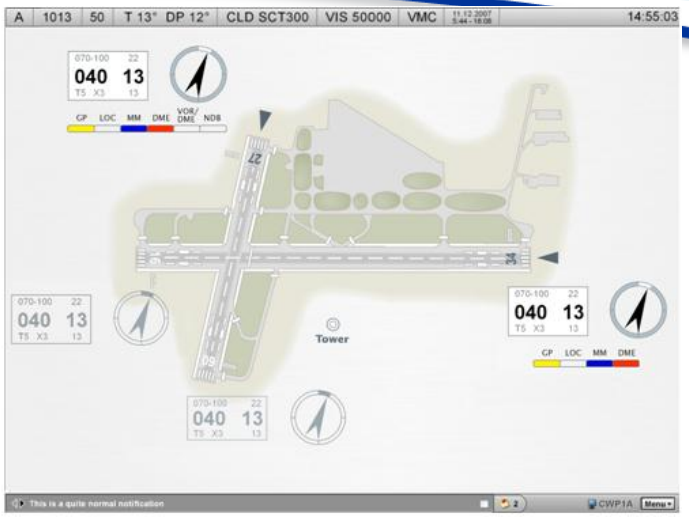


→ smartTools: MET, NAV, ATIS, LICOS

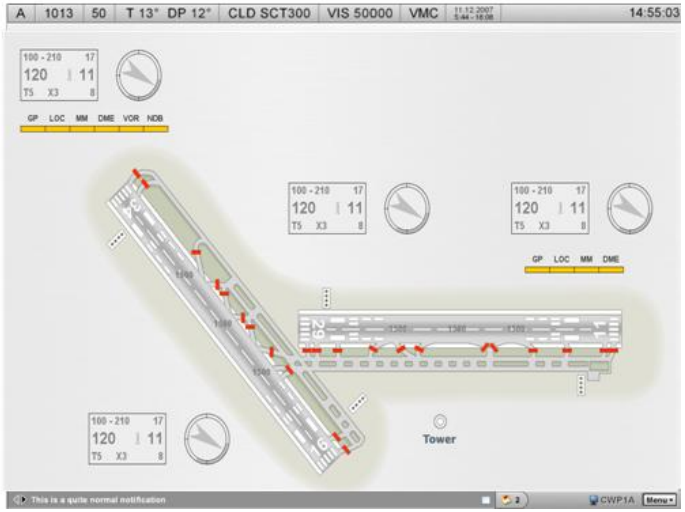
- → presenting all relevant data from an airport
- → data are display in different formats graphical / numerical oriented (including animations)
- → customization to local situation / specific airfield layouts
- → highlighting of critical information
  - RWY Block
  - Wind Thresholds – RVR Values
  - critical value changes
- → indication of outdated values

→ Sample Screens





Customized HMI for each Project!



→ smartStrips Basics

- → Electronic Flight Strip System for APRON, ACC, APP, TWR
  - → Efficient and highly usable management of flight data
  - → Strip printer replacement / Standalone system without FDPS connection
  - → Bi-directional FDPS interface

- → Focus on
  - → Functionality
  - → Usability
  - → Redundancy
  - → Safety

→ Control Room Design: Today and Future

Individual

Technique

Team

Organisation

Environment

Lowering of:
- Costs
- Modifications

Improvement of:
- Workflow
- Teamwork
- Organisation
- Communication

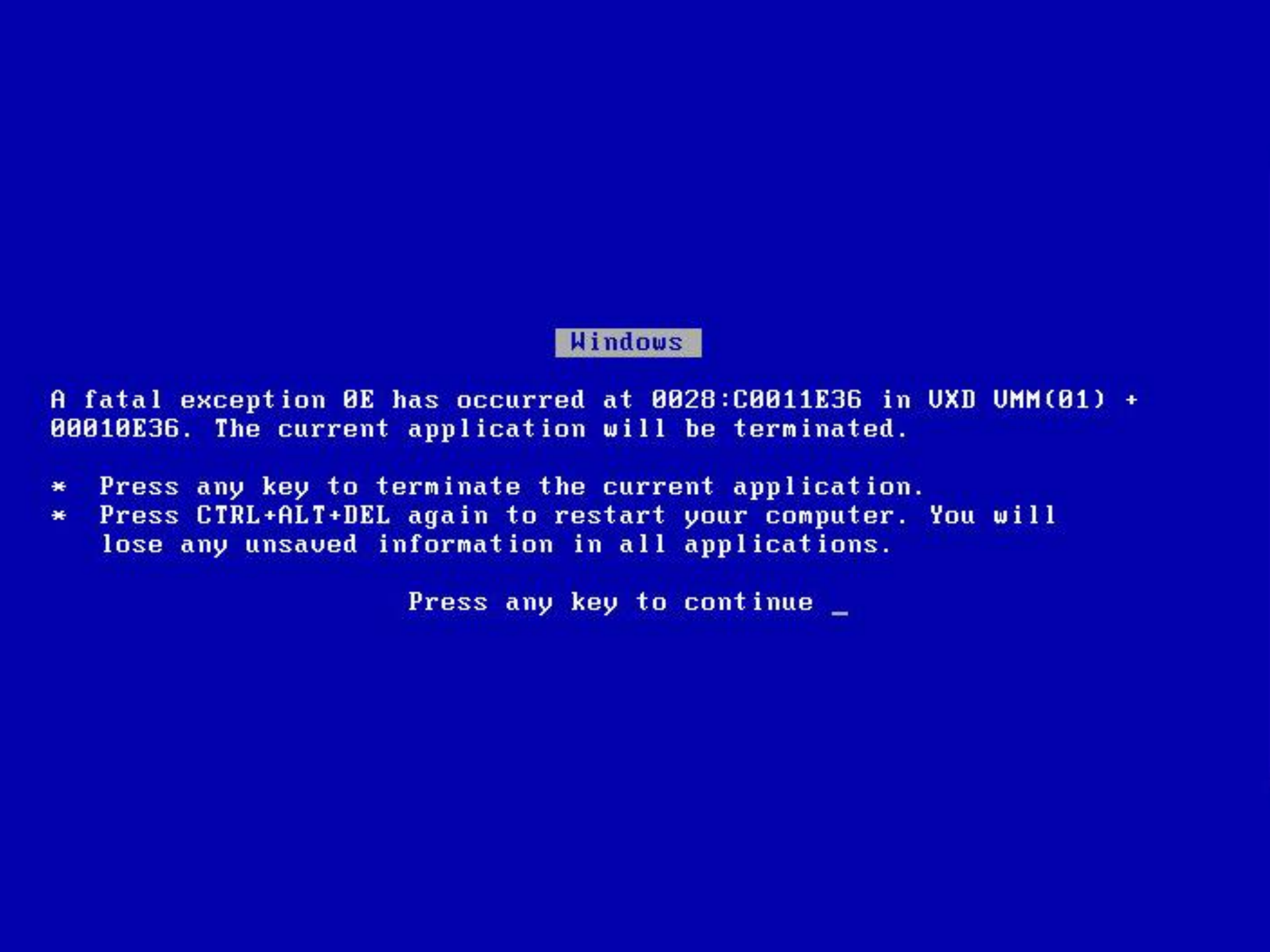
Enhancement of:
- Safety
- Dependability
- Productivity
- Job satisfaction

The Environment

FREQUENTIS

→ Safety First

- → **Airplanes cannot stop** in mid-air
- → There are **Human Lives** involved
- → **Aeronautical Services** must be provided **at all times**
- → **Data must be accurate**


Windows
A fatal exception 0E has occurred at 0028:C0011E36 in VXD VMM(01) +
00010E36. The current application will be terminated.
* Press any key to terminate the current application.
* Press CTRL+ALT+DEL again to restart your computer. You will
lose any unsaved information in all applications.
Press any key to continue _

→ High Level Requirement Examples

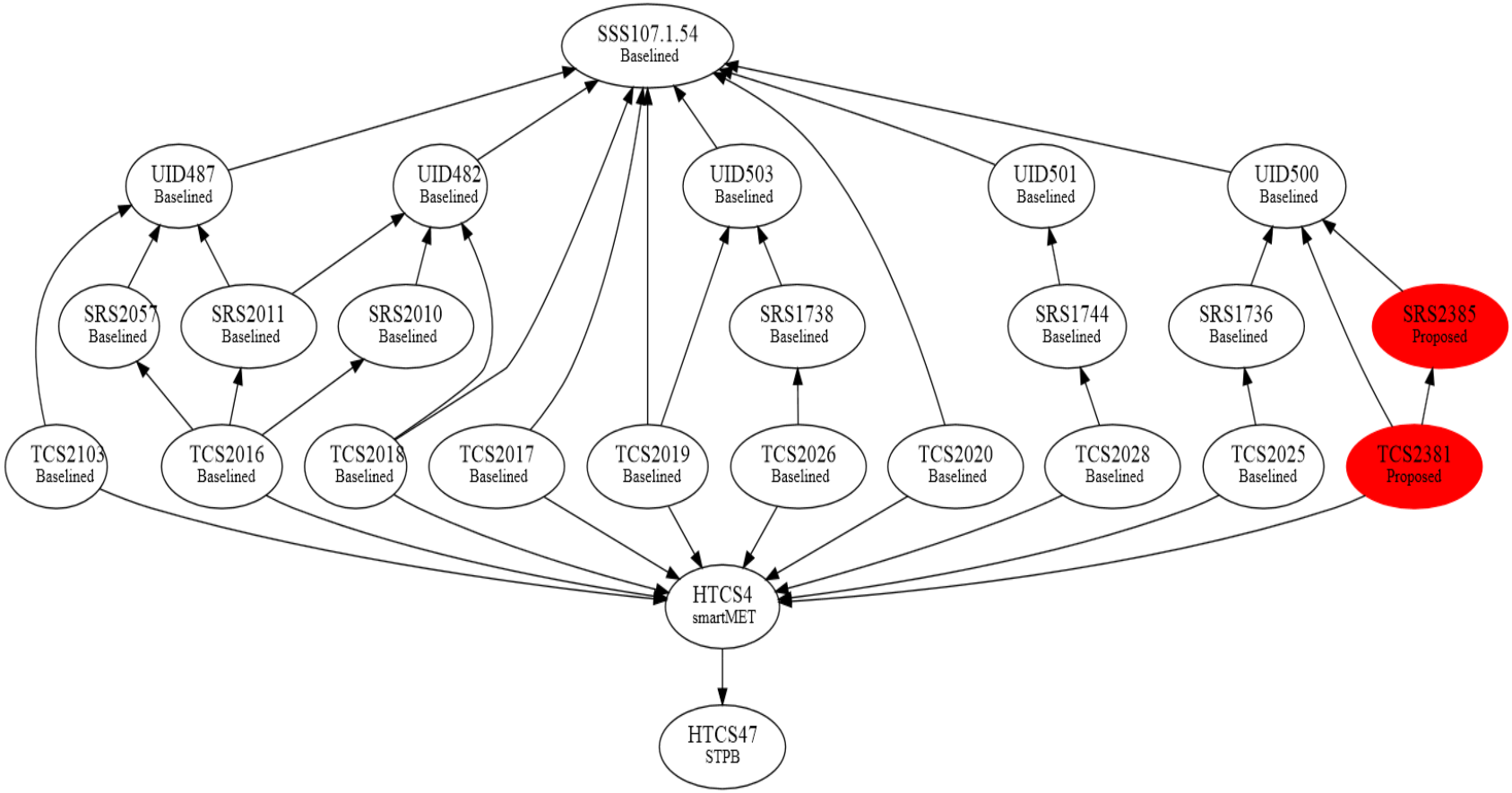
- → Failover and Redundancy:
  - In case of a single server failure, the failover to the other standby server shall be done in less than ten seconds for 95% of the failovers.
  - During recovery from any failure mode the system performance shall stay within the acceptable limits indicated in Appendix A.
  - During failure of all servers, the clients shall remain functional in a standalone mode.
  - **When changing a position due to a working position failure, the activation of the role, elsewhere, shall take effect within 2 seconds at an already running position for 95% of all changes and not longer than 4 seconds.**
- → User Interface
  - The response time on user actions shall conform to targets in Appendix A Table A-1 'Performance Target Response Times', during operation within the defined system capacity.
- → Other
  - No single points of failure
  - 24x7 operation – no „maintenance downtime“

→ Software Safety Assurance Standards

- → **DO-278/ED-109 Software Standard for Non-Airborne Systems**
- → **ED-153 - Guidelines for ANS[1] Software Safety Assurance**
  - – both provide guidelines for the assurance of software contained in non-airborne CNS/ATM[2] systems
  - – depending on the impact of a software failure, different levels of assurance
  - – tracing from high level requirements to low level requirements to code coverage / statement coverage / decision coverage (depending on assurance level) and test cases (system / software / component / unit)
- → **massive control structure around the software development process**
  - – **documentation & reviews at every stage**
  - – yields towards waterfall process with requirements engineering -> system design -> software design -> code implementation -> tests
  - – **auditable proof that sound engineering practices have been followed**

[1] ANS: air navigation services
[2] CNS/ATM: communication, navigation and surveillance / air traffic management

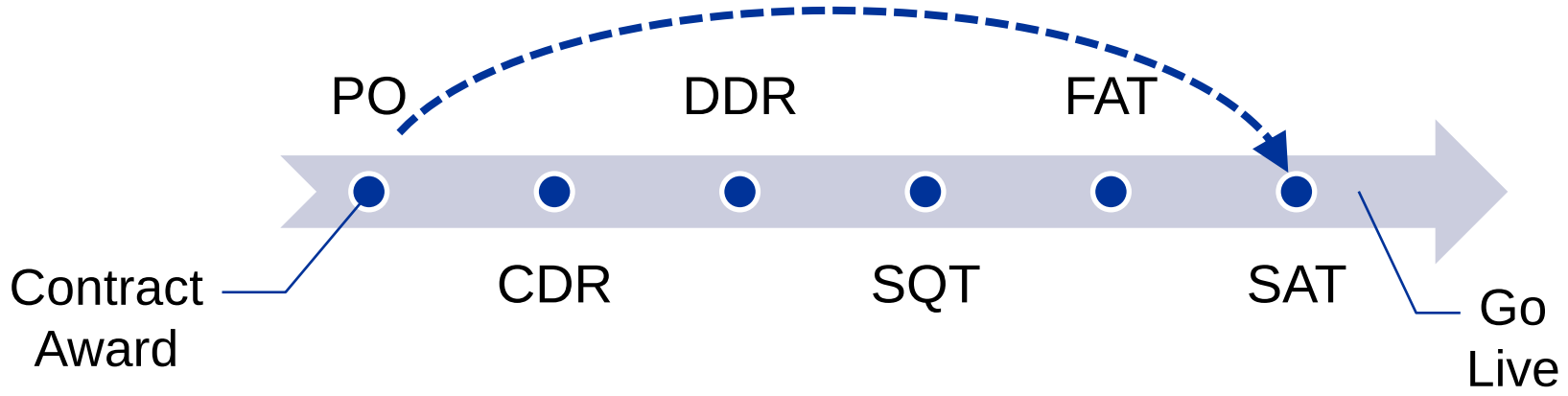
→ Requirement Traces



Challenges

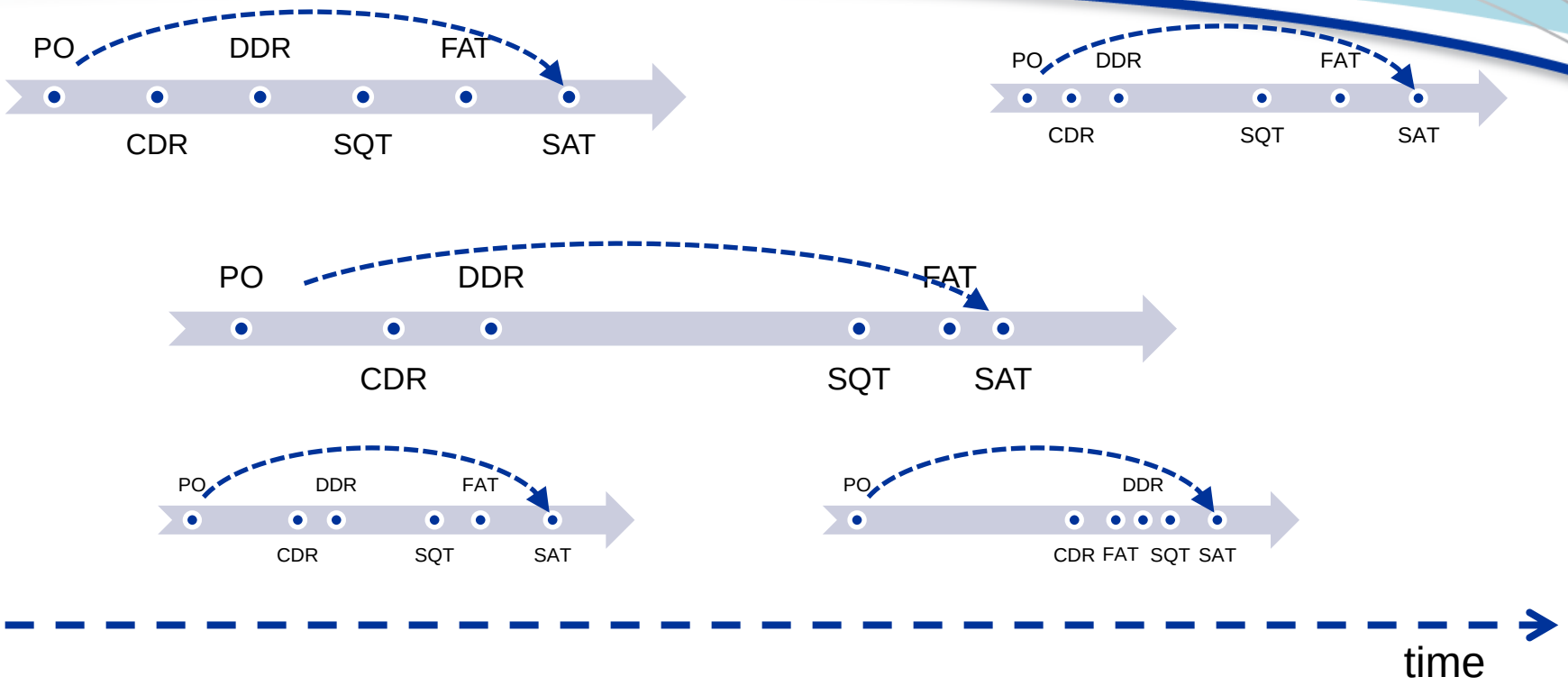
→ Challenges

- → Software Assurance is demanded by regulatives, and as a result, by our customers
- → Waterfall timeline of a typical project

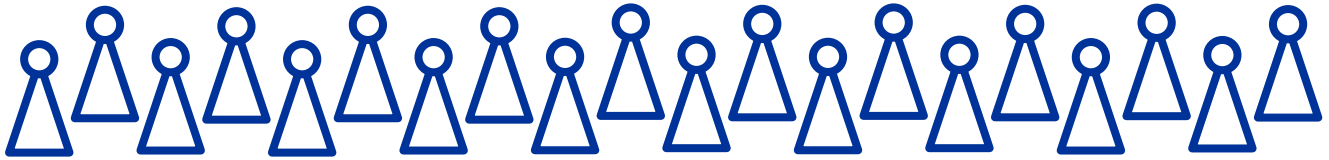


- → Multiple deliveries / parallel projects (out of product core)
- → Growing teams in several locations

→ Multiple Parallel Projects



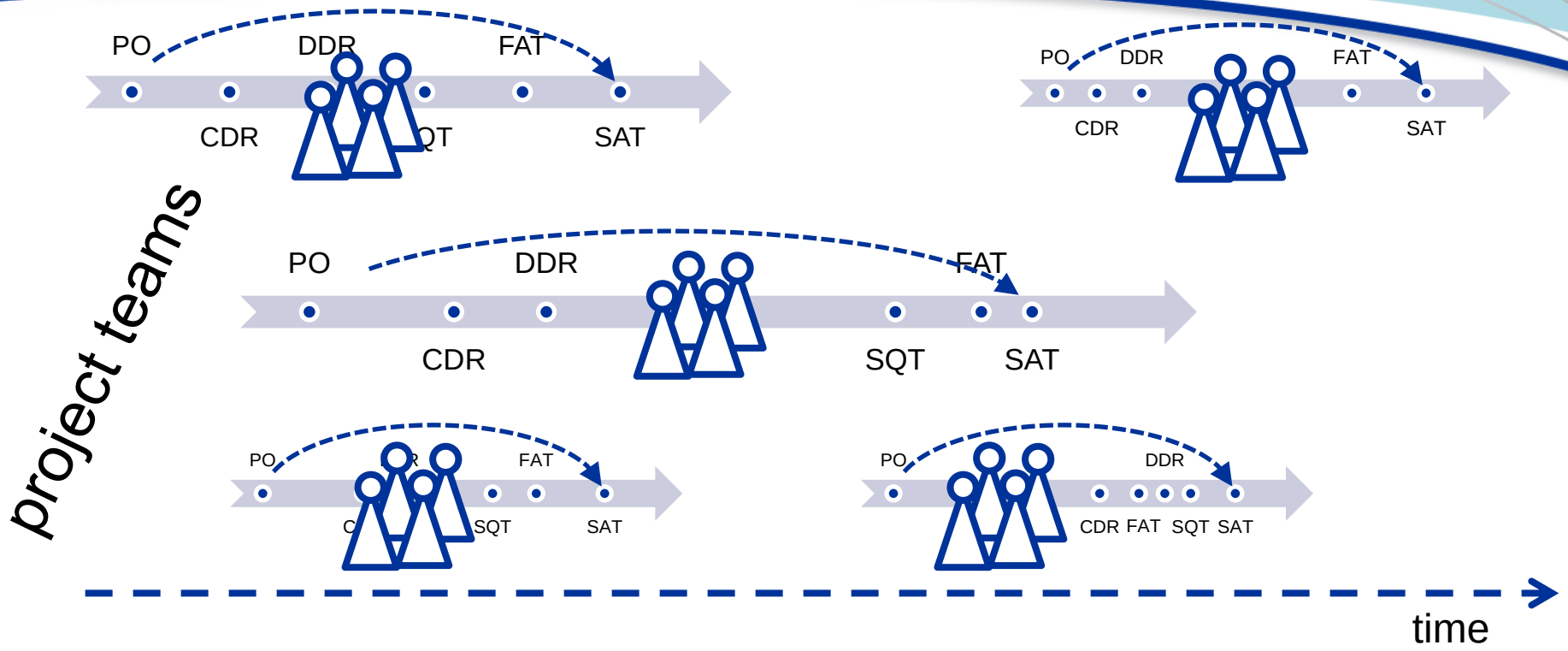
the teams



Vienna

Romania

→ Multiple Parallel Projects



software is made by people

software architecture follows people organisation

software branches, de-couple everything

→ De-coupled projects?

- De-coupled projects from each other
  - Separate Timelines
  - Separate Project Teams
  - Separate Software Branches

- Optimal team size? Move people around when priorities change?
- Different skill sets in different locations?
- Lifetime of these teams? Re-shuffling for next project?
- Moving know-how from all these projects to your product base?
- **How to keep de-coupled stuff together?**

Possible Approaches

FREQUENTIS

→ Agile Recommendations

- → Teams: 7 ± 2
- → Co-locate teams in one location
- → Build functional teams, not component teams
- → Ensure teams have all skills needed to deliver end-to-end
- → Invest in Story Writing, build and prioritise a Product Backlog
- → Deliver Often and Early
- → If It Is Hard, Do It More Often
- → Learn Frequently and Iteratively
- → Individuals and Interactions over Processes and Tools

→ Co-located, Distributed, Dispersed

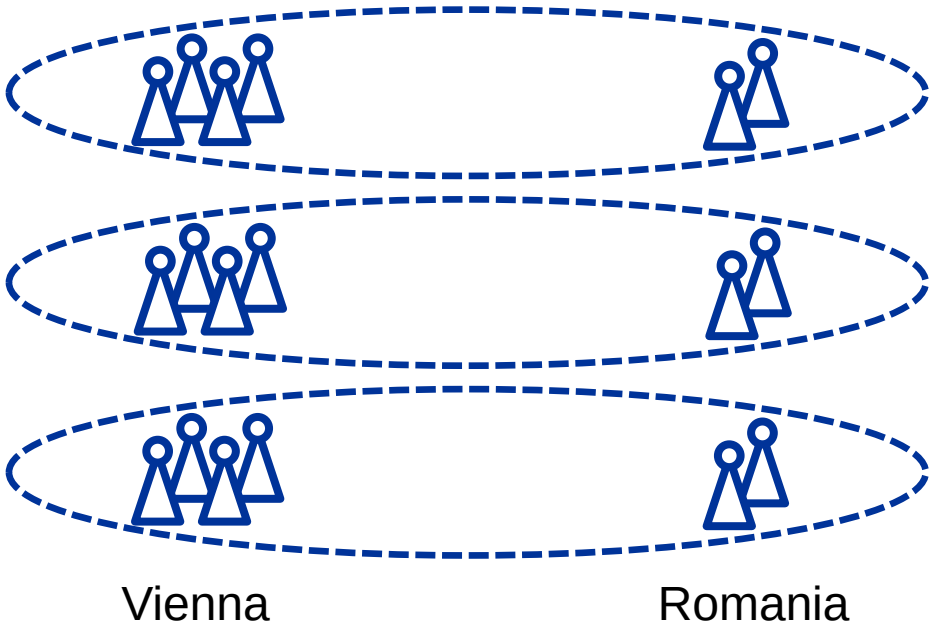
- → **Co-located: everyone is in one location**
- → Distributed: teams are in different locations, but within one team everyone is in one location
- → Dispersed: people within the team are in different locations

→ Co-located, Distributed, Dispersed

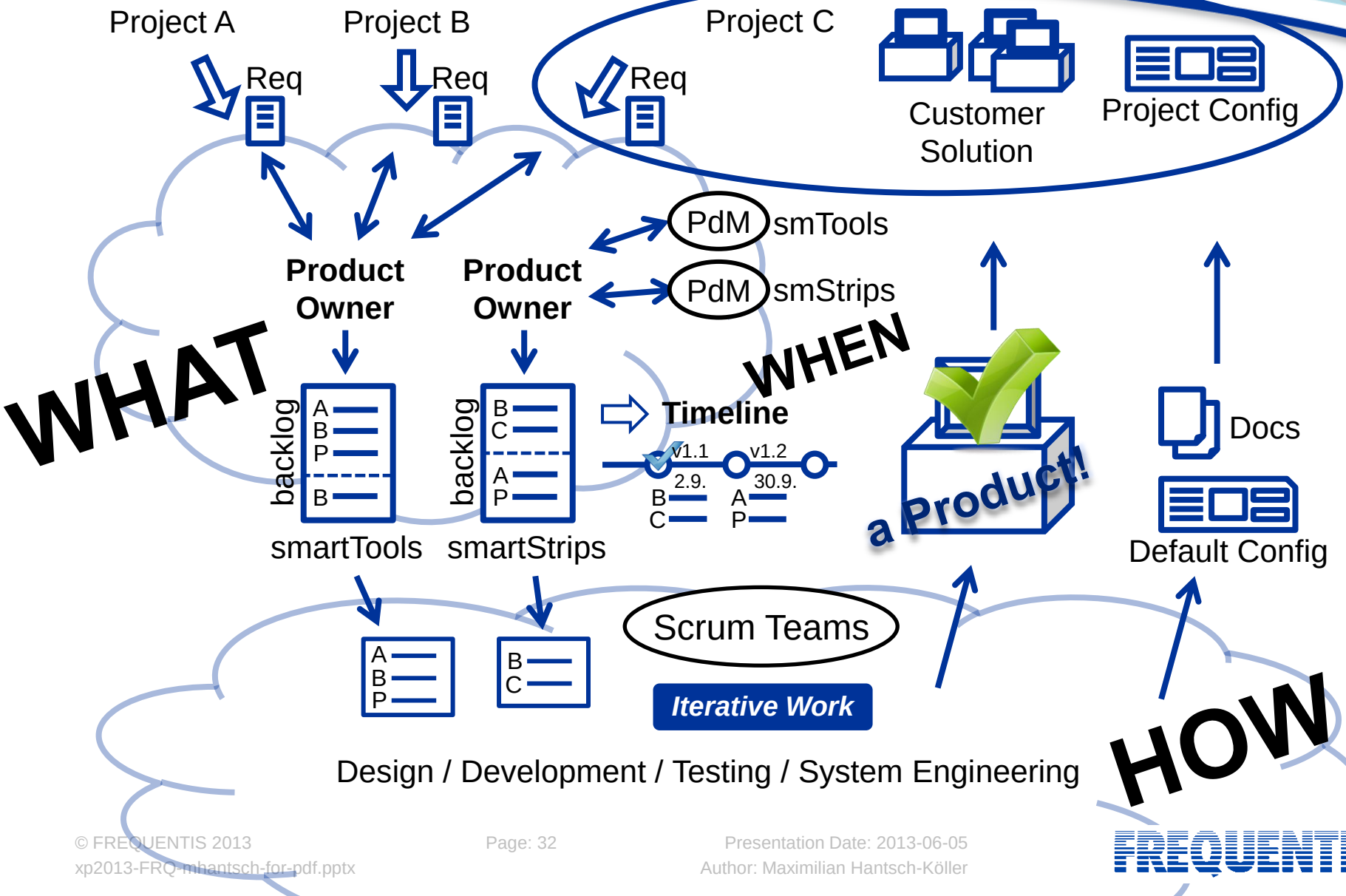
- → Co-located: everyone is in one location
- → **Distributed: teams are in different locations, but within one team everyone is in one location**
- → Dispersed: people within the team are in different locations

→ Co-located, Distributed, Dispersed

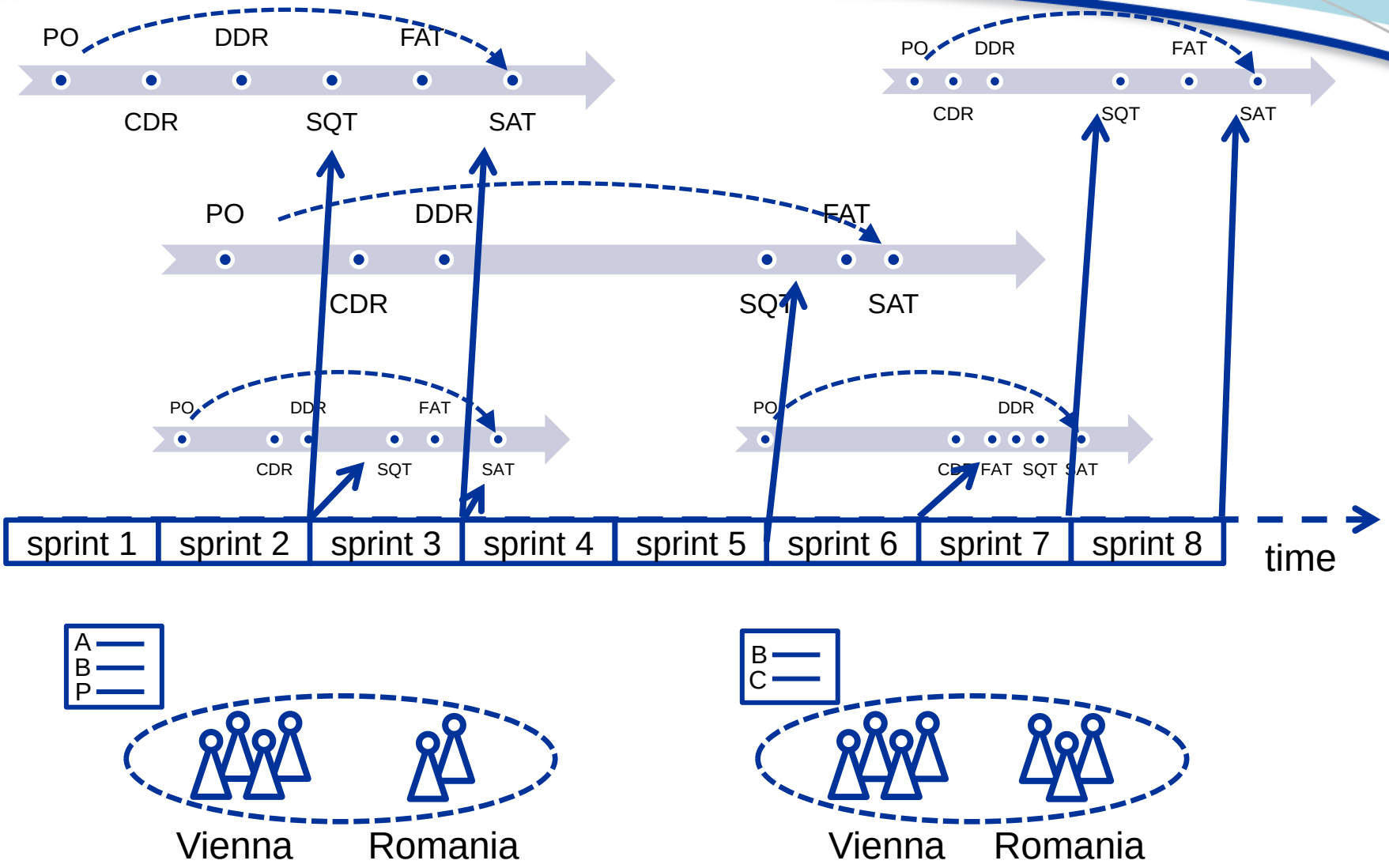
- → Co-located: everyone is in one location
- → Distributed: teams are in different locations, but within one team everyone is in one location
- → **Dispersed: people within the team are in different locations**



→ Build a Product Backlog



→ Back to our Projects



COMMUNICATION AND
INFORMATION SOLUTIONS
FOR A SAFER WORLD

Frequentis‘ Choice

FREQUENTIS

→ What we chose (1)

- → Organise around products: Product Owners, Product Backlog
  - interface projects to the development teams through product backlog
  - learn to write stories
  - product owner acts as gatekeeper
- → prioritise by reordering backlog items instead of reordering people
- → transparent resource planning based on velocity and story point estimations
- → drive active participation of projects
  - invest in communication

→ What we chose (2)

- → Use Scrum to deliver out of Dispersed Teams
  - some skills only available in one location (testing, lead architects)
- → Implications we knew we had to take care of:
  - online collaboration tools, video conferencing
  - scrum trainings, intercultural trainings
  - invest in communication
- → Project Know-how
  - some team members to be assigned as project experts
- → Helpful advice from the community: http://www.infoq.com/presentations/The-Kiev-Experiment

Daily Routine

FREQUENTIS

→ Daily Routine

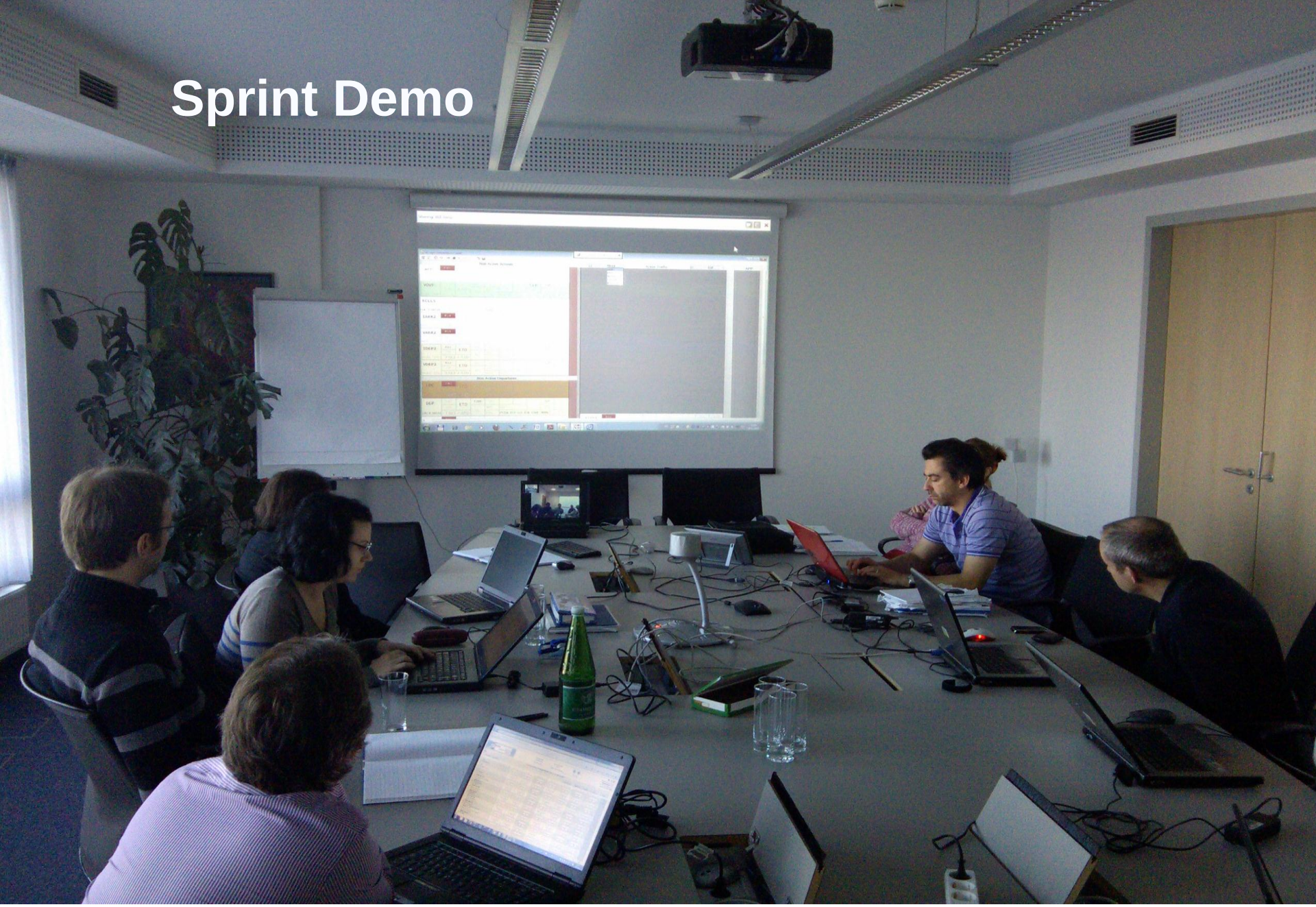
- → Video & Audio conferencing everywhere
- → Online backlog and task boards in Jira / GreenHopper
- → Collaboration Rules
  - e.g. Grooming with at least 3 team members, at least 1 from each location, and 1 with testing background (if possible)
- → Definition of Done
  - e.g. Jenkins builds blue, deployment on test system ok, all traces from requirements to test cases complete and reviewed, all test cases for backlog item successful, documentation updated etc.

→ Daily Routine

- → 4-weeks sprints
- → regular grooming sessions to pre-estimate stories
- → stories include acceptance criteria in Gerkhin language
- → mid-sprint review
- → weekly project briefing

- → Tooling
  - continuous integration / unit tests / junit
  - automated component testing / jbehave
  - continuous deployment
  - automated system tests / jbehave + Frequentis distributed system testing framework

Sprint Demo



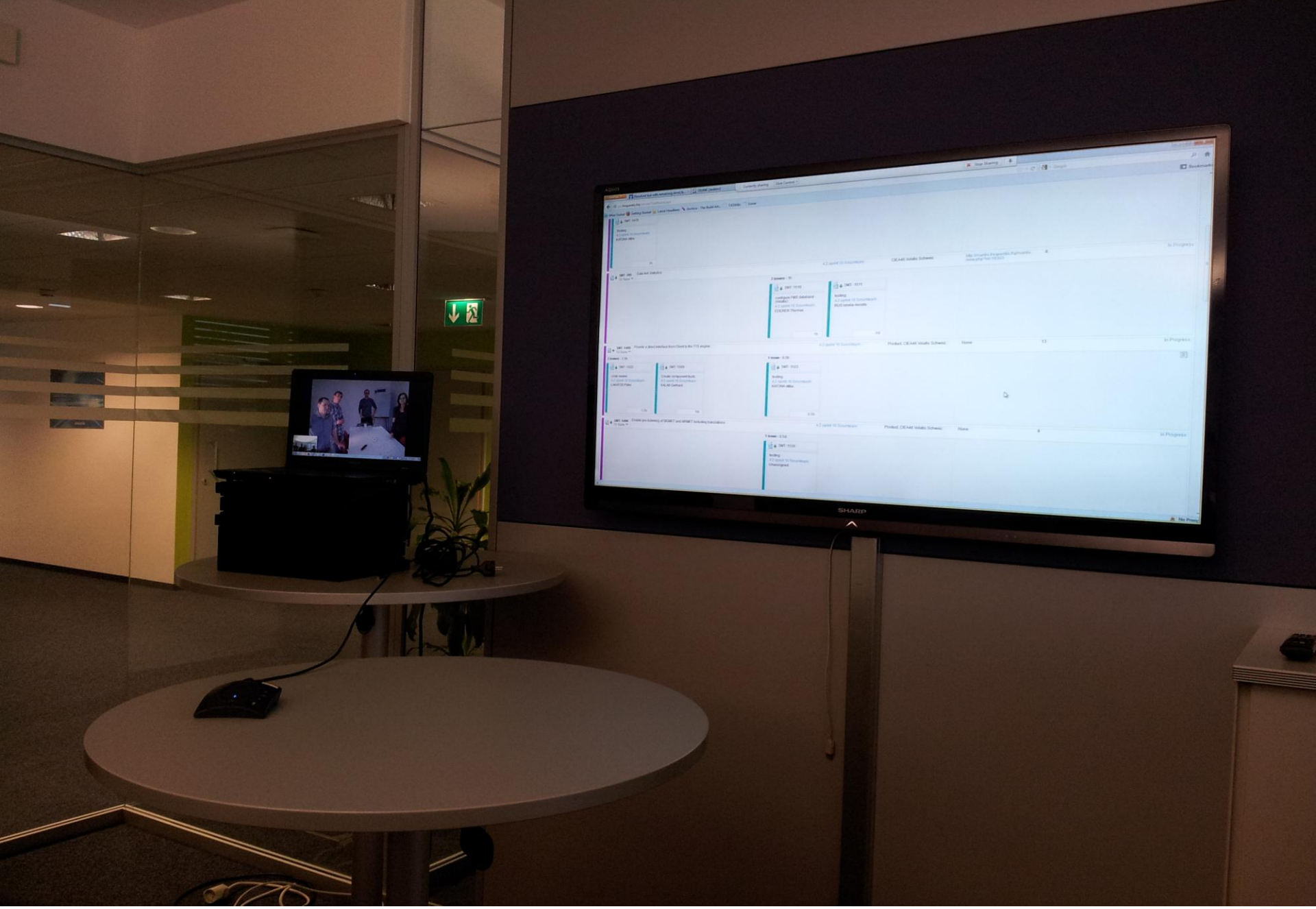
Grooming


Table A5-2 Use of change and time indicators in TAF
APP 5-12
sonny
100% APFELSAFT

→ Daily Standup

→ Daily Standup

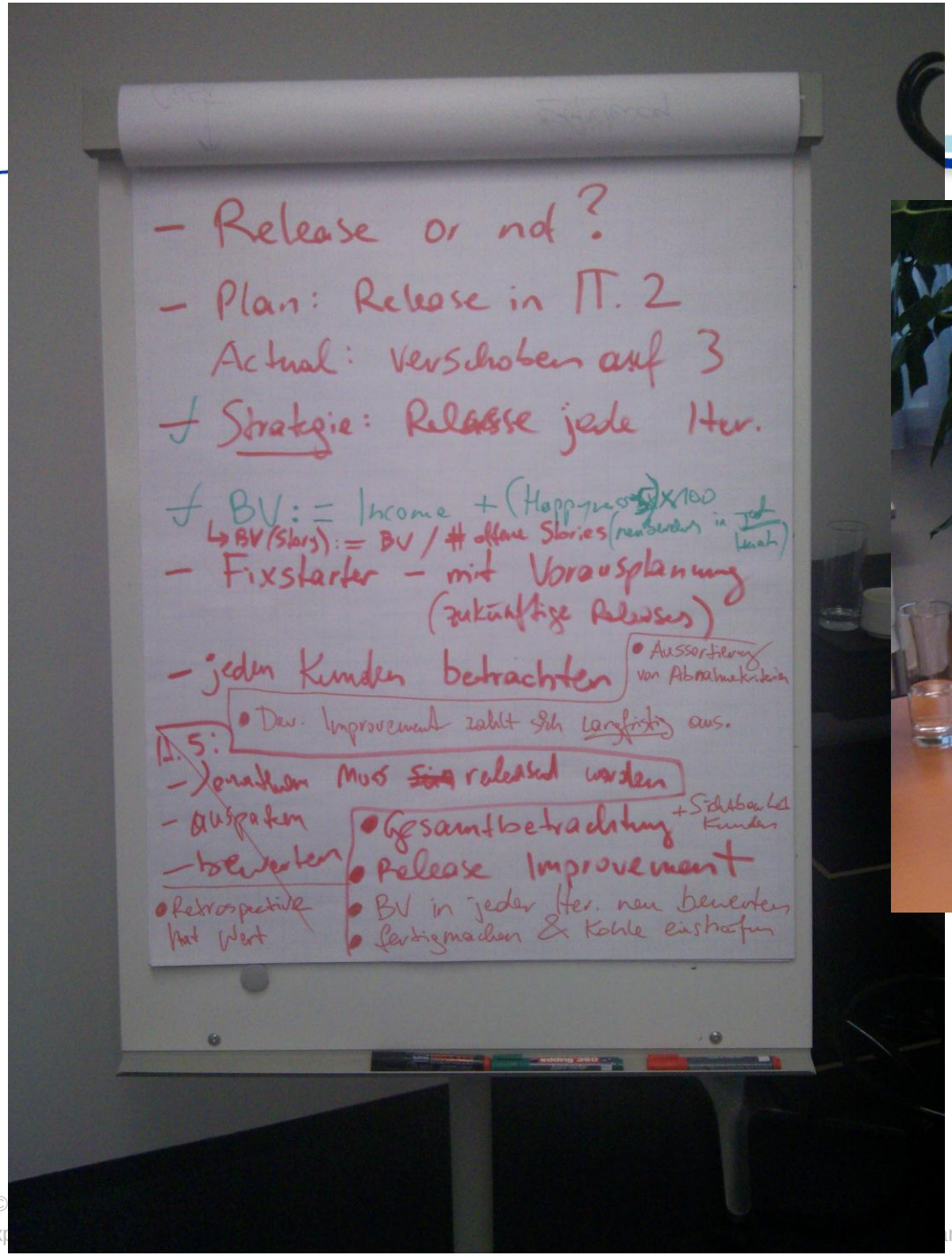
→ Daily Standup


SHARP

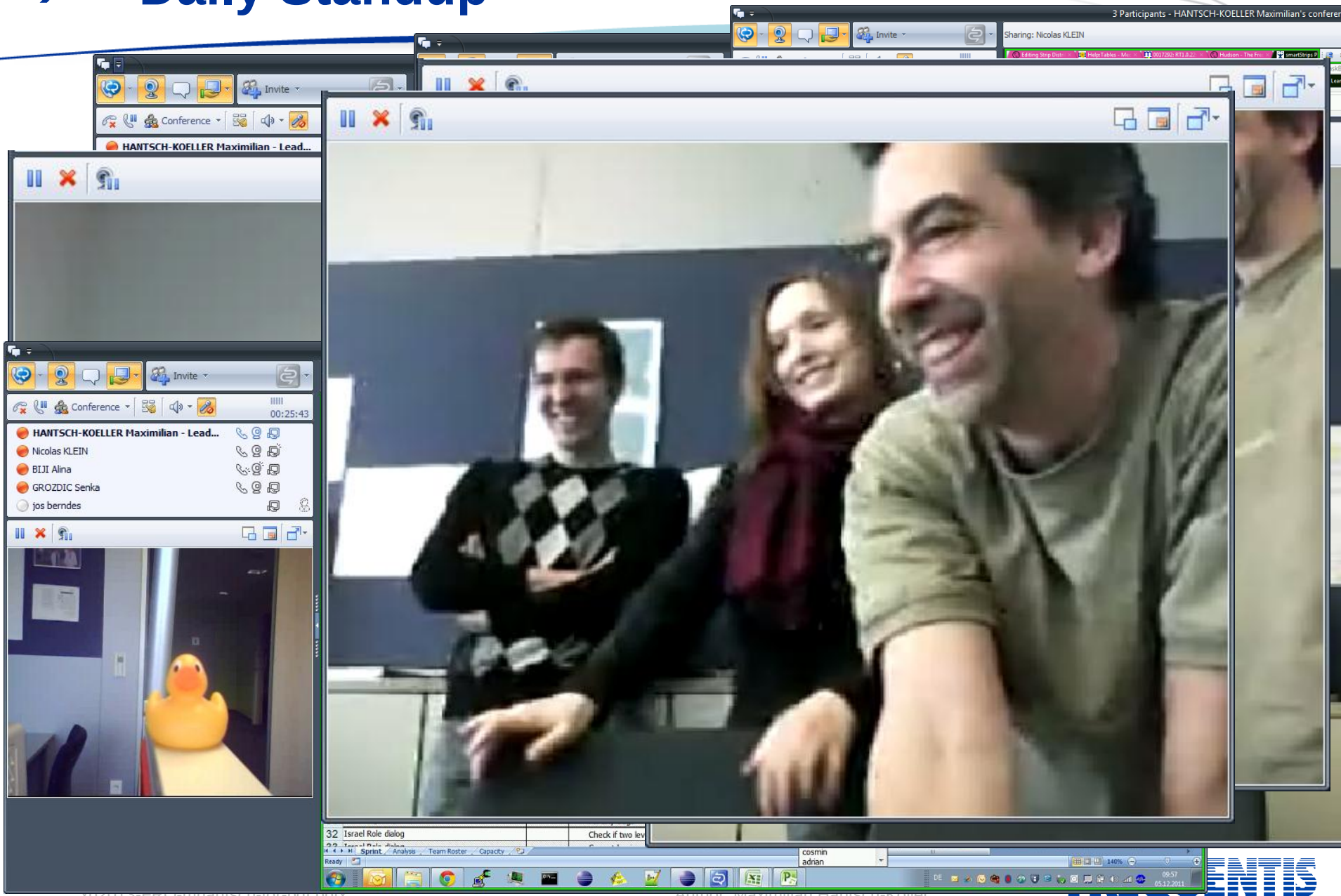
→ Daily Standup

→ Business Value Game (for POs and PdMs)

FREQUENTIS


- Release or nd?
- Plan: Release in IT. 2
Actual: verschoben auf 3
Strategie: Release jede Iter.
BV: = Income + (Happiness x100
↳ BV(Story) = BV / # offene Stories
- Fixstarter - mit Vorausplanung
(zukünftige Releases)
- jeden Kunden betrachten
• Aussortieren von Abnahmekriterien
• Dev. Improvement zahlt sich langfristig aus.
1.5:
- Iteration muss released werden
- auspacken
- bewerten
• Gesamtbetrachtung + Schuldenlast Kunden
• Release Improvement
• BV in jeder Iter. neu bewerten
• fertigmachen & Kohle einstreifen
• Retrospective hat Wert

→ Daily Standup



Observed Results & Lessons Learned

FREQUENTIS

→ Observed Results & Lessons Learned

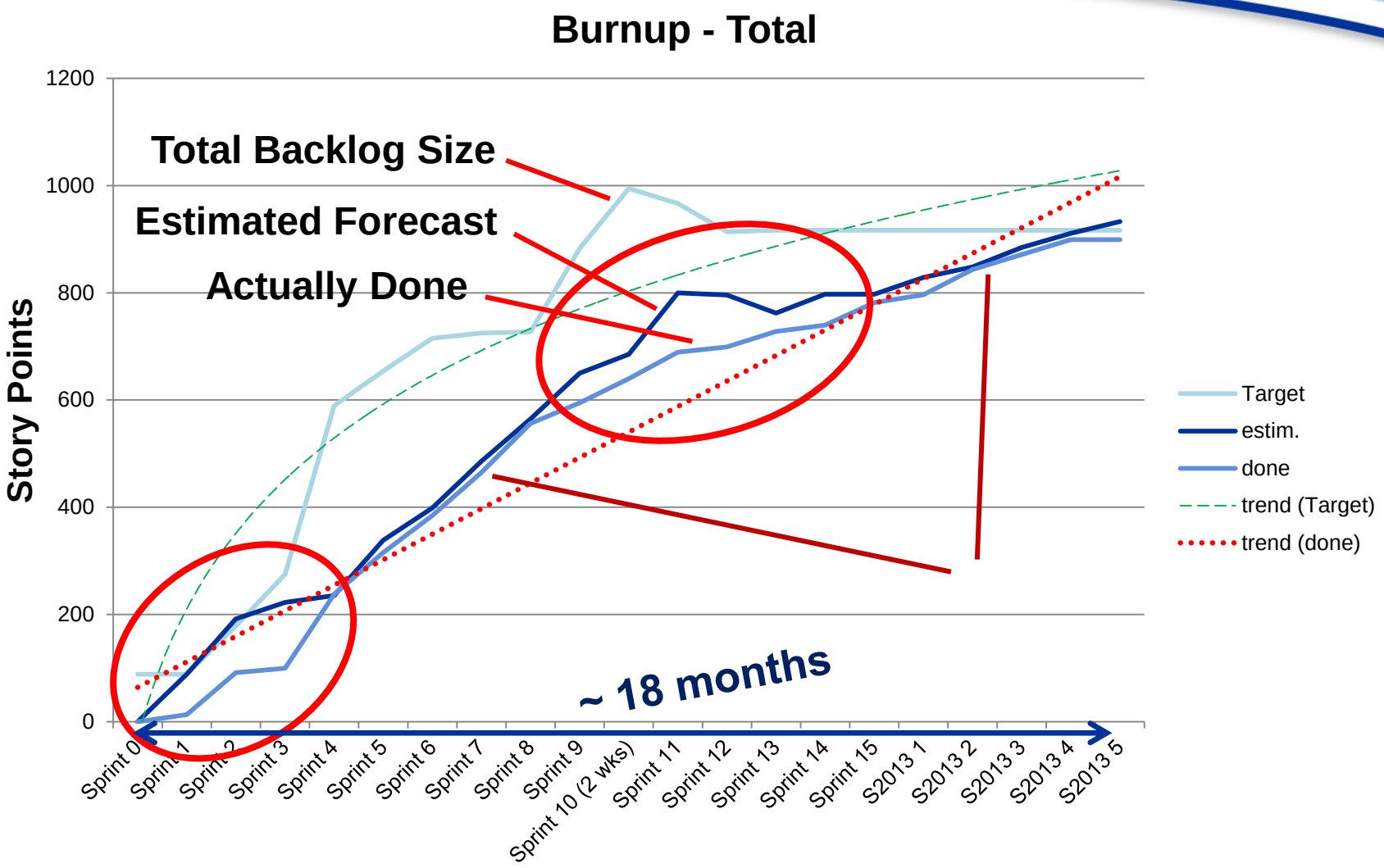
→ Observations

- Teams felt relieved from project priority clashes, they felt more appreciation, more self-control, and more motivation
- Teams learned the value of communication – started to talk to each other much more across borders
- Teams were able to increasingly clean up debt in the product
- Productisation of features increased
- Project timelines became more stable and achievable
  - after initial hick-up ;-)

→ Lessons Learned

- Believe in it and keep going forward. It does get better!
- Crucial to have full support by management. We had it.
- You can deviate from the book. But know why there is a recommendation.
- Stories are not Requirements.
  => Long-term management of product feature documentation?

→ Visibility


Burnup - Total
Total Backlog Size
Estimated Forecast
Actually Done
~ 18 months
Story Points
1200
1000
800
600
400
200
0
Sprint 0
Sprint 1
Sprint 2
Sprint 3
Sprint 4
Sprint 5
Sprint 6
Sprint 7
Sprint 8
Sprint 9
Sprint 10 (2 wks)
Sprint 11
Sprint 12
Sprint 13
Sprint 14
Sprint 15
S2013 1
S2013 2
S2013 3
S2013 4
S2013 5
Target
estim.
done
trend (Target)
trend (done)

→ Next Steps and Challenges

- → Offer calculation from story estimates
- → Expand Automated System Acceptance Tests
- → Share experience and roll-out to other areas of Frequentis
  - 2012: 22 CSM, 6 CPO
  - 2013: 30 CSM, 15 CPO
- → Regular Scrum Master & Product Owner Forums at Frequentis

→ At The Heart of Everything We Do…

COMMUNICATION AND
INFORMATION SOLUTIONS
FOR A SAFER WORLD

Thank you!

FREQUENTIS