



“Agilizing” the Architecture Department

Eltjo R. Poort
SATURN 2015

© 2015 CGI

CGI

Experience the commitment®

Eltjo Poort

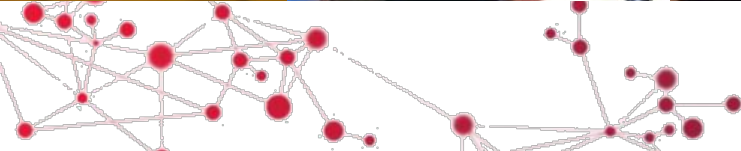
<http://eltjopoort.nl>

CGI Architecture Community of Practice lead

- Reviewing Bids & Projects
- Standardizing & Improving Architecture Practice

Researcher

- Improving Architecture Practices
- With Universities (VU, Twente, Utrecht, Eindhoven)
- Member of IFIP WG 2.10 Software Architecture



RCDA Risk and Cost Driven Architecture

Solution architecting principles and practices based on a view of architecture as a risk and cost management discipline

- Applicable in agile and traditional engagements
- Highly scalable and pragmatic
- Architectural decision making based on economic trade-offs
- Architecture communication in economic terms
- Traceability from requirements to cost



RCDA track record

RCDA results **reported by architects:**

- Increased effectiveness in solution shaping
- More focus on economic reality
- Better communication with business stakeholders
- Better risk and cost control in delivery

RCDA **publications:**

- Journal of Systems and Software
- IEEE Software Magazine (Sept. 2014)
- Leading architecture conferences: SATURN, LAC, WICSA (best paper award)

RCDA a recognized method in the Open Group Certified Architect program.

400 Architects trained

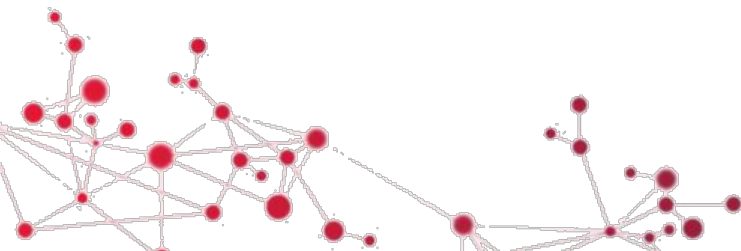
19 Proven practices

300 Pages of guidance

12 Peer-reviewed publications

“RCDA helps us to map out the required quality of IT-based systems in a structured way and provide governance to system development.”

*Klaas van der Ploeg, Architect,
ProRail.*



RCDA Principles of Agile Architecting

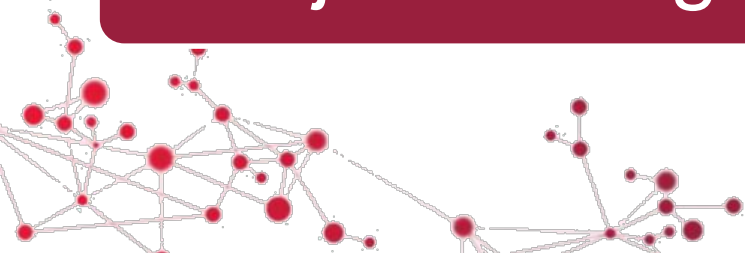
Decisions are your main deliverable

Keep a backlog of architectural concerns

Let economic impact determine your focus

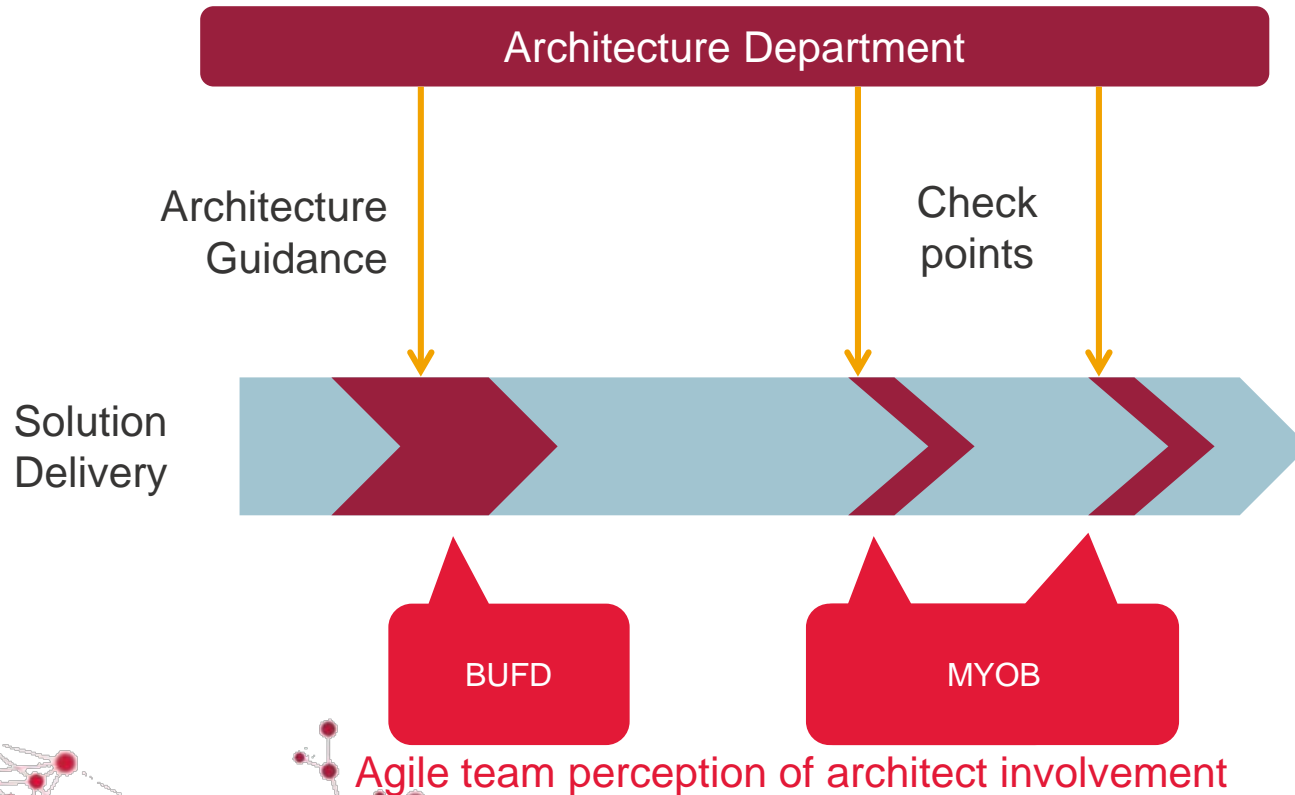
Keep it small

Use just enough anticipation



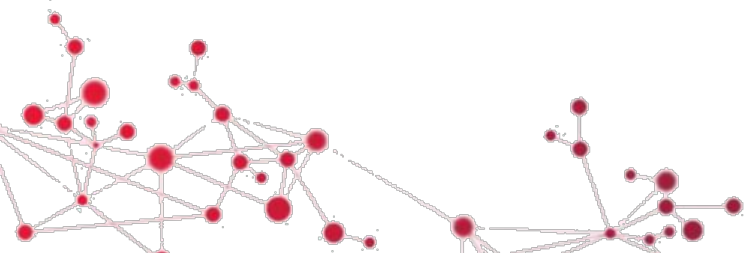
Implementing RCDA: existing context

- Large European transportation infrastructure organization
- IT is mission critical
- Traditional architecture department

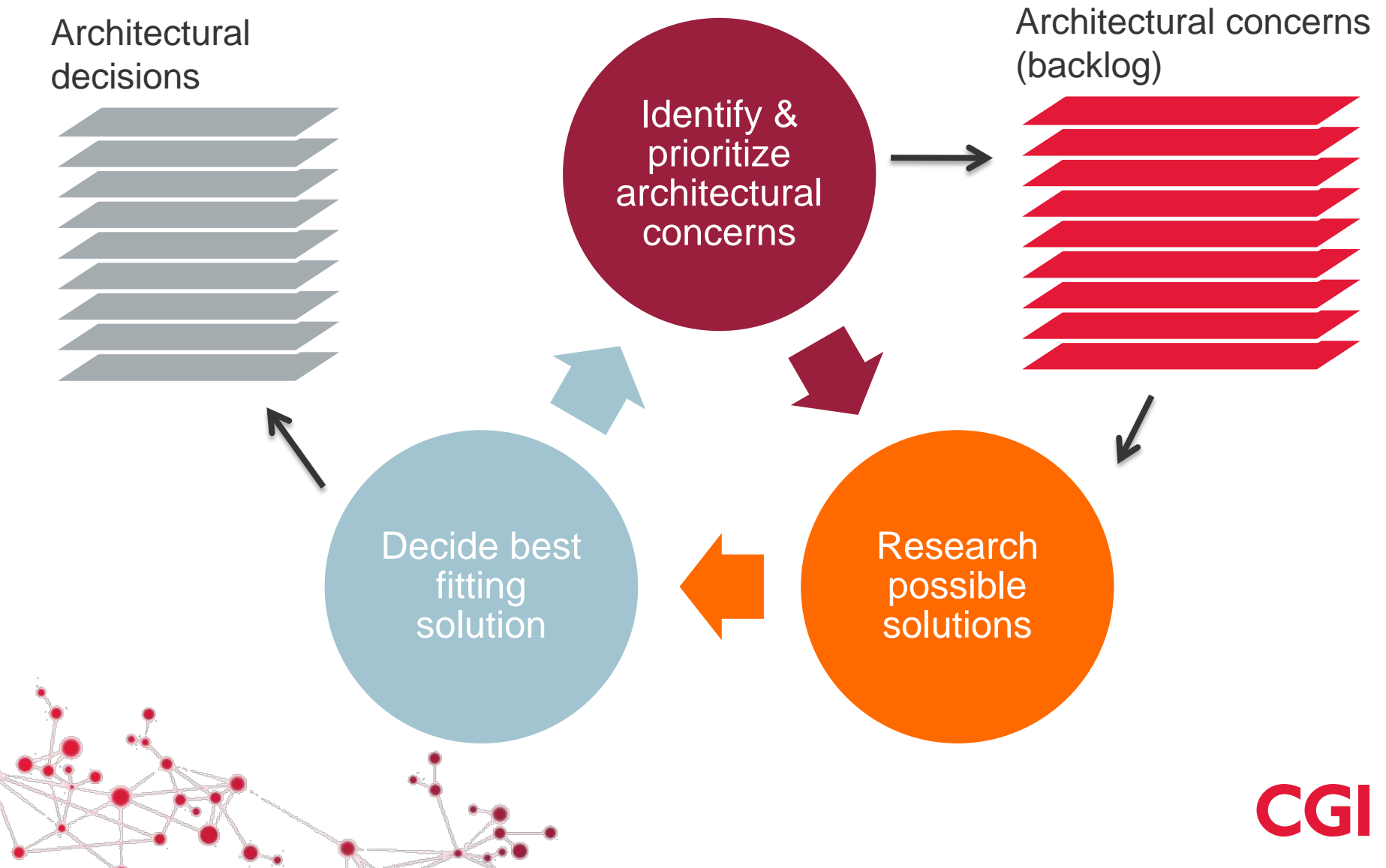


Implementing RCDA: Approach

- Train the architects (and designers,...)
 - 50 architects took 3-day course
 - Focus on owning architectural decisions & economic impact
- Coach architects applying RCDA practices on three running projects
 - 80 hours coaching over 3 months
 - Architectural Requirements Prioritization
 - Architecture Documentation
 - Architecture Roadmapping
- Advice on implementing RCDA practices in knowledge systems

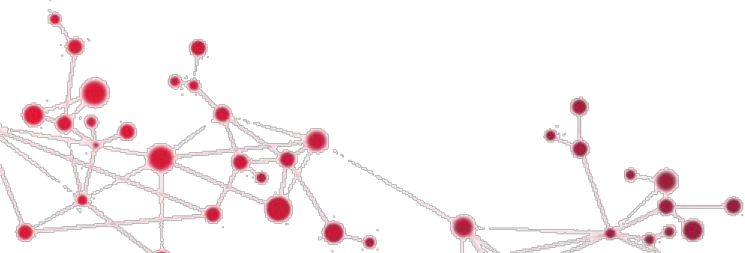
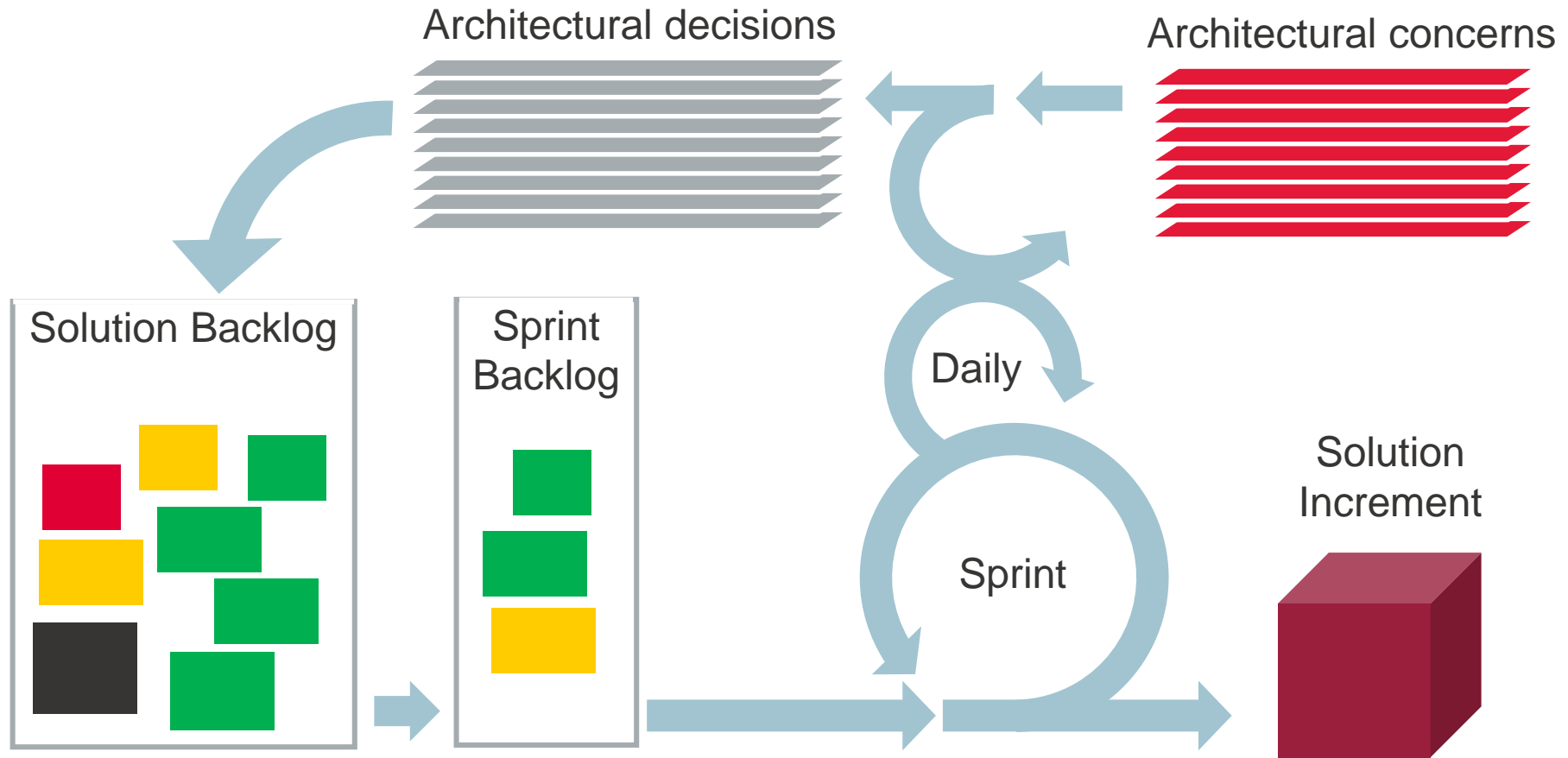


Training: Architecture as decision making



Architectural Decisions & the Agile Team

SCRUM



Architecting the Time Dimension

Just Enough Anticipation

Flow of architectural decisions ahead of development

Metaphor: Runway extended while in operation

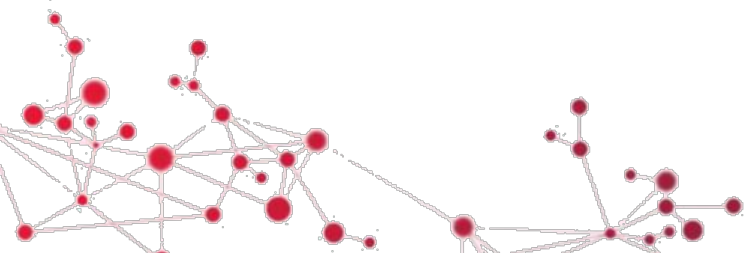
- Just long enough to accommodate anticipated airplanes



Key tools to determine right amount of anticipation:

- Dependency analysis
- Technical debt control
- Economic trade-off: Net Present Value, Real Options Analysis

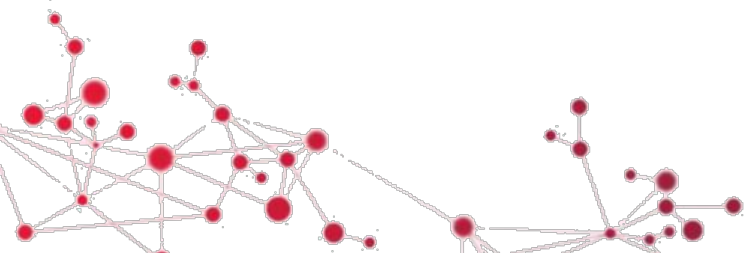
Source: Nanette Brown, Rod Nord, Ipek Ozkaya



Balance your backlog

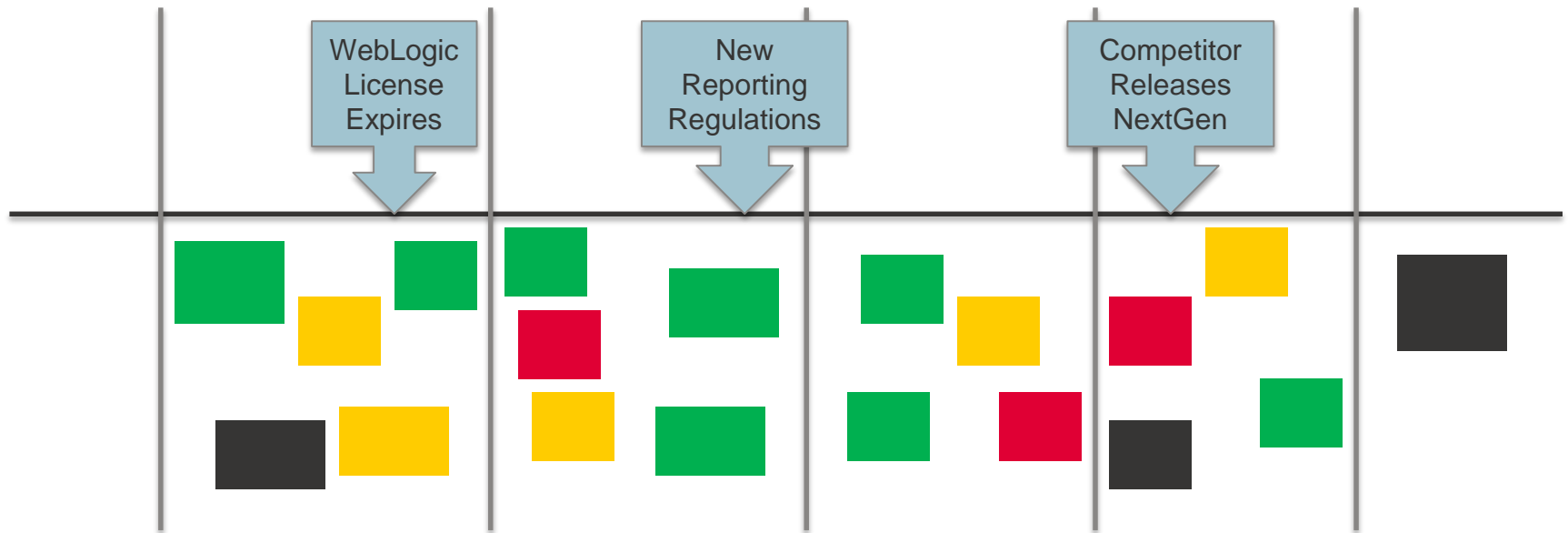
Architecture and other solution improvements

	Visible	Invisible
Positive Value	New features Added functionality	Architectural, Structural features
Negative Value	Defects	Technical Debt



Architecture Roadmapping

Sprint planning with Just Enough Anticipation



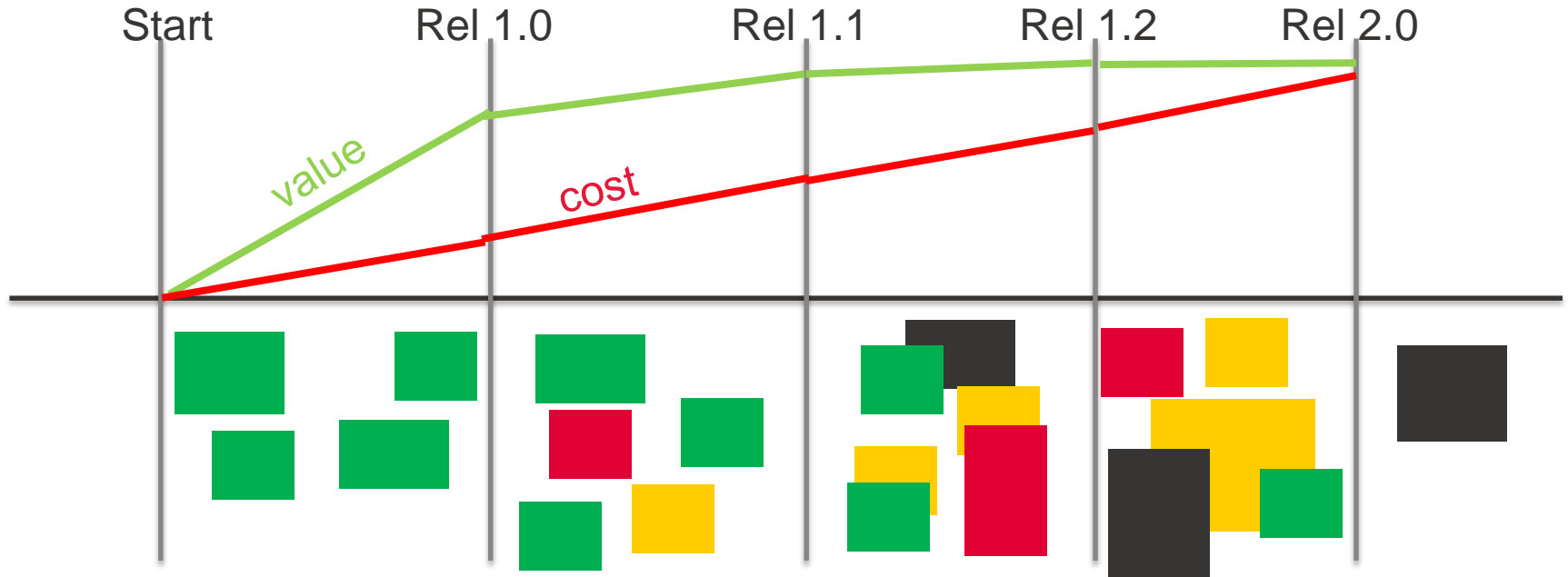
Assign solution improvement items to sprints based on

- Dependency analysis
- Real option value
- Technical debt control

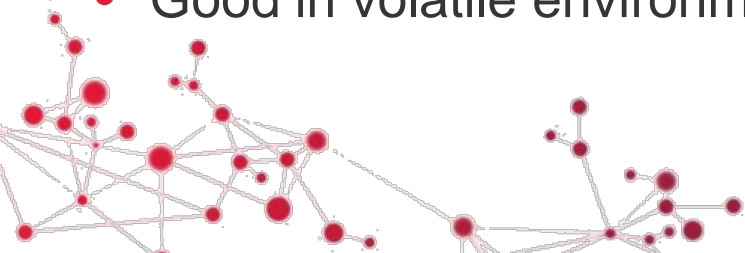


Architecture Roadmapping

Release strategy 1: value-first



- In line with Agile philosophy
- May increase TCO (more refactoring)
- Too “greedy” algorithm may run project into wall (complete rebuild)
- Good in volatile environments



Architecture Roadmapping

Release strategy 2: architecture-first



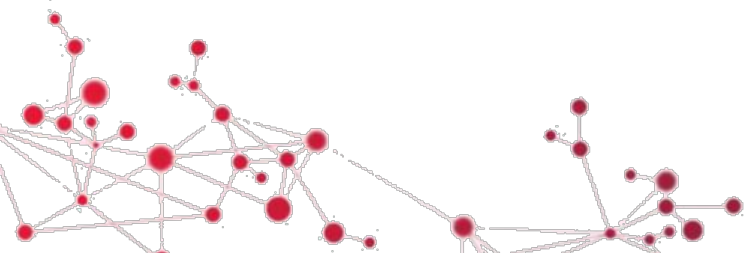
- In line with plan-driven philosophy
- Late delivery of value → risk of cancellation
- Risk of building wrong architecture (if context changes)
- Good for complex solutions



Lessons Learned

Recommendations for “Agilizing” the Architecture Department

- Teach architects:
 - **Decisions** (not documents) are their primary deliverable
 - **Economic reasoning** makes architecture’s value evident
- Make the **time** dimension a first class architectural entity
 - **Architecture Roadmap** at least as important as Reference Architecture
 - Architect with **Just Enough Anticipation** using dependency analysis, real option value and technical debt control



Questions?



Spare slides follow