

Prozessmodelle

für die Anwendung und Gestaltung von ERP-Systemen

Wer macht was (womit)?



E-mail: SCOTTADAMS@AOL.COM



Inhalt



1. Framework-Überblick

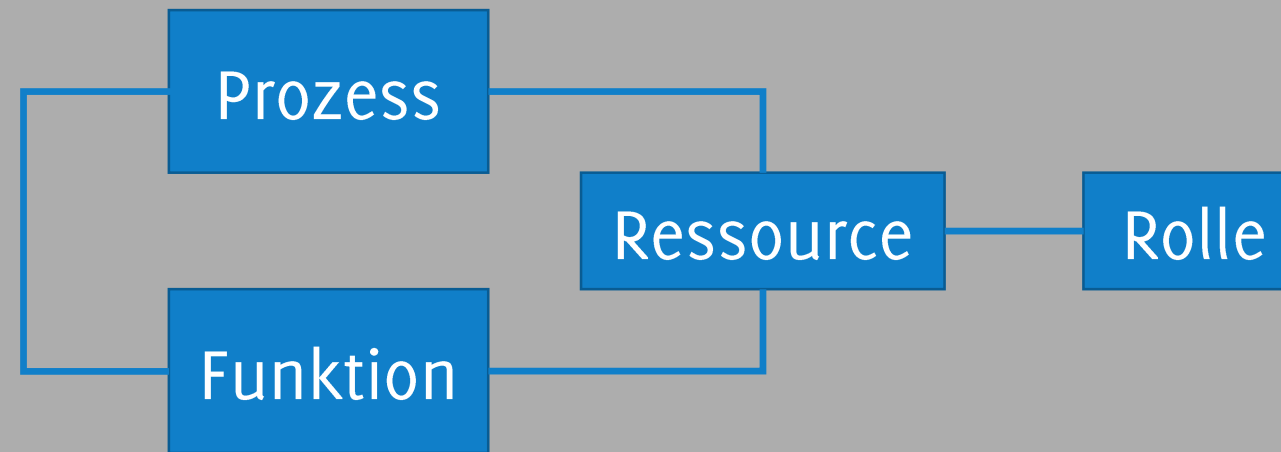


1. Framework-Überblick

Basiskomponenten



Frameworkdesign



Frameworkbewertung:
Umsetzungs- bzw.
Reifegrad



2. Framework-Ziele



2. Framework-Ziele

2.1 Standardisierung



Innenperspektive

Vermeidung von
Missverständnissen



Starre Symbolik

Außenperspektive

Unterstützung von
Benchmarking und
Outsourcing



Falsch messen,
das Falsche messen

2. Framework-Ziele



2.2 Performancemanagement

Innenperspektive

Verbesserung der operativen
Leistungsfähigkeit



Kostenrisiko,
Bürokratisierung,
Behinderung

Außenperspektive

Verbesserung der
Wettbewerbsfähigkeit



Wettbewerbsnachteile durch
schlechte Umsetzung bzw.
prozyklisches Verhalten

2. Framework-Ziele

2.3 Risikomanagement



Innenperspektive

Vermeidung von Kontroll-
und Steuerungslücken



mikropolitische Widerstände
gegen Transparenz und
Veränderung

Außenperspektive

Senkung von Haftungsrisiken
durch Orientierung an
Marktüblichkeit



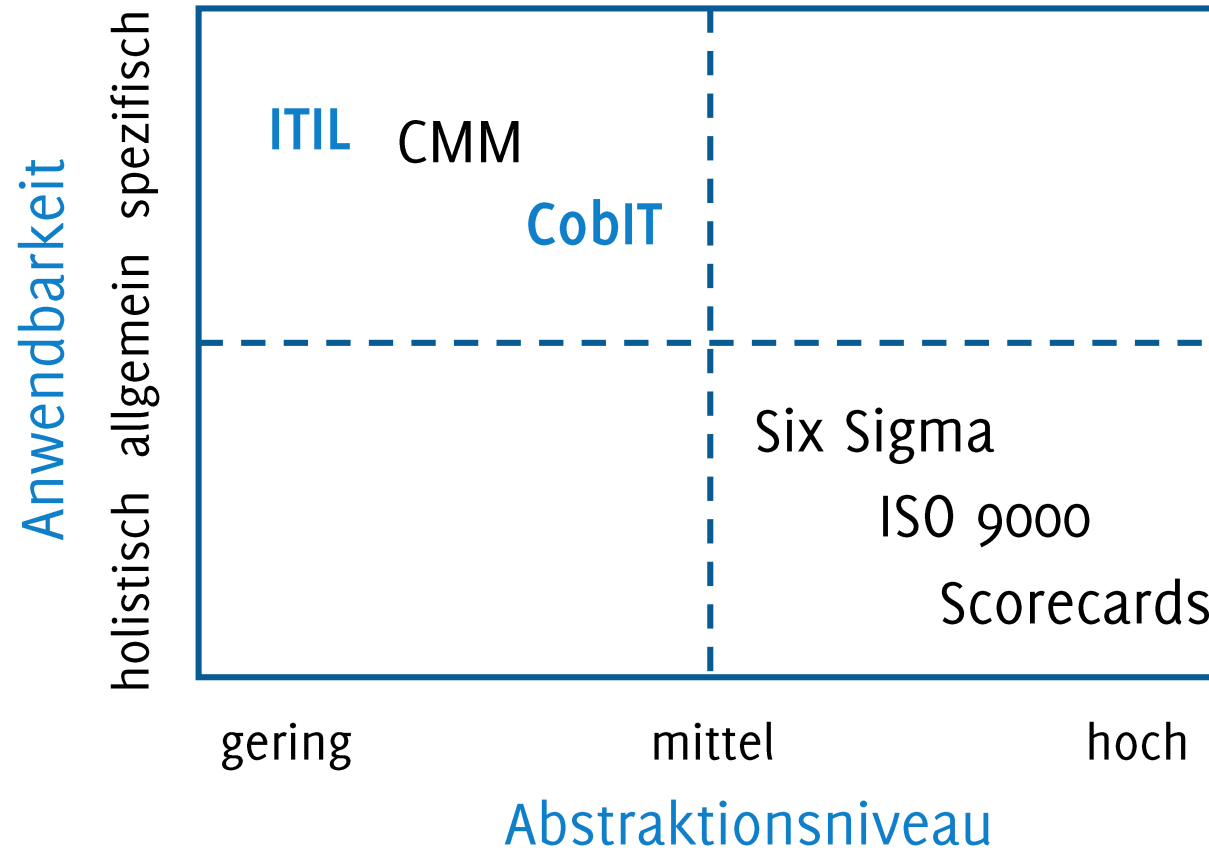
... bis sich neue Standards
durchgesetzt haben.



3. Beispiel-Frameworks



3.1 Positionierung nach Gartner, Inc.



ITIL: IT Infrastructure Library
CMM: Capability Maturity Model
CobIT: Control Objectives for Information and related Technologies
ISO: International Organization for Standardization

www.computerworld.com/s/article/907997/Model_Mania

3. Beispiel-Frameworks



3.2 Empfehlung Forrester Research

Governance-Unterstützung durch:

1. **Cobit** (IT Governance)
2. **ITIL** (IT Service Management)
3. **ISO 17799** → ISO 27000 (Informationssicherheit)
4. **BSC** (Messung und Kommunikation)

Quelle: C. Symons, Forrester Research: Helping Business Thrive On Technology Change – A Road Map To Comprehensive IT-Governance

3. Beispiel-Frameworks

3.3 ITIL

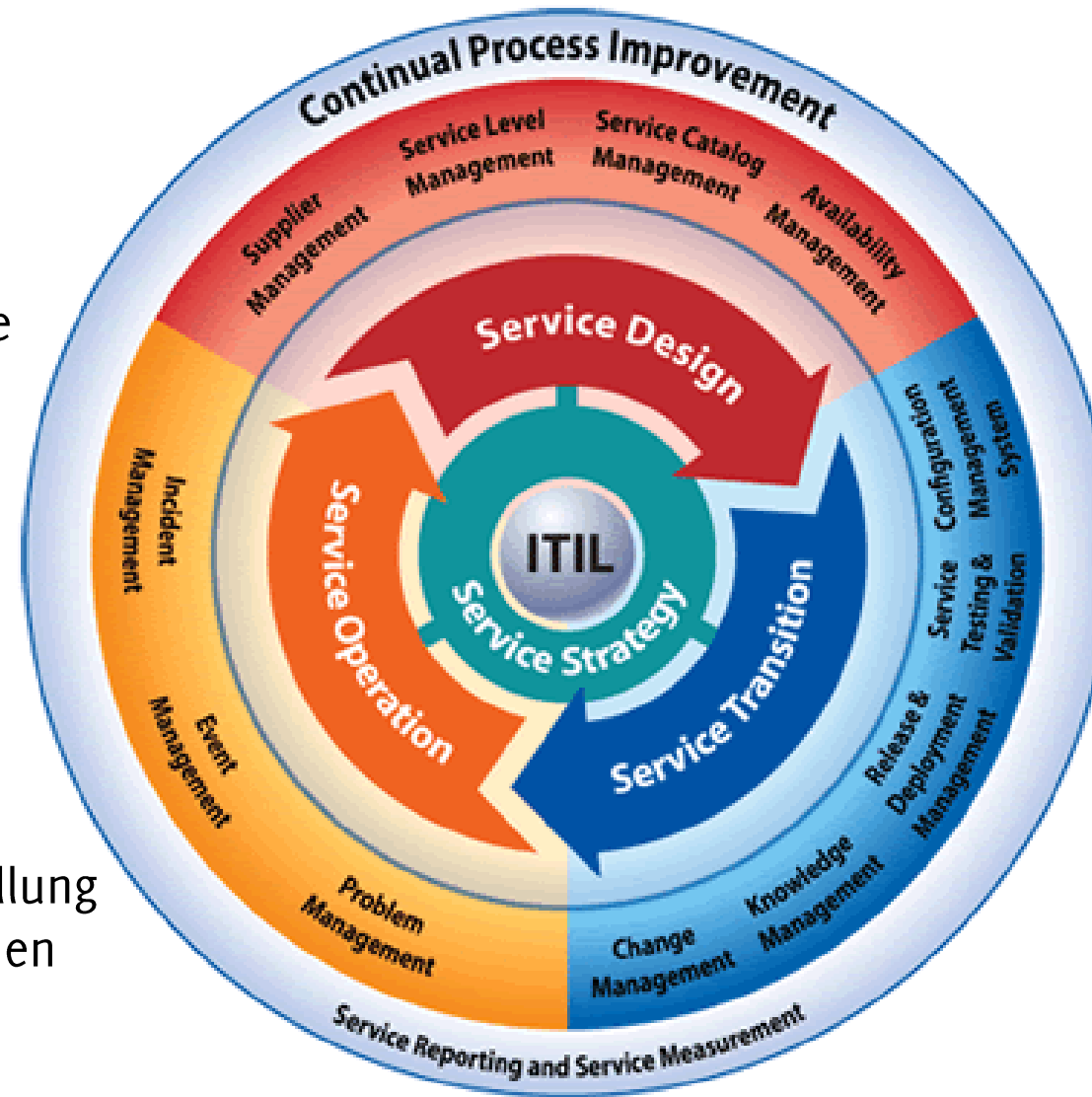


Service Strategy:

- Zielsetzung des Service Lifecycle
- Zusammenhang zur Geschäftsperspektive

Service Operation:

- störungsfreie Bereitstellung von Services im täglichen Betrieb
- Störungsbehandlung



Service Design:

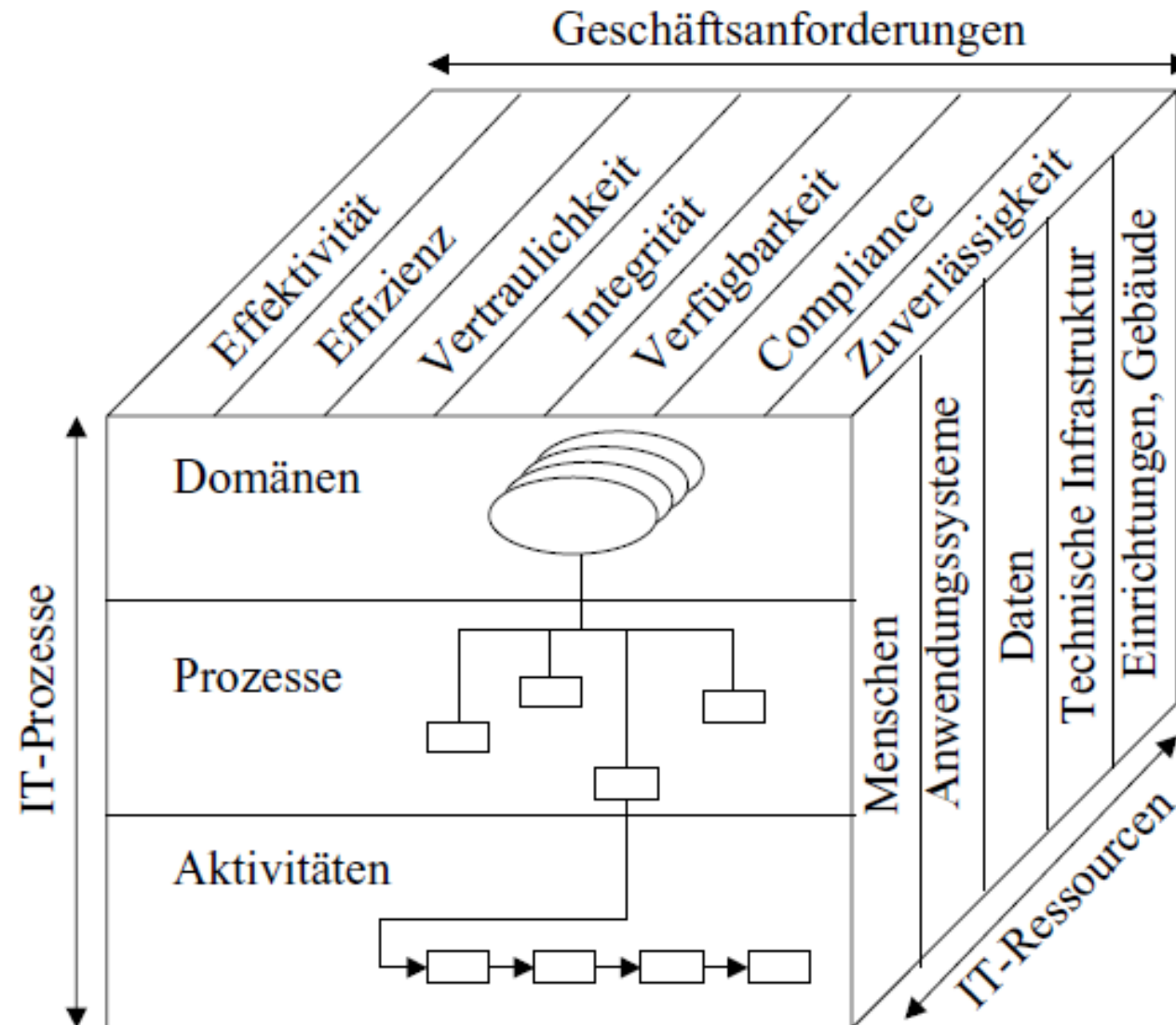
- Architektonische Rahmenbedingungen
- Inhaltliche und Sicherheitsaspekte

Service Transition:

- Praktische Umsetzung der geschäftlichen Anforderungen
- Schwerpunkt Produktionssetzung von Änderungen

3. Beispiel-Frameworks

3.4 Cobit



Process Optimization and Management System Development

A short case study about
managing agile projects

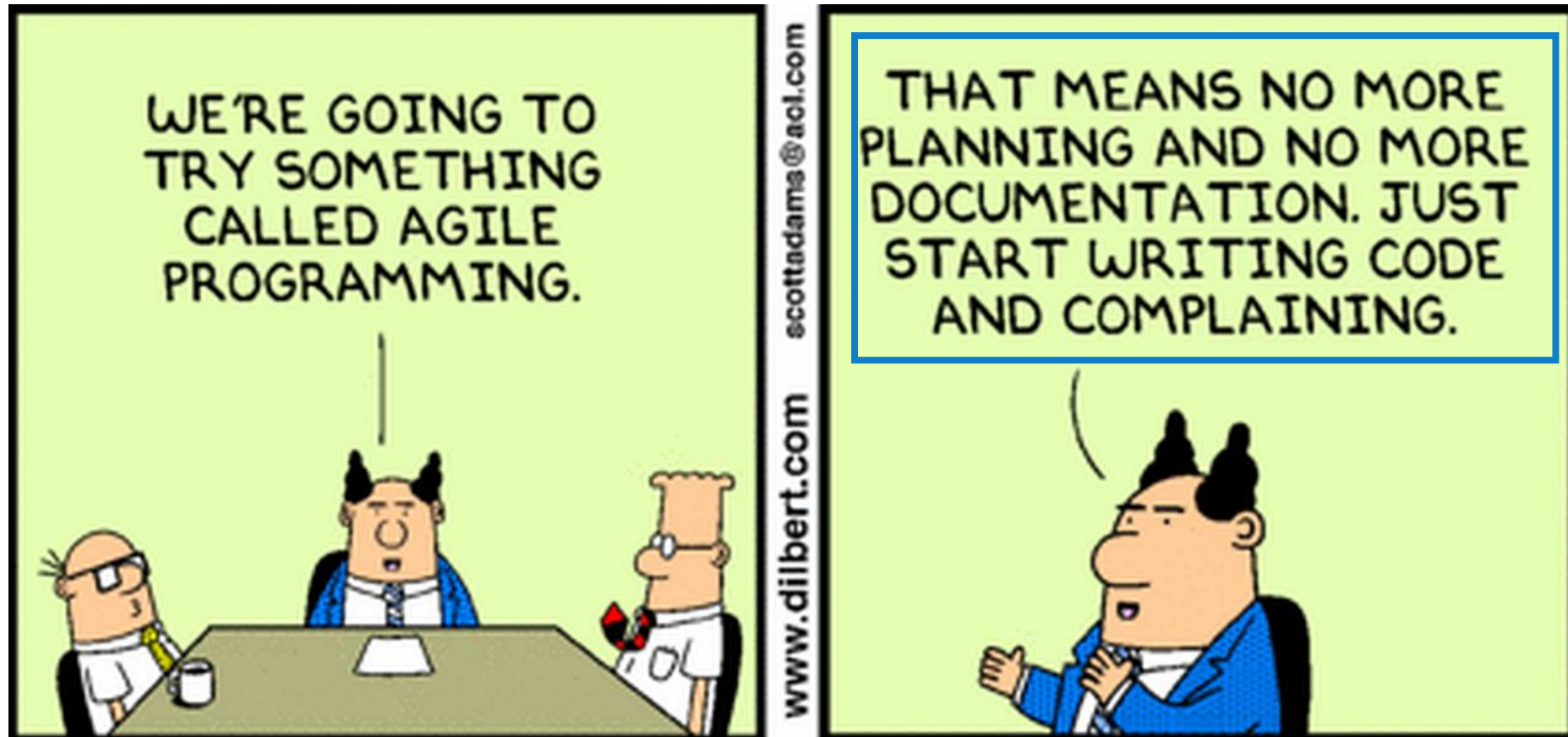
Content





1. Context

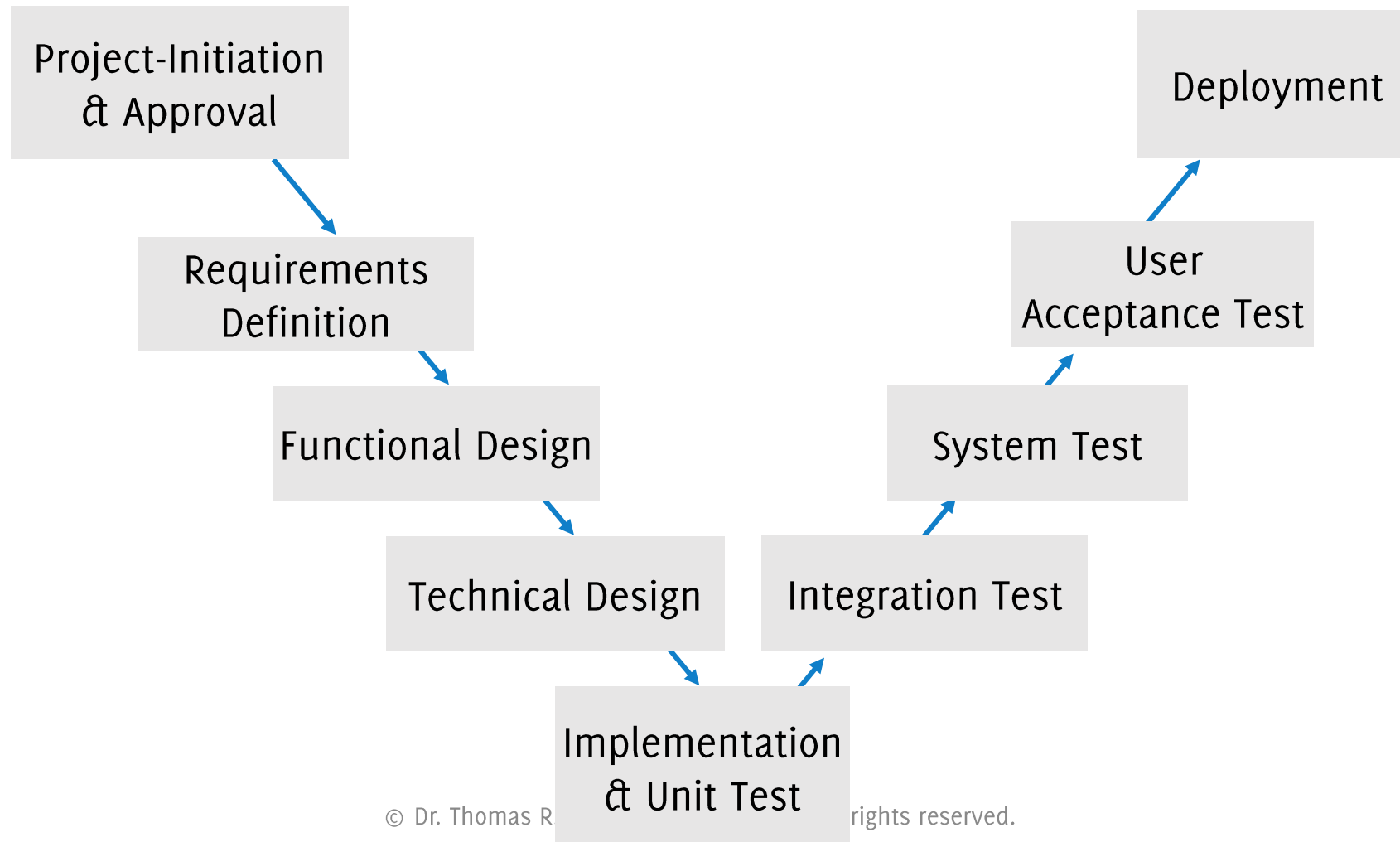
1.1 Process Model: Traditional vs. Agile





1. Context

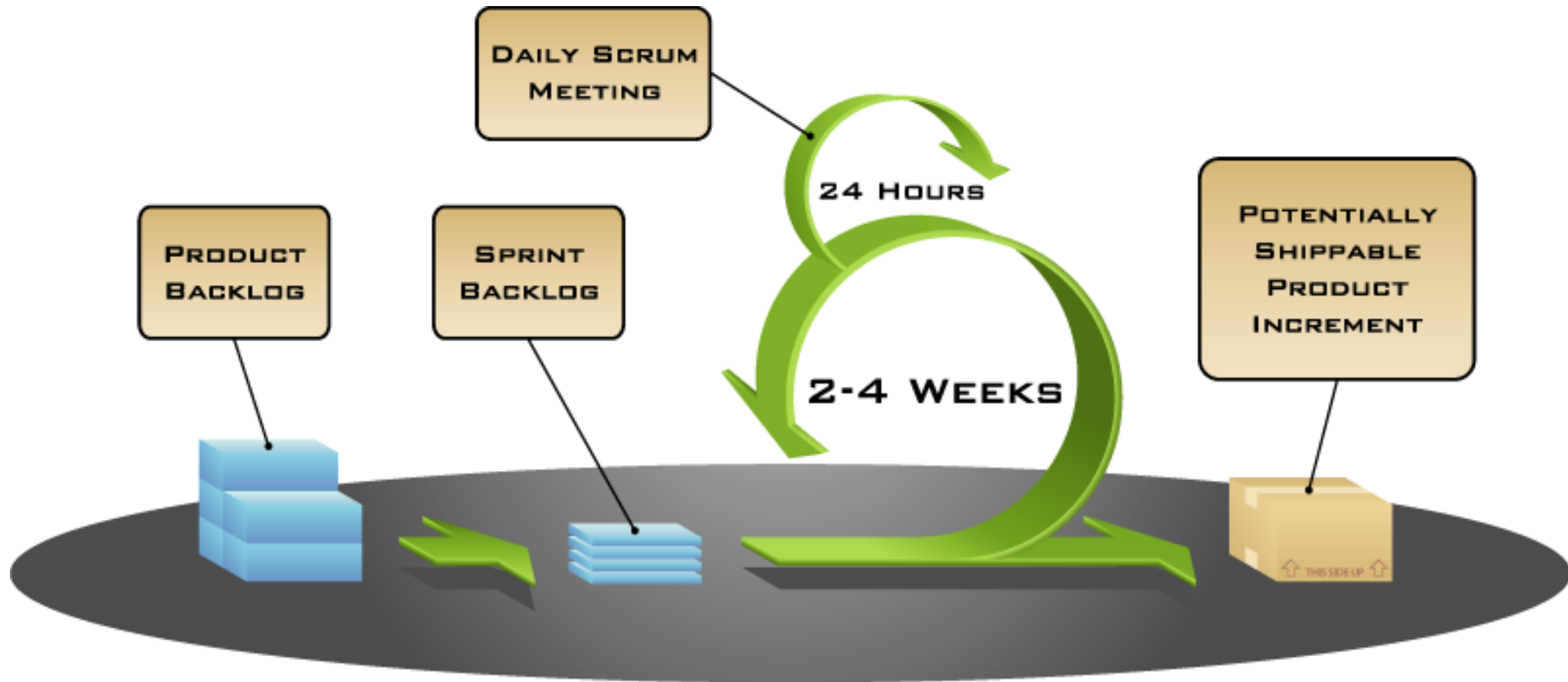
1.1 Process Model: Traditional vs. Agile





1. Context

1.1 Process Model: Traditional vs. Agile





1. Context

1.2 Just another management fad?

Main difference between traditional and agile models:
Degree of centralization in planning.

Otherwise there are agile elements in waterfall planning as well as waterfall elements in agile planning, depending on the scale.

In the end the probability of success depends on corresponding complexities:
If the project scope's complexity is underestimated, disproportionately high cost of adaptation might result, up to loss of control.



2. Project Risk Assessment

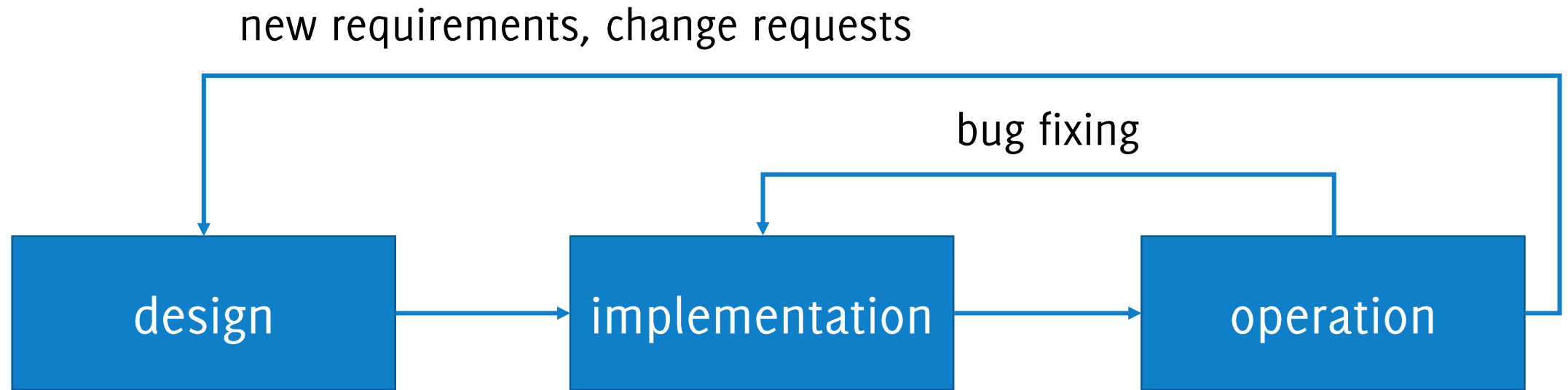
2.1 Casuistics alone won't help...





2. Project Risk Assessment

2.2 Sources of Problems in SW-Projects



- insufficient specification
- missing requirement
- misinterpretation of reqs
- incorrect implementation
- missing implementation
- unexpected results:
 - related to reqs
 - unrelated to reqs
- „genuine“ error

3. Initiative


3.1 JIRA





Developer(s)	Atlassian, Inc.
Initial release	2002 ^[1]
Stable release	6.0.3 / 25 June 2013; 14 days ago
Written in	Java
Operating system	Platform-independent
Type	Bug tracking system, project management software
License	Proprietary, free for use by official non-profit organizations, charities, and open-source projects, but not religious organizations ^{[2][3]}
Website	atlassian.com/software/jira 




 Products | [Company](#) | Try | Buy


[Company](#) | [Overview](#) | [About Us](#) | [Careers](#) | [Customers](#) | [Community](#) | [Press](#) | [Contact](#)



PEOPLE | [VALUES](#) | [FOUNDATION](#) | [SHIPIT DAYS](#) | [AWARDS](#)


Mike Cannon-Brookes and Scott Farquhar met while studying at the [University of New South Wales](#); they founded Atlassian in Sydney in 2002.^[10] The company made \$59 million in revenue in 2011,^[11] is on a \$100 million run-rate for the current fiscal year^[11] and has 26,000 customers globally.^[12] As of 2012 it has offices in [San Francisco](#), [Amsterdam](#) and [Tokyo](#).



The company was self-funded for many years, starting with a \$10,000 [credit card](#) taken out by the founders, but in July 2010 it raised its first institutional funding: \$60 million in [venture capital](#) from [Accel Partners](#).^[13] On June 24, 2011, Atlassian announced its first big investment in another company: [Cloud9](#) , a SaaS-based IDE platform.^[14]


The Leadership Team






Scott Farquhar
CO-FOUNDER AND CEO
 




Mike Cannon-Brookes
CO-FOUNDER AND CEO
 



Jay Simons
PRESIDENT
 

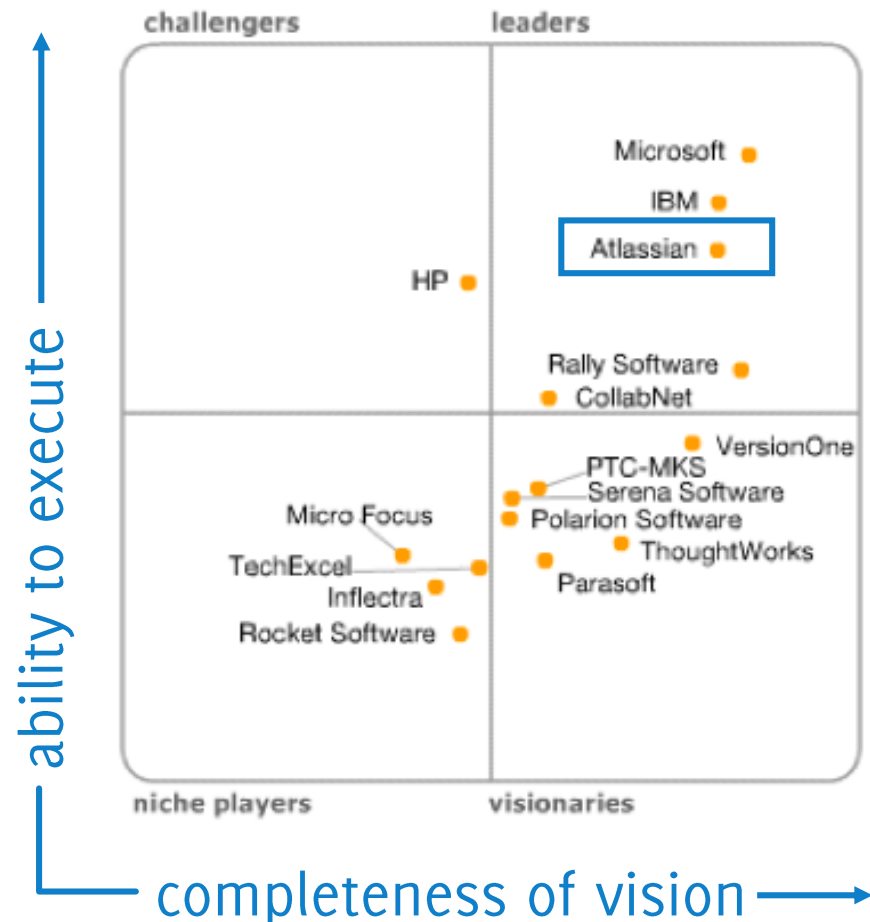


Alex Estevez
CHIEF FINANCIAL OFFICER




3. Initiative

3.2 Gartner about JIRA: „visionary leader“



Market leading system for project and process organization:

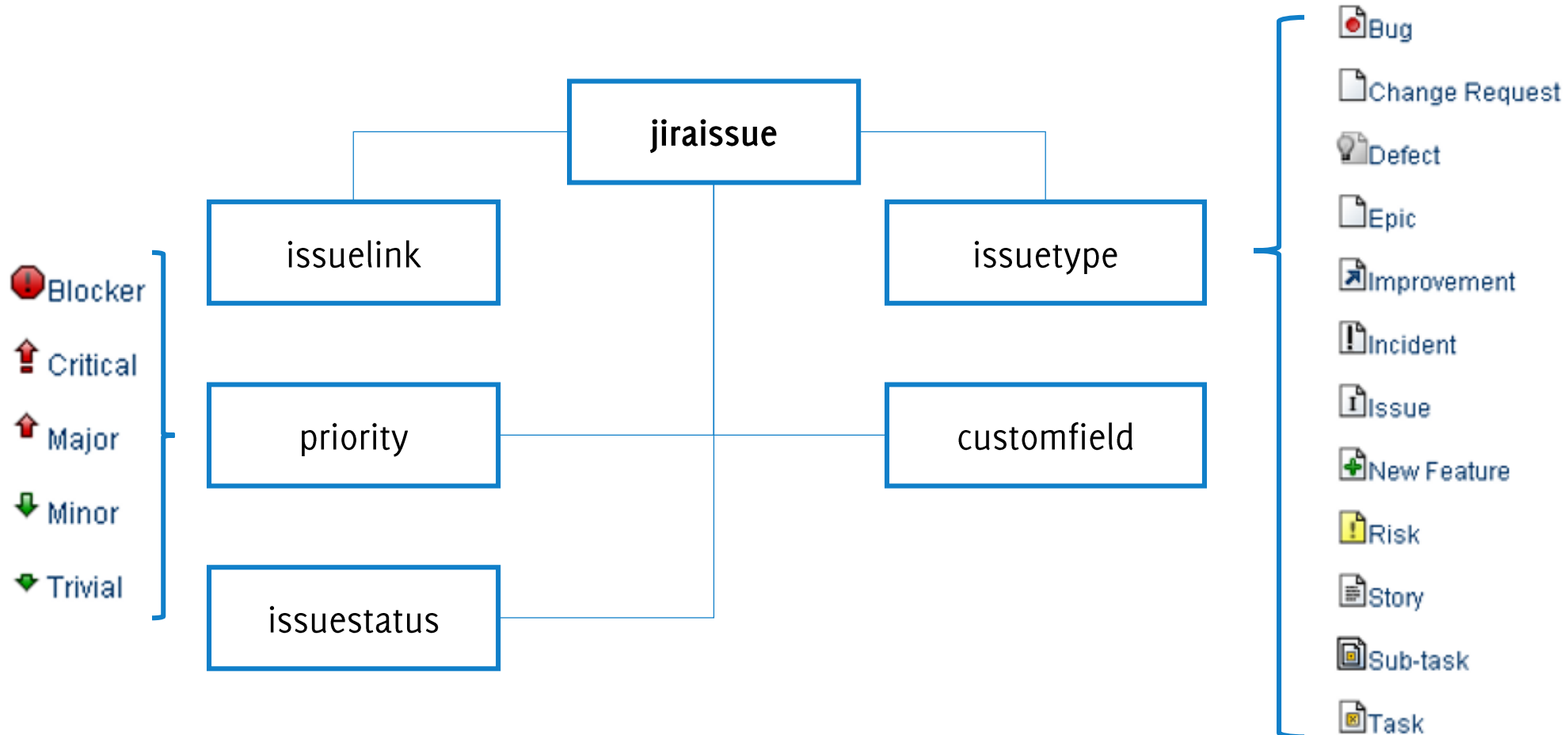
- easy and flexible customizing
- high usability
- seamless integration
- free configuration

⇒ maximum proximity to implementation



3. Initiative

3.3 JIRA: some basic elements





3. Initiative

3.4 JIRA Reporting: from JQL ...

The screenshot shows the JIRA web interface. At the top, the JIRA logo is on the left, and the user 'Rosie Jameson' is on the right next to a 'Quick Search' bar. Below the logo is a navigation bar with links for Dashboards, Projects, Issues (selected), Agile, and Administration. A '+ Create Issue' button is on the far right of this bar.

The main section is titled 'Issue Navigator'. It contains a search bar with a green checkmark icon and the text 'Query line: 1 character: 61'. The query entered is 'project in projectsLeadByUser(kolofsen) and summary ~ "link"'. To the right of the search bar is a 'Syntax Help' link with a question mark icon. Below the search bar is a 'Search' button and a checked 'Auto-complete' checkbox.

Below the search bar, it says 'Displaying Issues 1 to 2 of 2 matching Issues.' followed by a table of results.

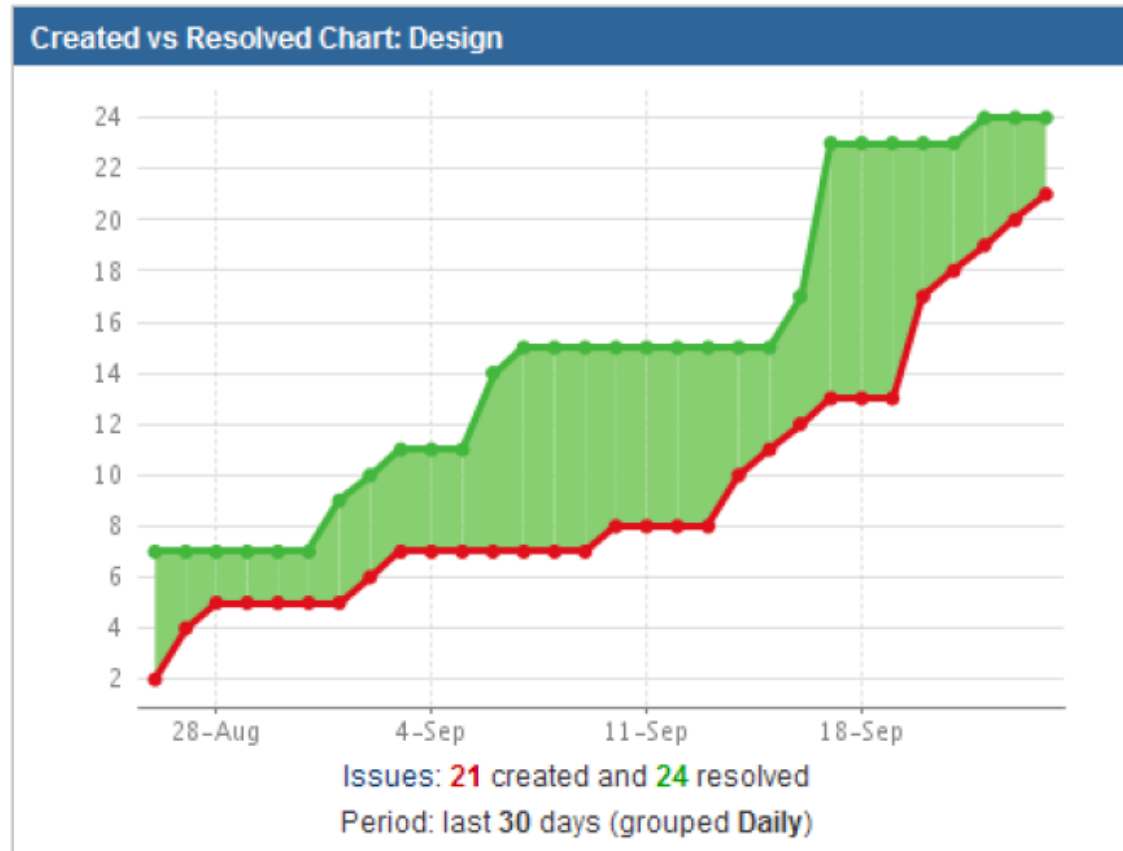
T	Key	Summary	Assignee	Reporter	P	Status	Resolution	Created	Updated
	BOX-112	Super duper link	Ken Olofsen	Frother		Open	Unresolved	25/Jun/10	25/Jun/10
	BOX-111	This link does not work	Ken Olofsen	Frother		Open	Unresolved	25/Jun/10	25/Jun/10

Below the table, it says 'Displaying issues 1 to 2 of 2 matching issues.'



3. Initiative

3.5 JIRA Reporting: ... to Dashboards

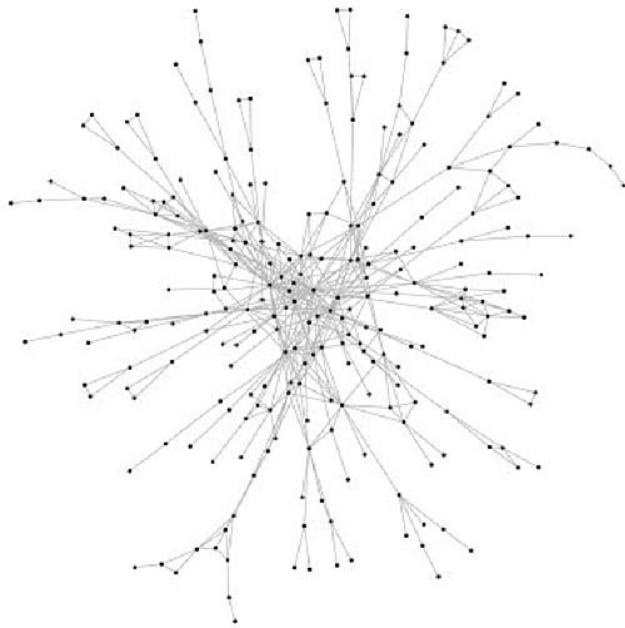


so why should it
be so difficult to
establish an
appropriate
project steering?



3. Initiative

3.7 Surprise: (Almost) Limitless Complexity



Challenge JIRA:
from best possible
project support follows
maximum complexity;
e.g. possible variations
of trees on n labeled nodes:

$$y = n^{n-2}$$

(Cayley's formula).



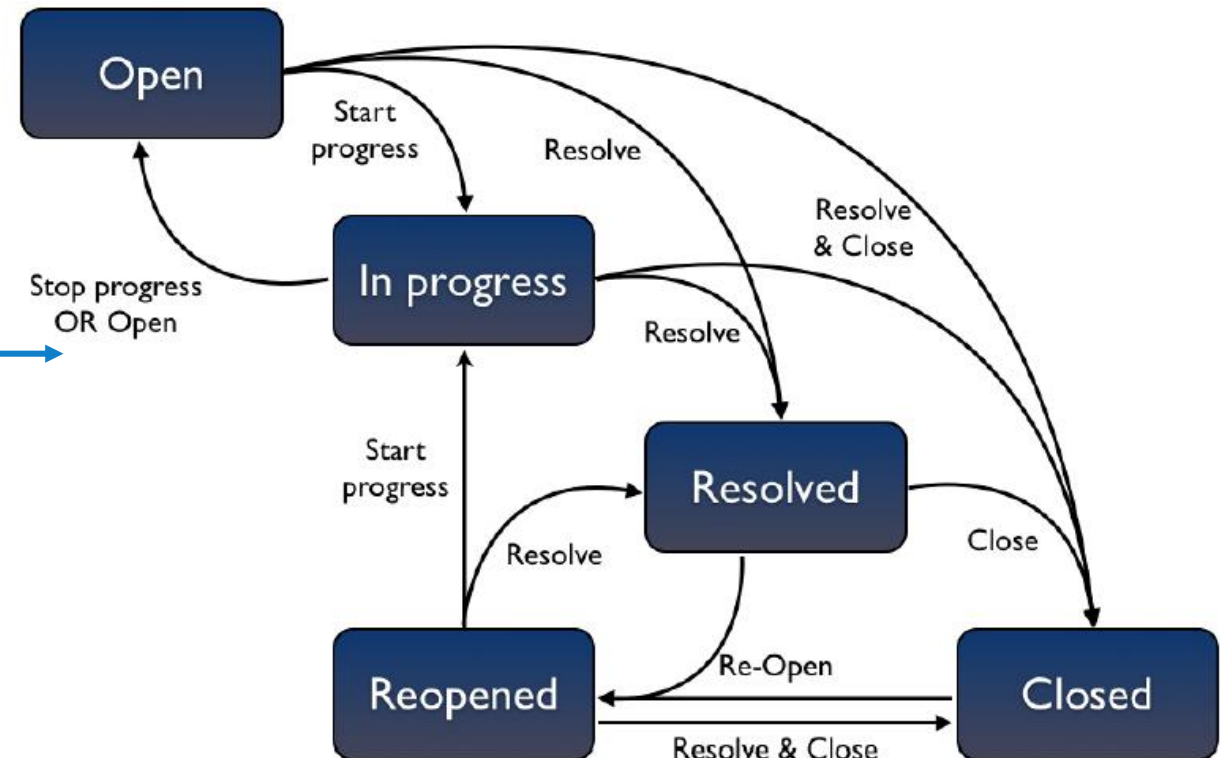


3. Initiative

3.8 JIRA Customizing: Challenge intensified

Customizing:

- Issuetypes
- Issuetype-Customfields
- Workflows
- Issuelinks
- ...





3. Initiative

3.9 JIRA's Strengths as Weaknesses?

JIRA's main weakness is conditional on its design principles and immediately results from its strengths: JIRA is able to model the organizational reality best possible and on almost any scale.

On the downside, this maximum proximity to implementation inevitably means maximum steering complexity (/„distance“), which is also mentioned by Gartner as „**cautions**“:

- Lack of a single integrated dashboard experience
- No single control point for workflow
- Limited support for complex processes



4. Solution

4.1 Integrating JIRA

The trade-off problem between free customizing and standardized reporting can't be solved from within the system itself.

My process model:

1. Reverse engineering JIRA
2. Synchronising JIRA with our own database
3. Adding extended features (views, functions, procedures)
4. Applying Business Intelligence/Analysis Services

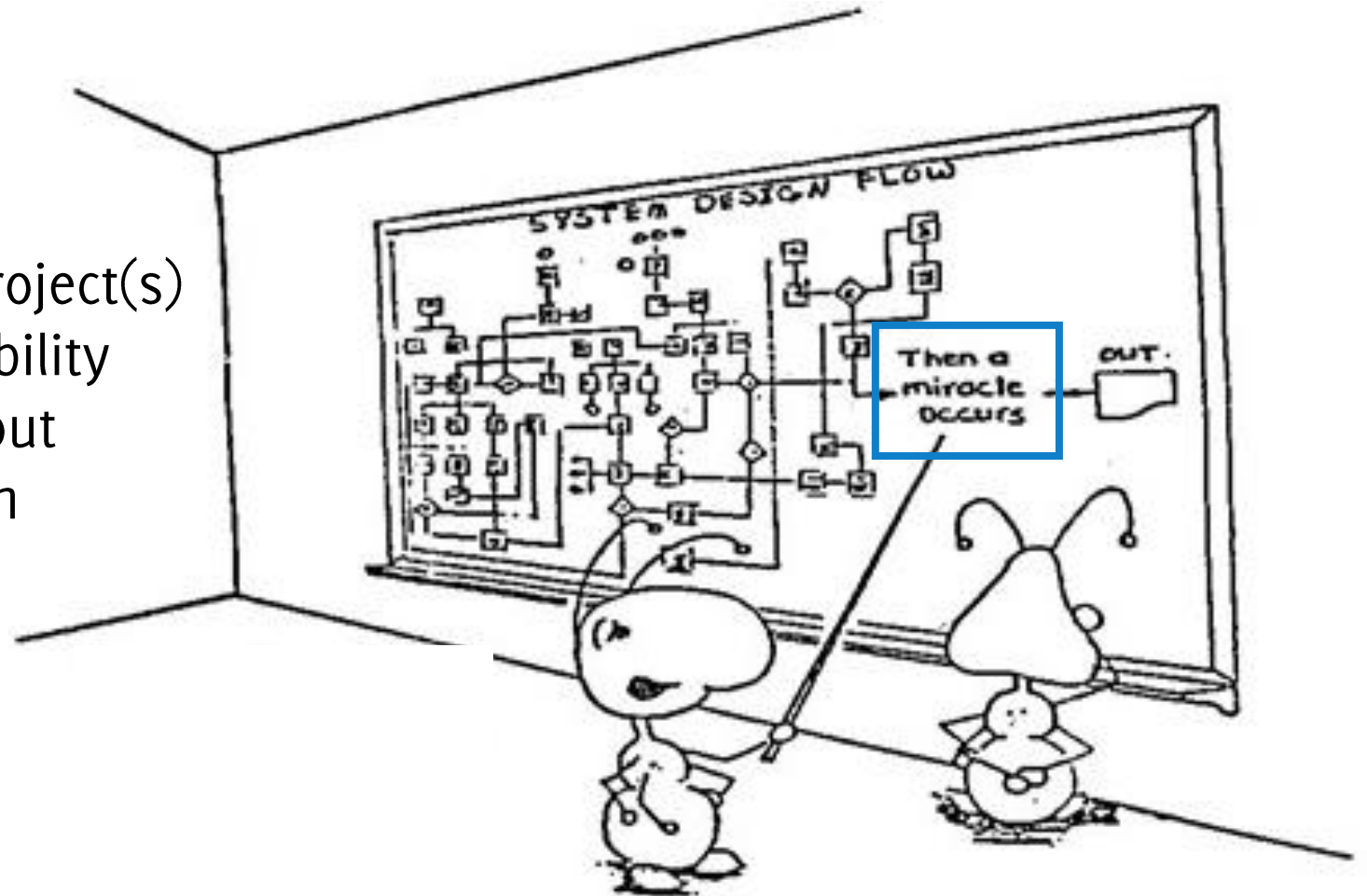


4. Solution

4.2 Constraints

Non-invasive design:

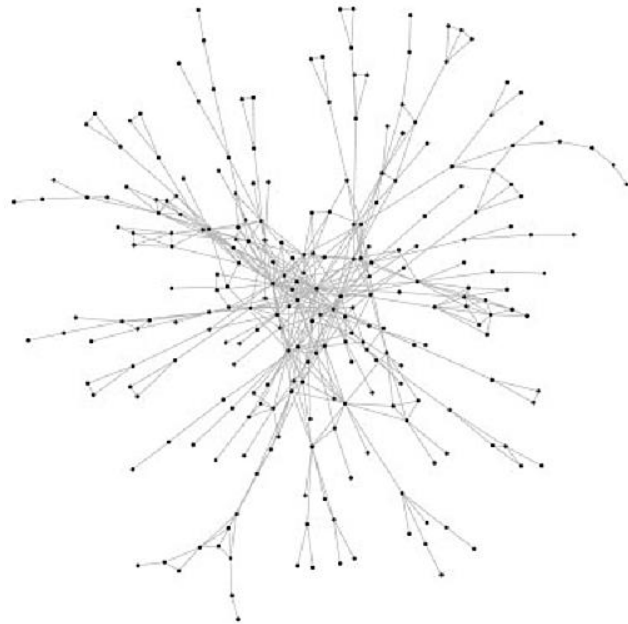
- no interference with the project(s)
- no limitation of JIRA's flexibility
- complexity reduction without relevant loss of information





4. Solution

4.3 Then a miracle occurs



The simple solution lies in intelligently consolidating the dynamic issue-networks:

- ⇒ almost real-time, robust reports about the health status of even the most complex projects
- ⇒ the best statistics is a complete inventory: potentially lossless consolidation of all activities
- ⇒ scale-invariant, easy drill-down to the smallest, elementary details
- ⇒ ...