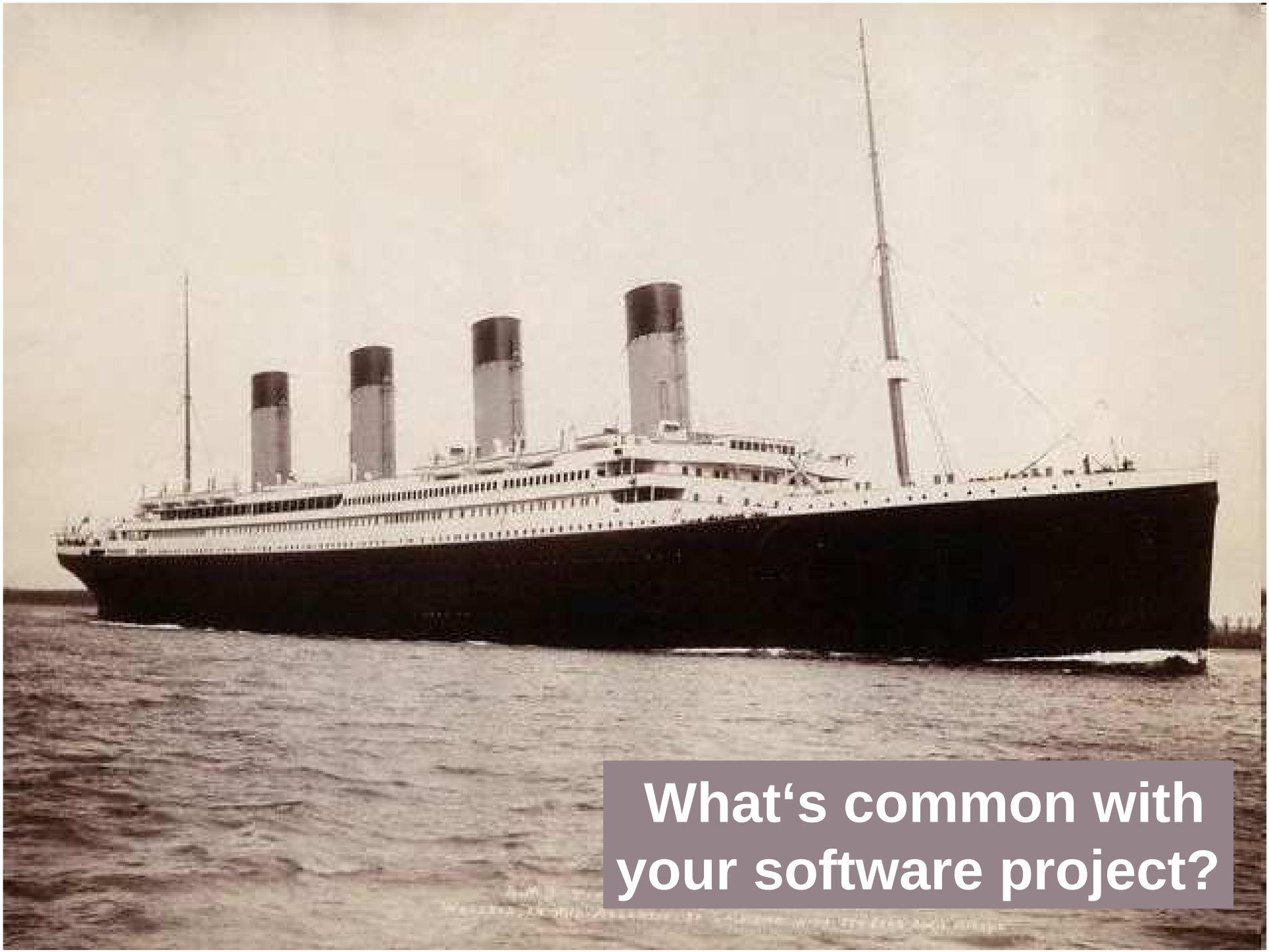
Building for Performance and Scalability

How to integrate performance management into your development process

Alois Reitbauer, Technology Strategist

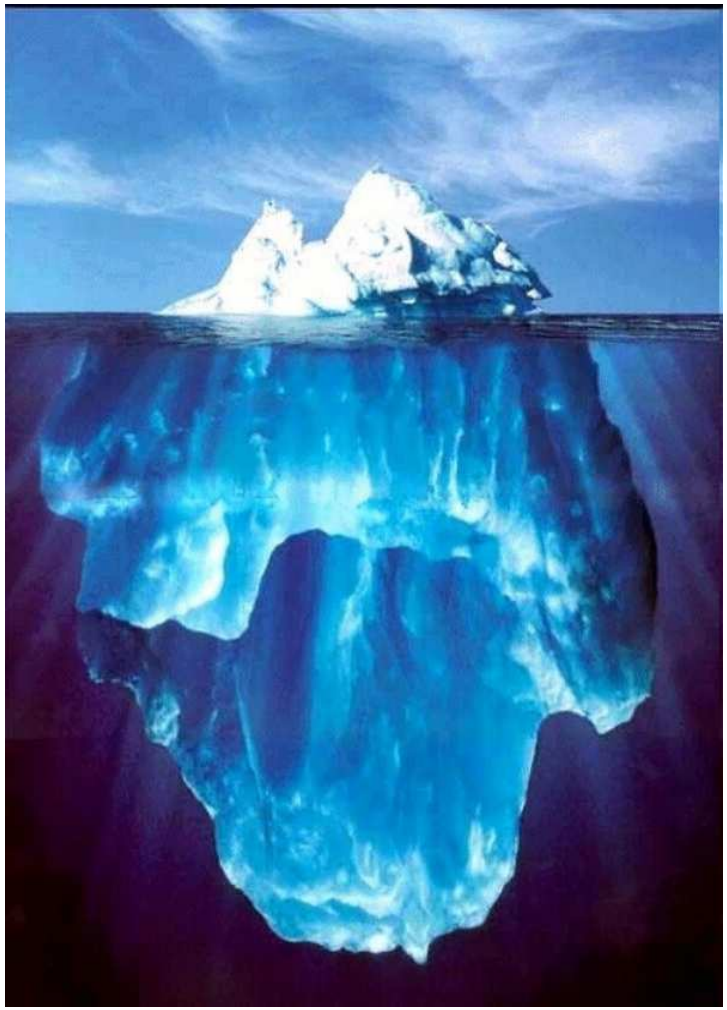
http://blog.dynatrace.com

dynaTrace software

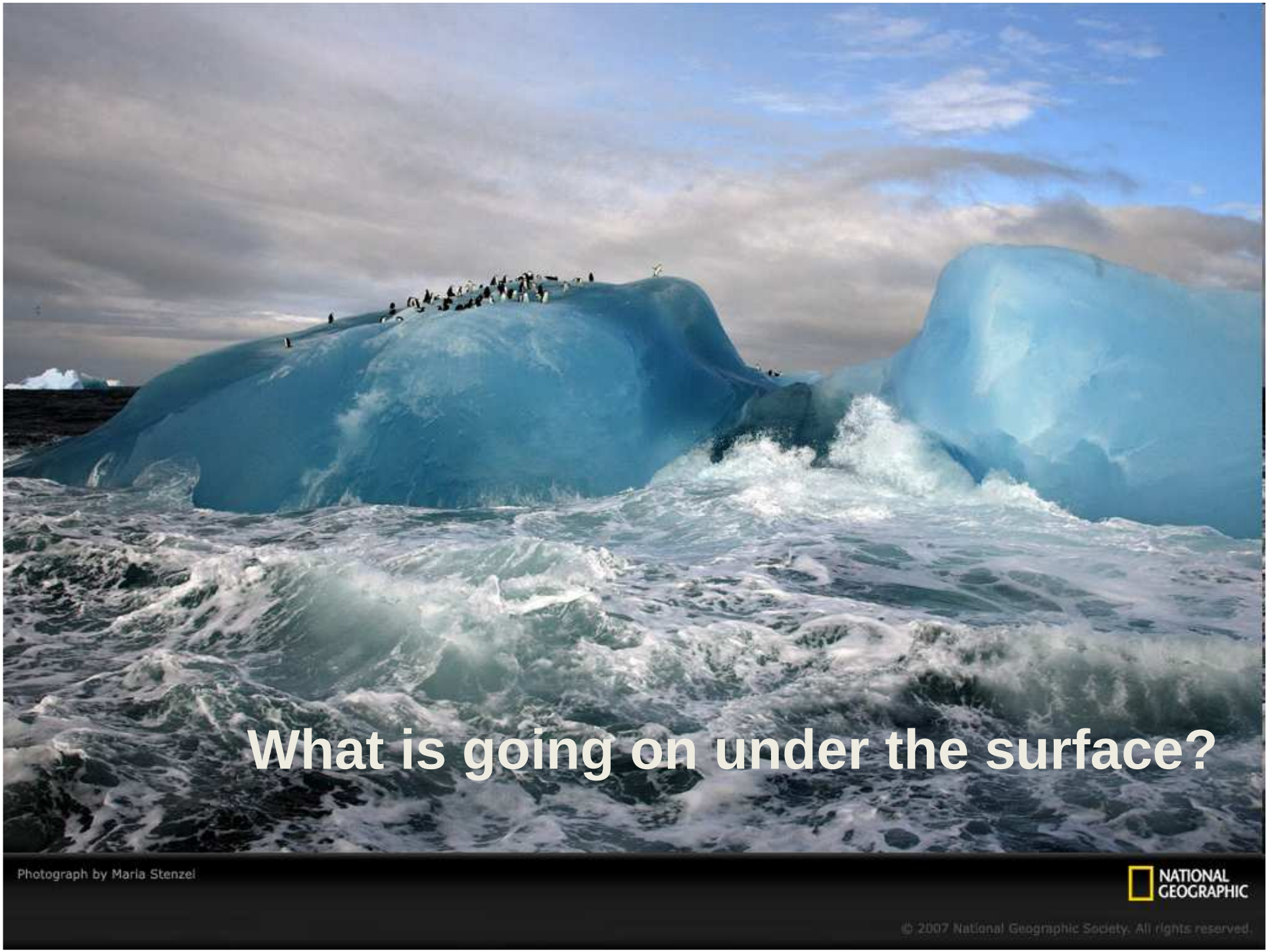

What‘s common with
your software project?

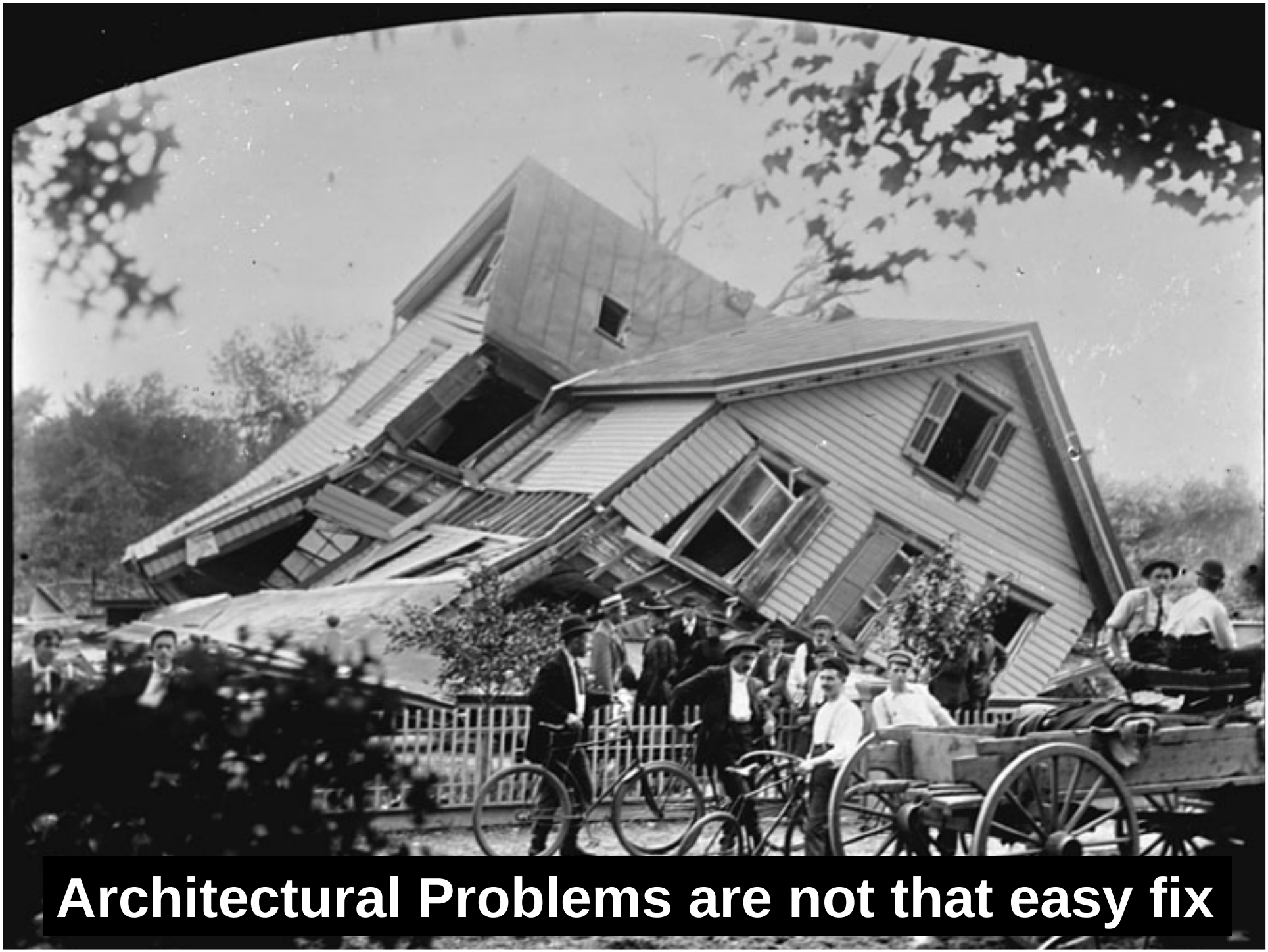
The Iceberg

- Performance problems often don‘t show up during development
- Performance testing is done late in the lifecycle (if at all)
- Problems manifest late in load testing or production
- Development must solve problems in a „hostile“ environment



dynaTrace software

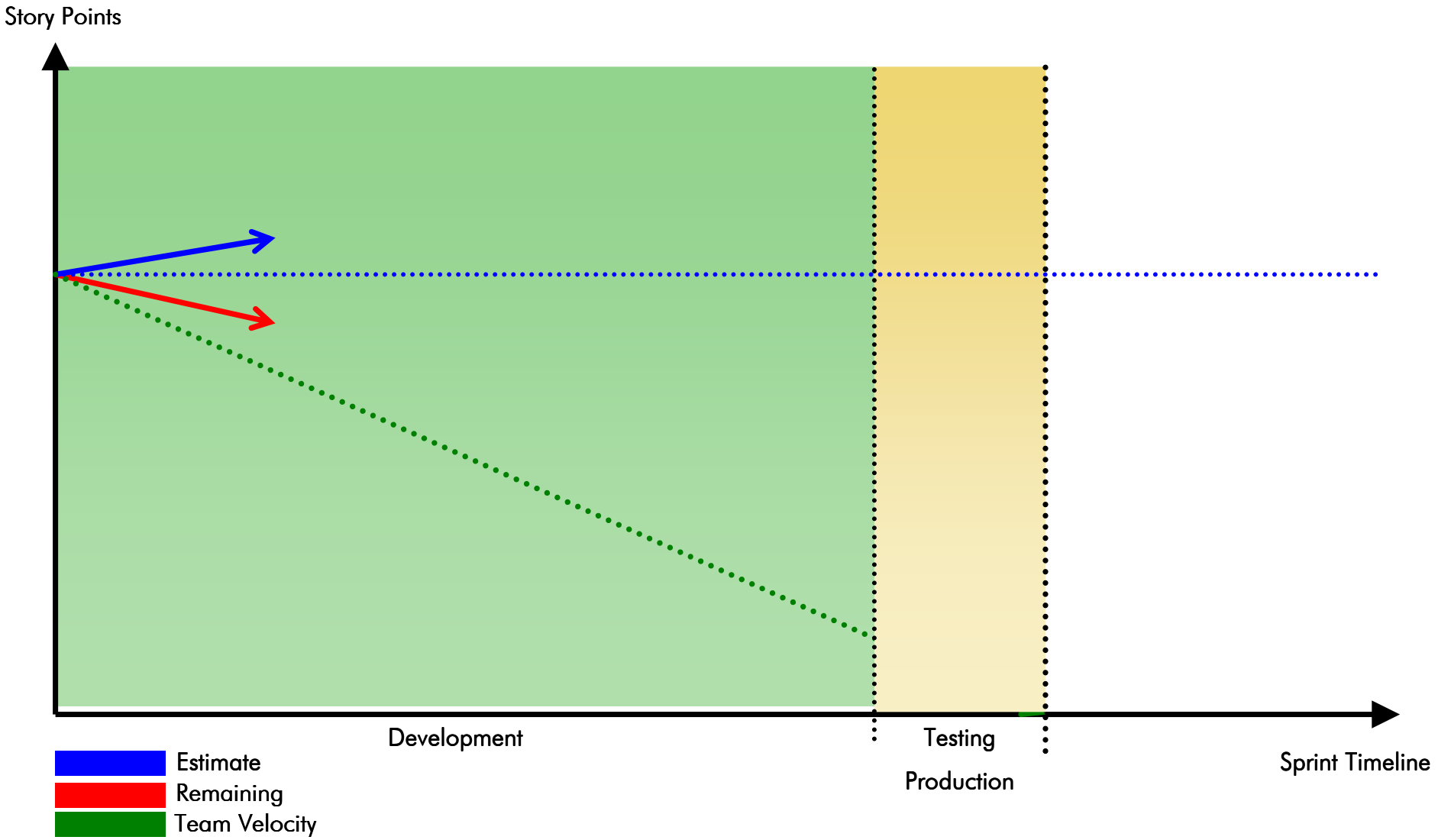

What is going on under the surface?
Photograph by Maria Stenzel
NATIONAL GEOGRAPHIC
© 2007 National Geographic Society. All rights reserved.


Architectural Problems are not that easy fix

However, you are agile and flexible

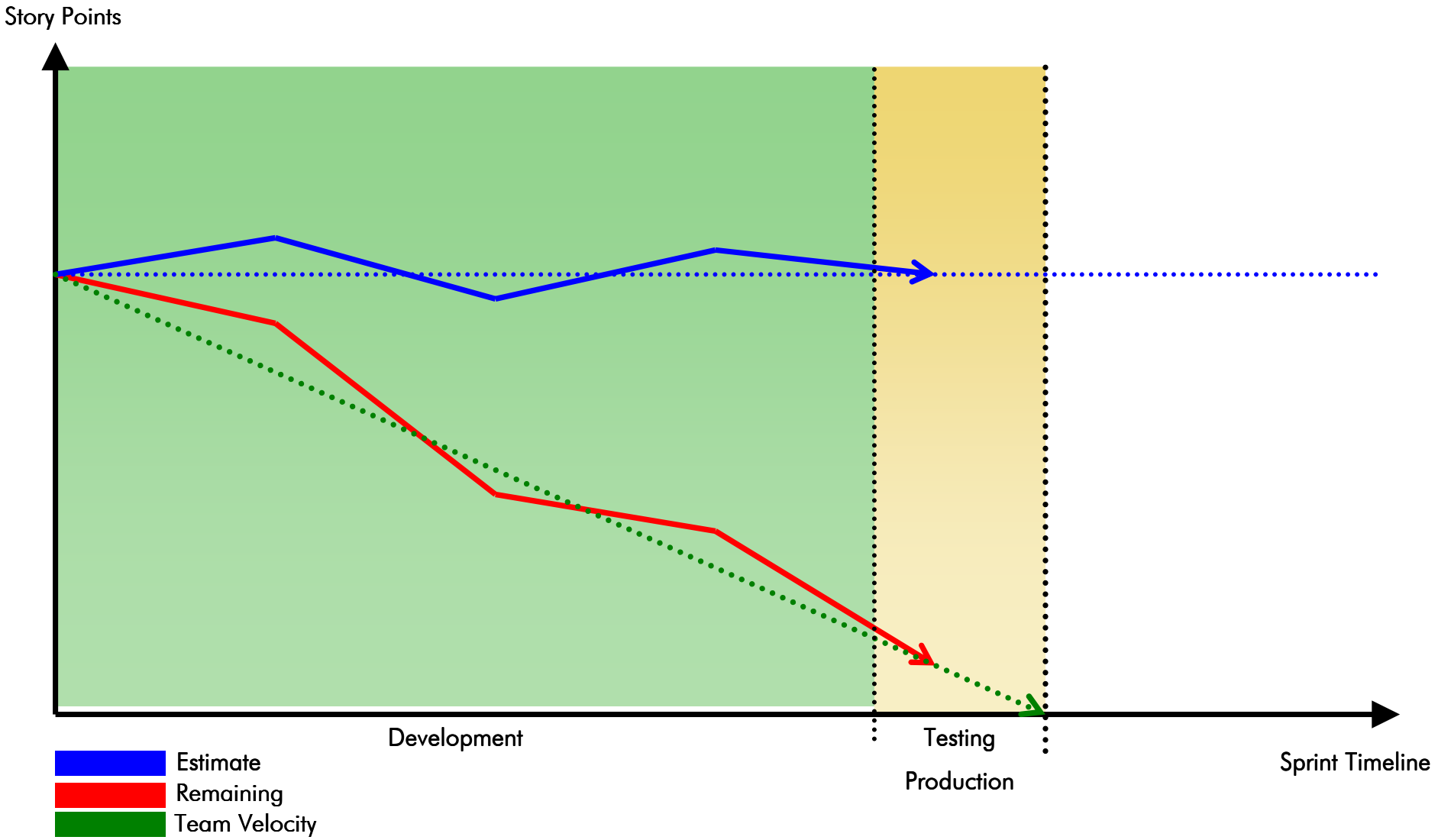
dynaTrace software

Monitoring in Agile Development



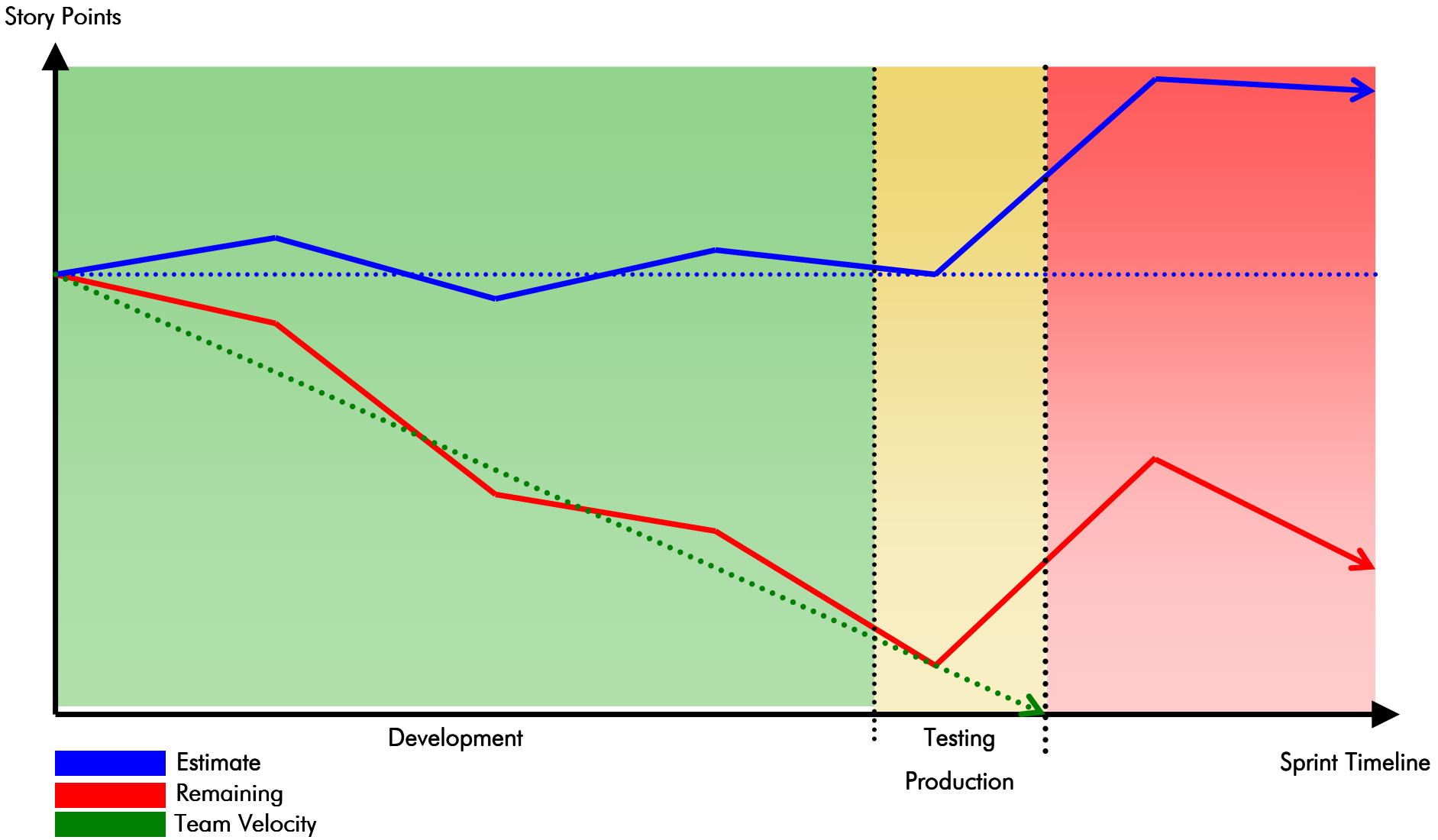
dynaTrace software

You are in control



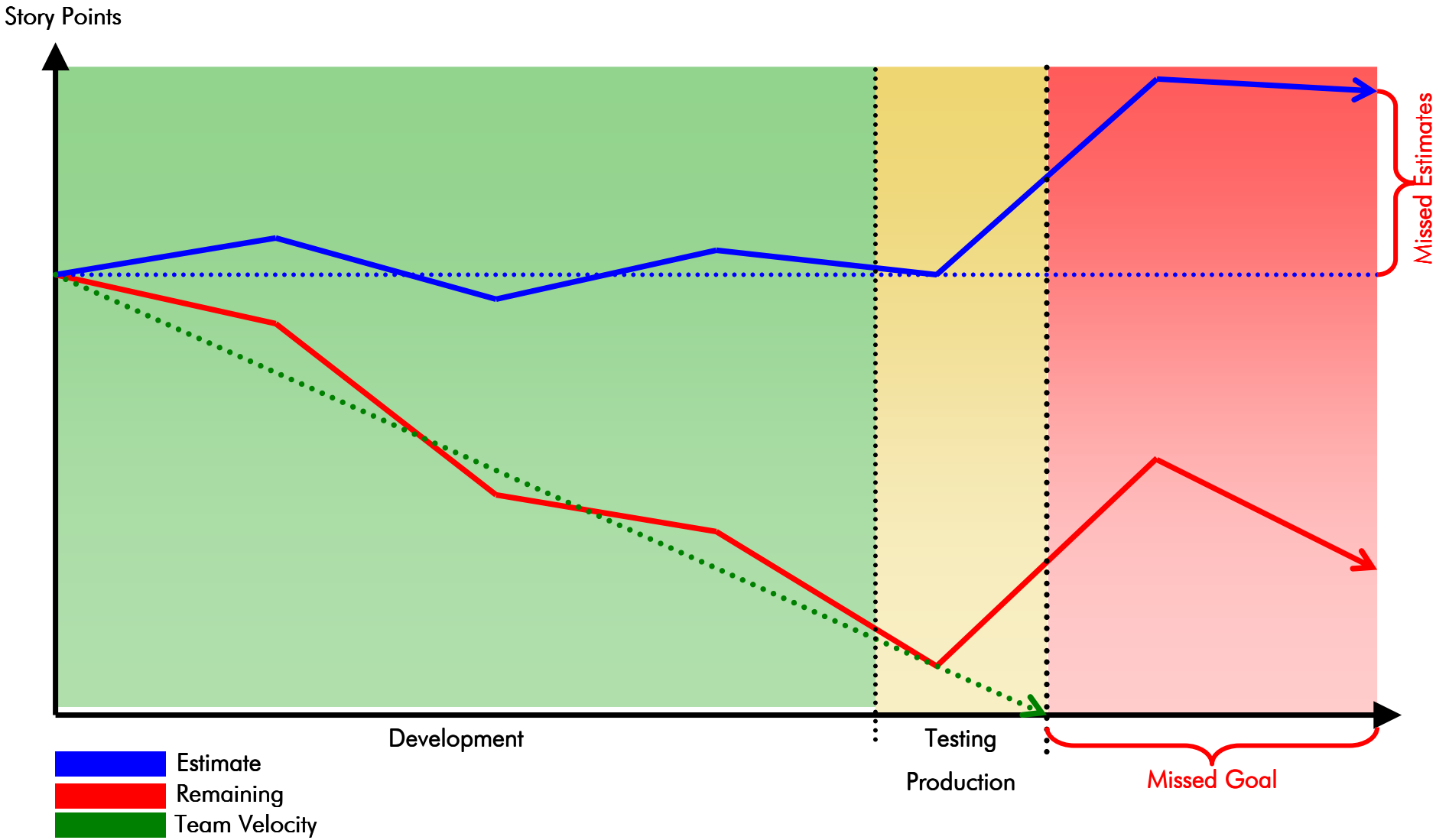
dynaTrace software

What happened?



dynaTrace software

Missed Goals and Estimates



dynaTrace software

So ....

... make performance management part of your development processes

dynaTrace software

BUT .....

Taken from K. Scott Allen‘s Blog : (www.odetocode.com)

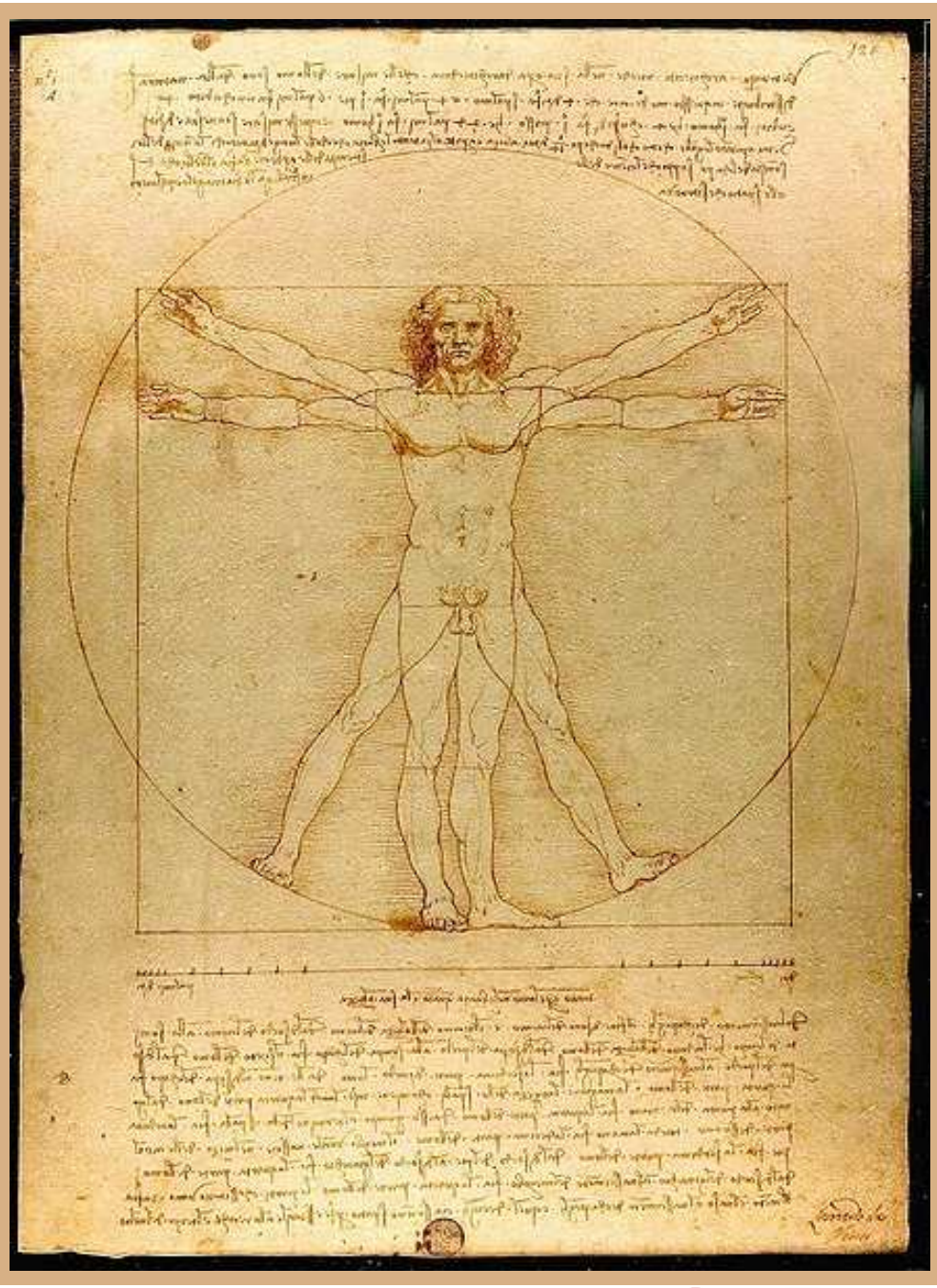
dynaTrace software

What if this was the problem?

Method	Class	Argument
⊟ exampleLazyLoading()	com.dynatrace.samples.database.Hibern...	
openSession()	org.hibernate.impl.SessionFactoryImpl	
createQuery(java.lang.String)	org.hibernate.impl.SessionImpl	from Person p
⊟ list()	org.hibernate.impl.QueryImpl	
⊟ list(java.lang.String, org.hibernate.engine.QueryParameters)	org.hibernate.impl.SessionImpl	
isTransactionInProgress()	org.hibernate.impl.SessionImpl	
⊟ prepareStatement(java.lang.String)	org.apache.derby.client.am.Connection	select person0_.ID as ID0_, person0_.firstName as firstName0_, person0_.lastName
new PreparedStatement40(org.apache.derby.client.am.A	org.apache.derby.client.am.PreparedSta...	org.apache.derby.client.am.Agent;org.apache.derby.client.am.Connection;java.lan
executeQuery()	org.apache.derby.client.am.PreparedSta...	select person0_.ID as ID0_, person0_.firstName as firstName0_, person0_.lastName
⊟ initializeCollection(org.hibernate.collection.PersistentCollection	org.hibernate.impl.SessionImpl	
⊞ prepareStatement(java.lang.String)	org.apache.derby.client.am.Connection	select addresses0_.person_ID as person6_1_, addresses0_.ID as ID1_, addresses0_
executeQuery()	org.apache.derby.client.am.PreparedSta...	select addresses0_.person_ID as person6_1_, addresses0_.ID as ID1_, addresses0_
⊟ initializeCollection(org.hibernate.collection.PersistentCollection	org.hibernate.impl.SessionImpl	
⊞ prepareStatement(java.lang.String)	org.apache.derby.client.am.Connection	select addresses0_.person_ID as person6_1_, addresses0_.ID as ID1_, addresses0_
executeQuery()	org.apache.derby.client.am.PreparedSta...	select addresses0_.person_ID as person6_1_, addresses0_.ID as ID1_, addresses0_
⊟ initializeCollection(org.hibernate.collection.PersistentCollection	org.hibernate.impl.SessionImpl	
⊞ prepareStatement(java.lang.String)	org.apache.derby.client.am.Connection	select addresses0_.person_ID as person6_1_, addresses0_.ID as ID1_, addresses0_
executeQuery()	org.apache.derby.client.am.PreparedSta...	select addresses0_.person_ID as person6_1_, addresses0_.ID as ID1_, addresses0_
⊟ initializeCollection(org.hibernate.collection.PersistentCollection	org.hibernate.impl.SessionImpl	
⊞ prepareStatement(java.lang.String)	org.apache.derby.client.am.Connection	select addresses0_.person_ID as person6_1_, addresses0_.ID as ID1_, addresses0_
executeQuery()	org.apache.derby.client.am.PreparedSta...	select addresses0_.person_ID as person6_1_, addresses0_.ID as ID1_, addresses0_
⊟ initializeCollection(org.hibernate.collection.PersistentCollection	org.hibernate.impl.SessionImpl	
⊞ prepareStatement(java.lang.String)	org.apache.derby.client.am.Connection	select addresses0_.person_ID as person6_1_, addresses0_.ID as ID1_, addresses0_
executeQuery()	org.apache.derby.client.am.PreparedSta...	select addresses0_.person_ID as person6_1_, addresses0_.ID as ID1_, addresses0_
⊟ initializeCollection(org.hibernate.collection.PersistentCollection	org.hibernate.impl.SessionImpl	
⊞ prepareStatement(java.lang.String)	org.apache.derby.client.am.Connection	select addresses0_.person_ID as person6_1_, addresses0_.ID as ID1_, addresses0_
executeQuery()	org.apache.derby.client.am.PreparedSta...	select addresses0_.person_ID as person6_1_, addresses0_.ID as ID1_, addresses0_
⊟ initializeCollection(org.hibernate.collection.PersistentCollection	org.hibernate.impl.SessionImpl	
⊞ prepareStatement(java.lang.String)	org.apache.derby.client.am.Connection	select addresses0_.person_ID as person6_1_, addresses0_.ID as ID1_, addresses0_
executeQuery()	org.apache.derby.client.am.PreparedSta...	select addresses0_.person_ID as person6_1_, addresses0_.ID as ID1_, addresses0_
⊟ initializeCollection(org.hibernate.collection.PersistentCollection	org.hibernate.impl.SessionImpl	
⊞ prepareStatement(java.lang.String)	org.apache.derby.client.am.Connection	select addresses0_.person_ID as person6_1_, addresses0_.ID as ID1_, addresses0_
executeQuery()	org.apache.derby.client.am.PreparedSta...	select addresses0_.person_ID as person6_1_, addresses0_.ID as ID1_, addresses0_
⊟ initializeCollection(org.hibernate.collection.PersistentCollection	org.hibernate.impl.SessionImpl	

Tree | Visualization | API Distribution

Anatomy of Performance and Scalability Problems

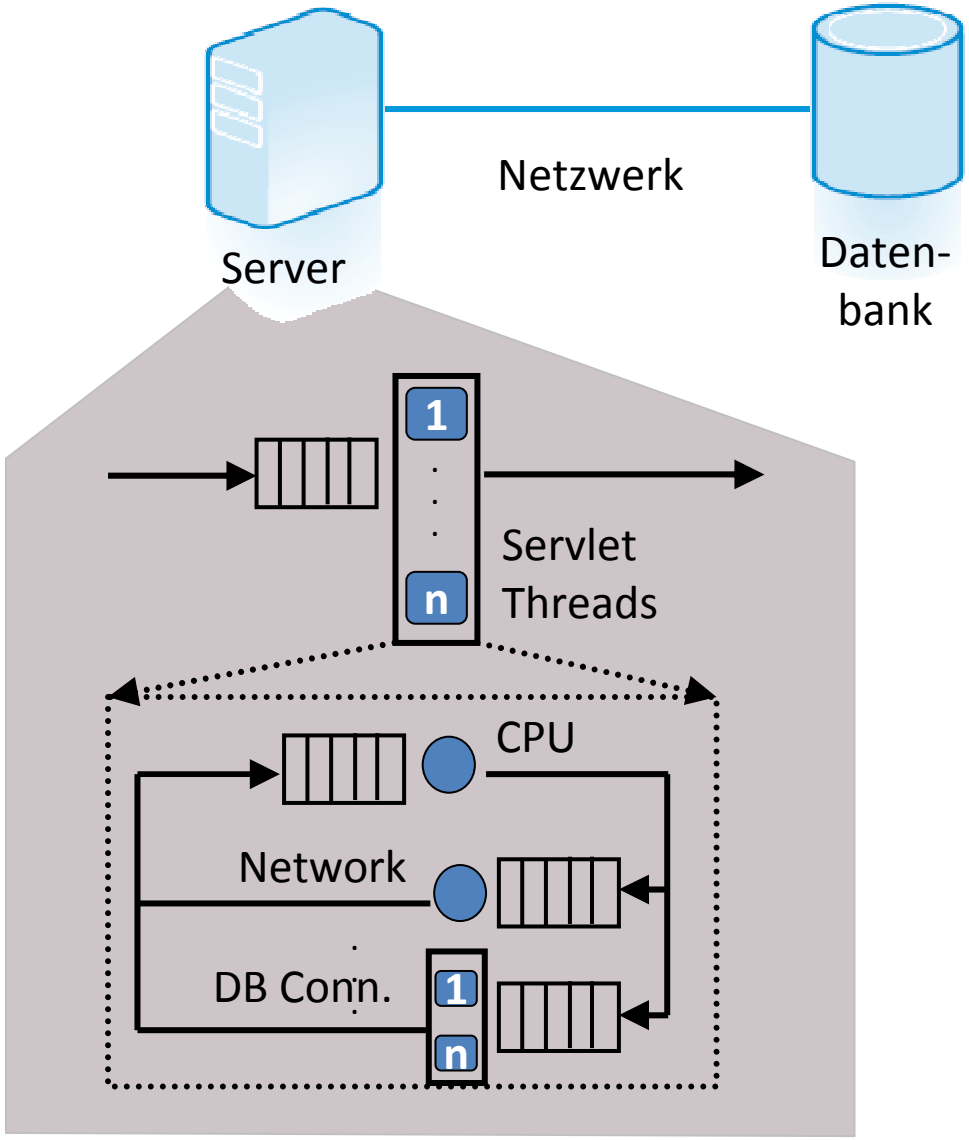


dynaTrace software

The Performance/Scalability View

- Queueing Theory
  - Modelling of bottleneck resource in a software systems
  - Modell resources as nodes and requests as queues
  - Example: CPU or number of current servlet request
- Queueing Networks
  - Model a computer system as a number of interconnected resources
  - Link resources together
  - Example: CPU, Network, Connection Pools, Servlet Threads, ...

A Web Application as a Queue Network

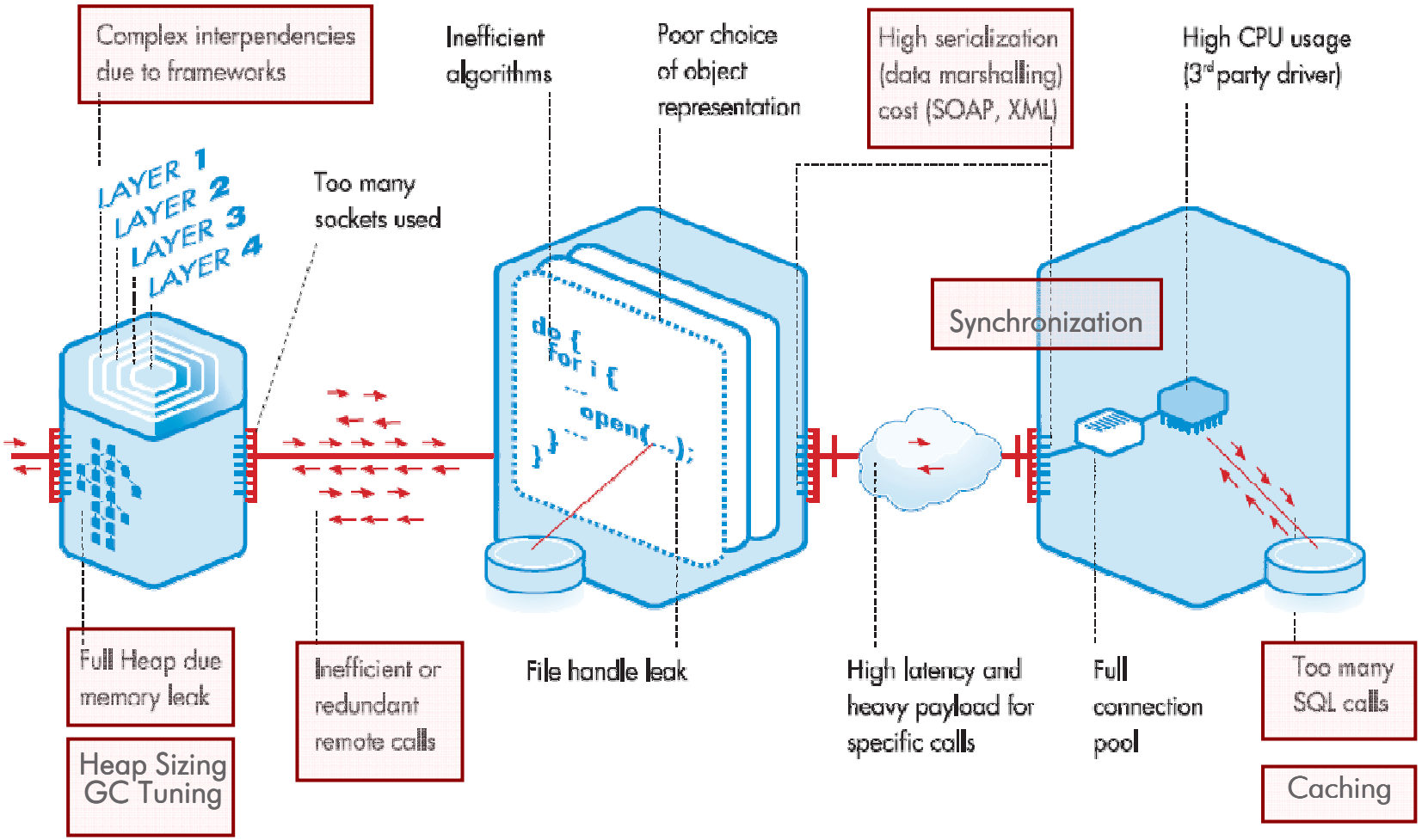


dynaTrace
software

Limiting Factors for Software Systems

- CPU Usage
  - Scale with Hardware
- Memory Usage
  - Scale with Hardware
- I/O and Network
  - Limited and difficult to scale
  - Architectural changes required
- Access to Shared Ressources
  - Difficult to scale or not scalable at all
  - Architectural changes required

Avoid Anti Patterns



dynaTrace software

How to integrate performance management into development?

dynaTrace software

What to do ... and when

- Developer Workplace
  - Impact analysis of code changes
  - Automatic verification against architectural rules
  - Verify that code change introduced no performance regressions
- Continuous Integration
  - Verify against earlier builds
  - Check ever use case so changes have no side effects
  - Monitoring at code level to see regressions early
- Load Testing
  - Run major performance test
  - Perform scalability and deployment tuning

Performance Mangement in Development

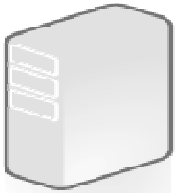
Developer

(1) Develop Feature

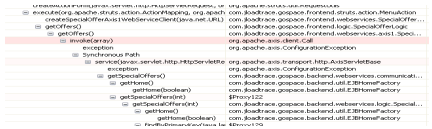
(2) Local Verifcation

(3) Commit Feature

CI System



(4a) Automated Build
(4b) Unit Test
(4c) Performance Tests

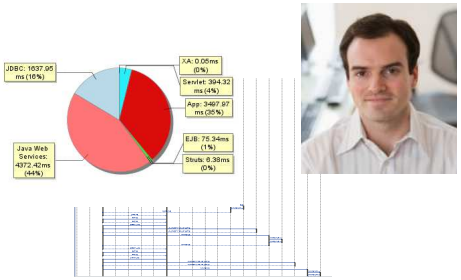


(5a) Realtime Peformance Feedback

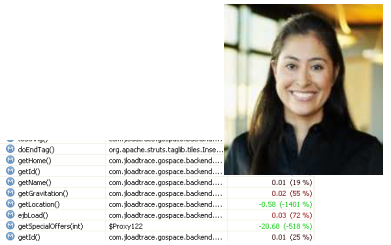
(5c) Realtime Performance Regression

(5b) Realtime Architecture Validation

Development Manager

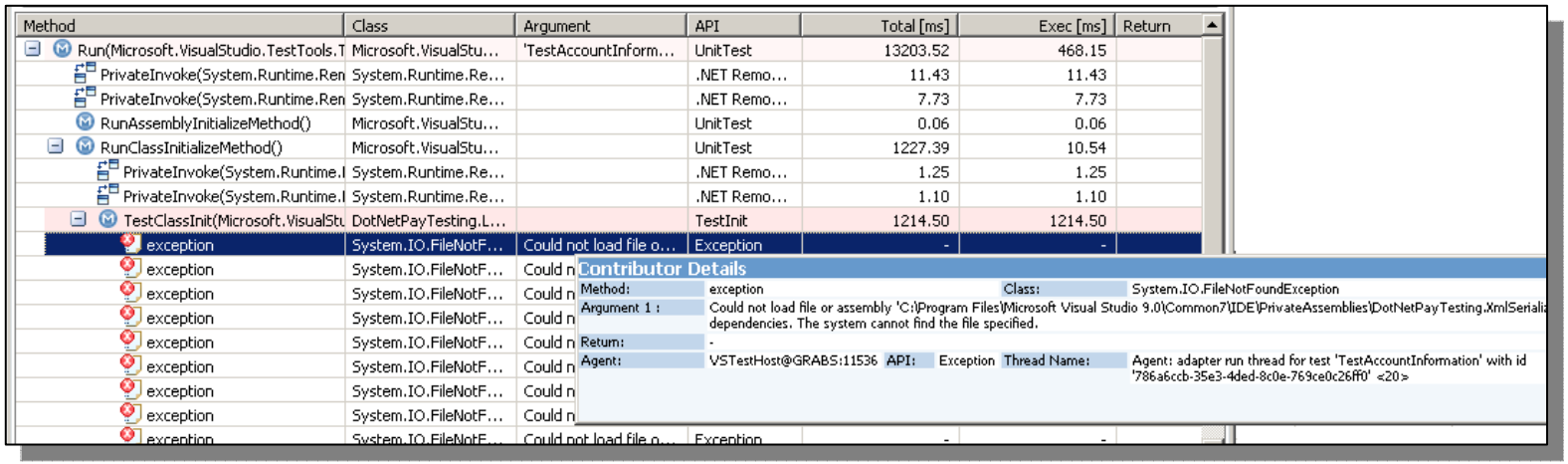


Test Automation



Architecture Validation with Unit Testing

- Unit tests only verify functionality
- Let your unit tests fail if
  - The same SQL Statement is executed more than once
  - Too many remoting rountrips
  - Certain exceptions are thrown and handled
  - Objects are not released
  - ...

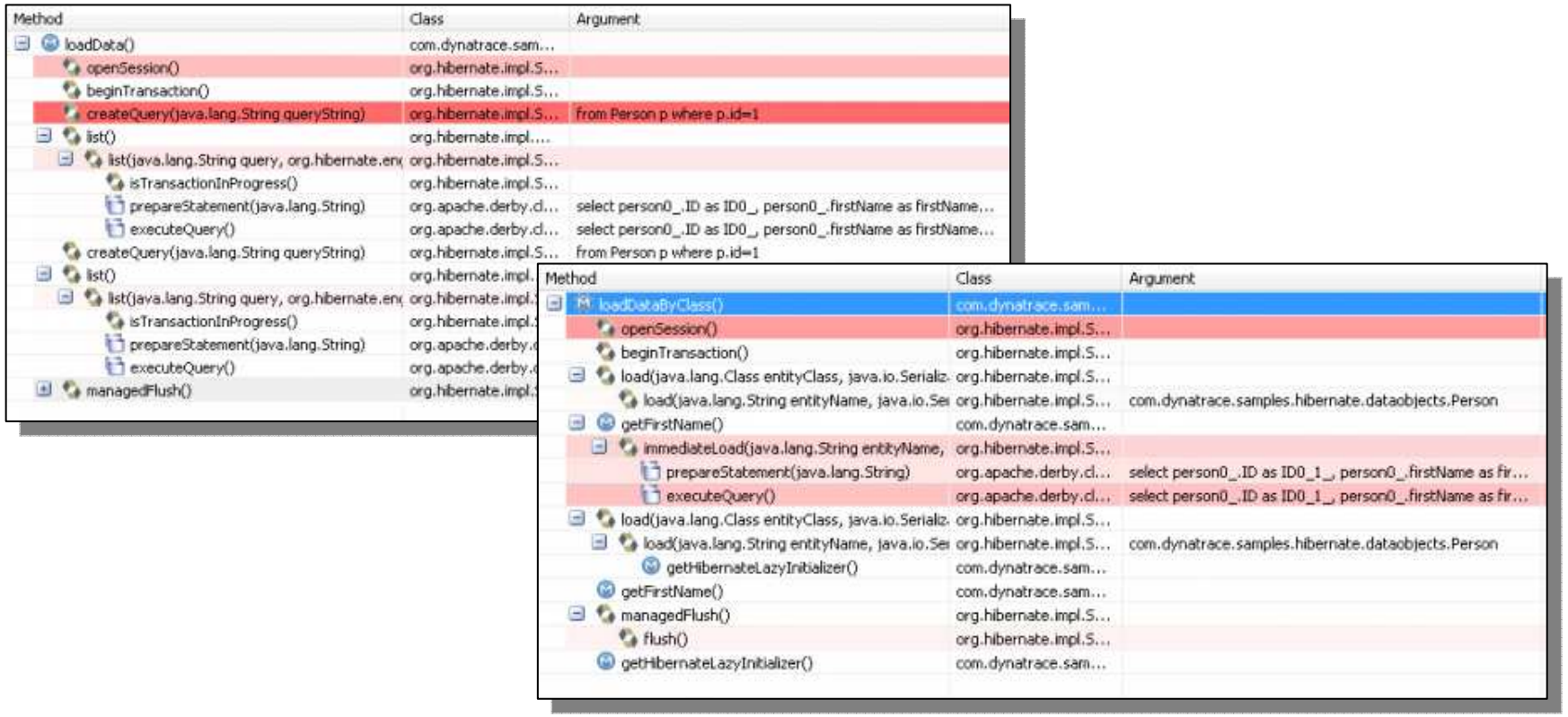


Analyze execution of each single unit test – identify architectural issues

dynaTrace software

Learn how foreign code works

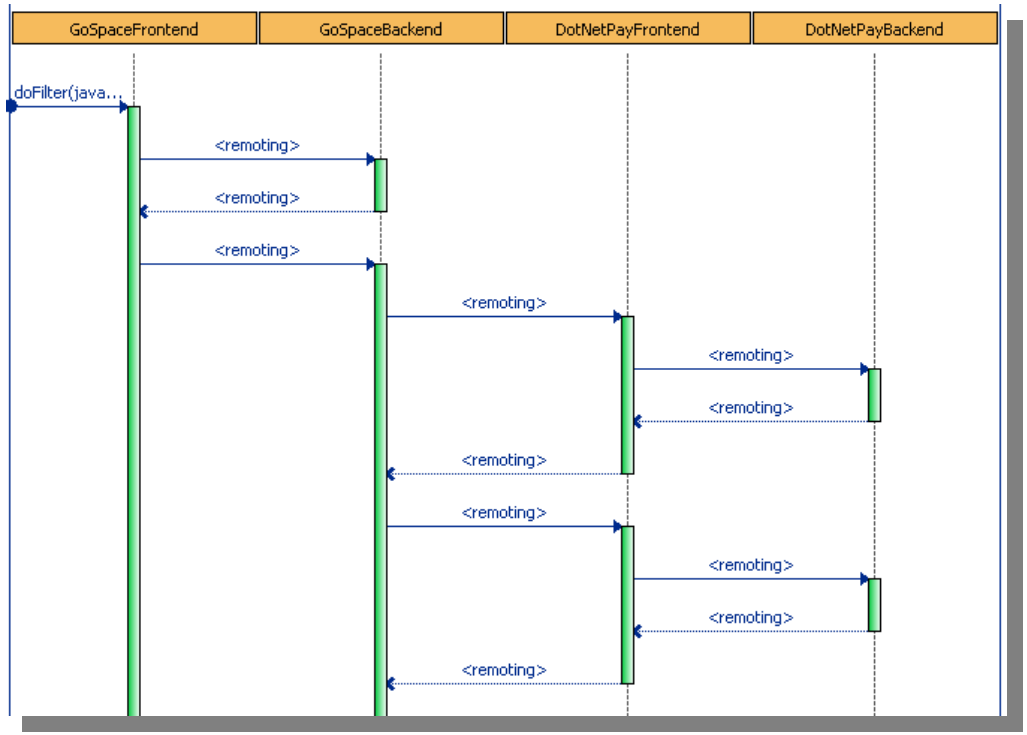
- Get insight into whats going on „under the hood“
  - Understand internals of the frameworks you use
  - Choose the right usage for your use case



dynaTrace software

Gain insight into Service Interactions

- Do you know how your services really interact?
  - Compare your model to the real world
  - Identify configuration issues
  - Analyze individual transaction flows



Analyze all service interactions for single transactions

dynaTrace software

Is it really that easy?

dynaTrace software

Well, there are some hurdles

- Some performance problems will not show up in development
- „Real-world“ components are missing
- Timing information only of limited value
- Performance test case design can be hard

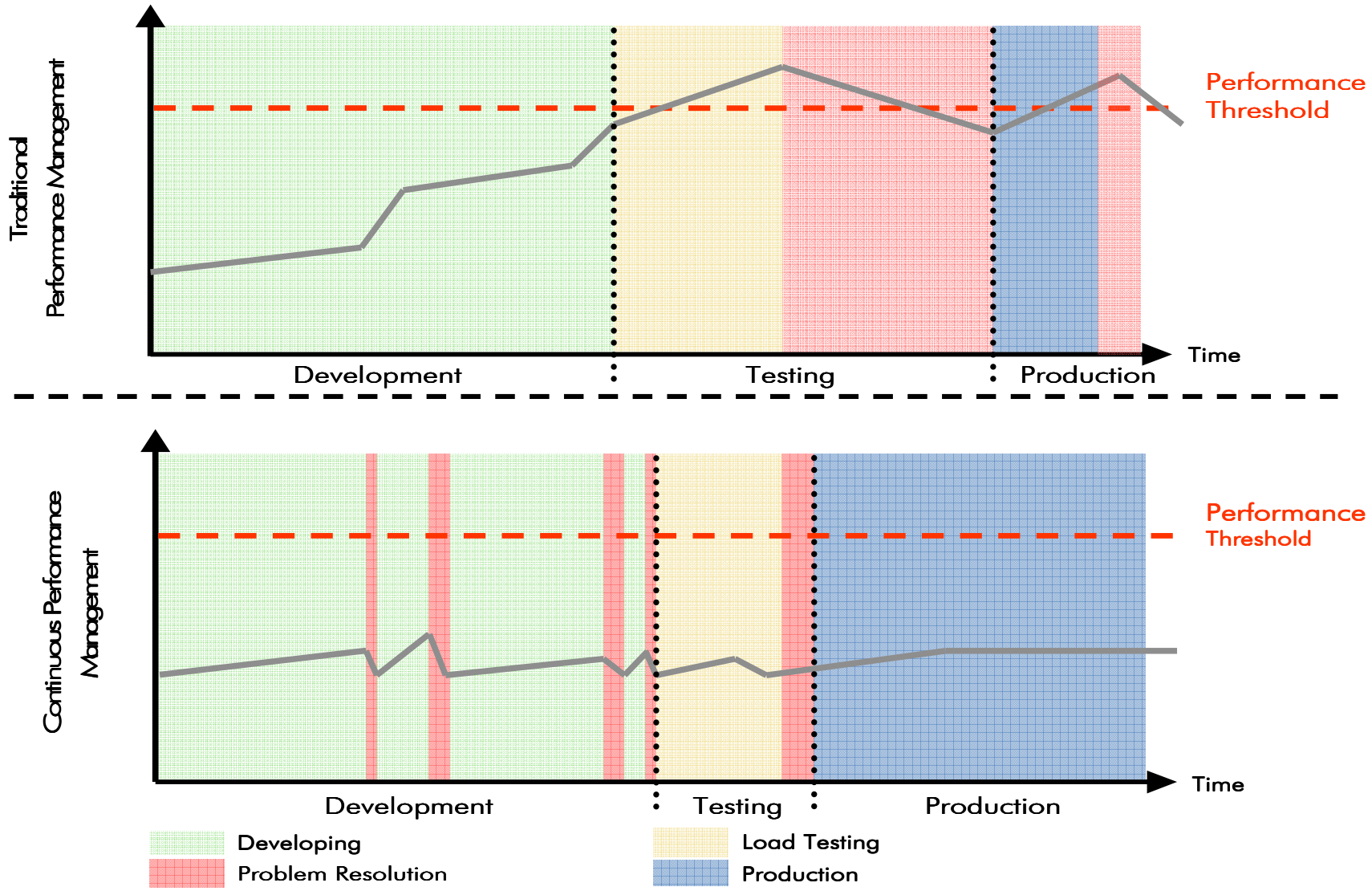
However, about 50 percent of performance problems can be found in early development stages

dynaTrace software

What‘s the benefit?

dynaTrace software

Avoid the iceberg



dynaTrace
software

Feel free ...

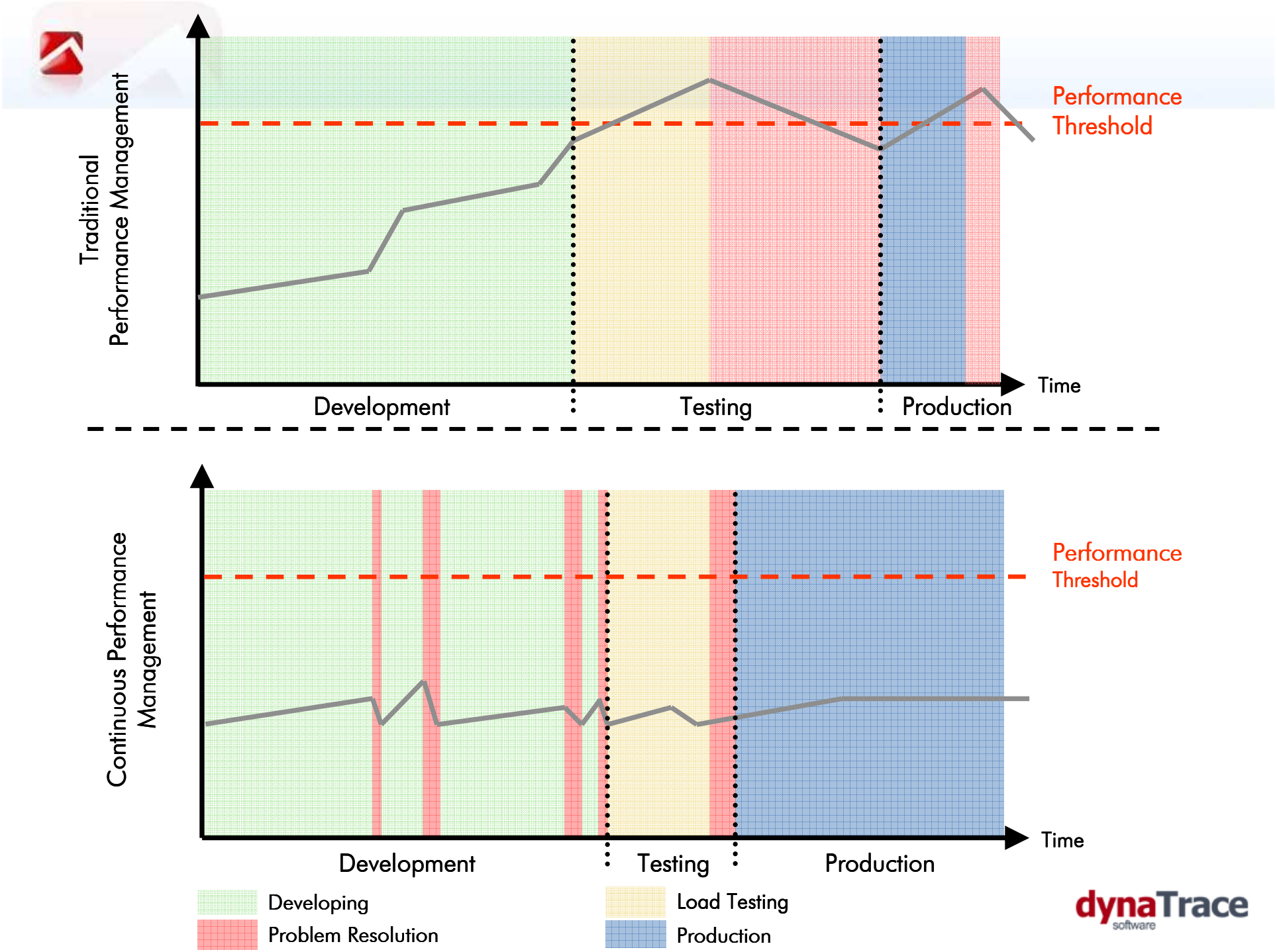
... to contact me

Alois Reitbauer

alois.reitbauer@dynatrace.com

http://blog.dynatrace.com

dynaTrace software


Traditional
Performance Management
Performance
Threshold
Time
Development
Testing
Production
Continuous Performance
Management
Performance
Threshold
Time
Development
Testing
Production
Developing
Problem Resolution
Load Testing
Production
dynaTrace
software