Software Quality Control

Wie stelle ich die Wartbarkeit von Code in Eigenentwicklung und Outsourcing sicher?

Dr. Elmar Juergens

CQSE
Continuous Quality in Software Engineering

Über Mich

Forschung

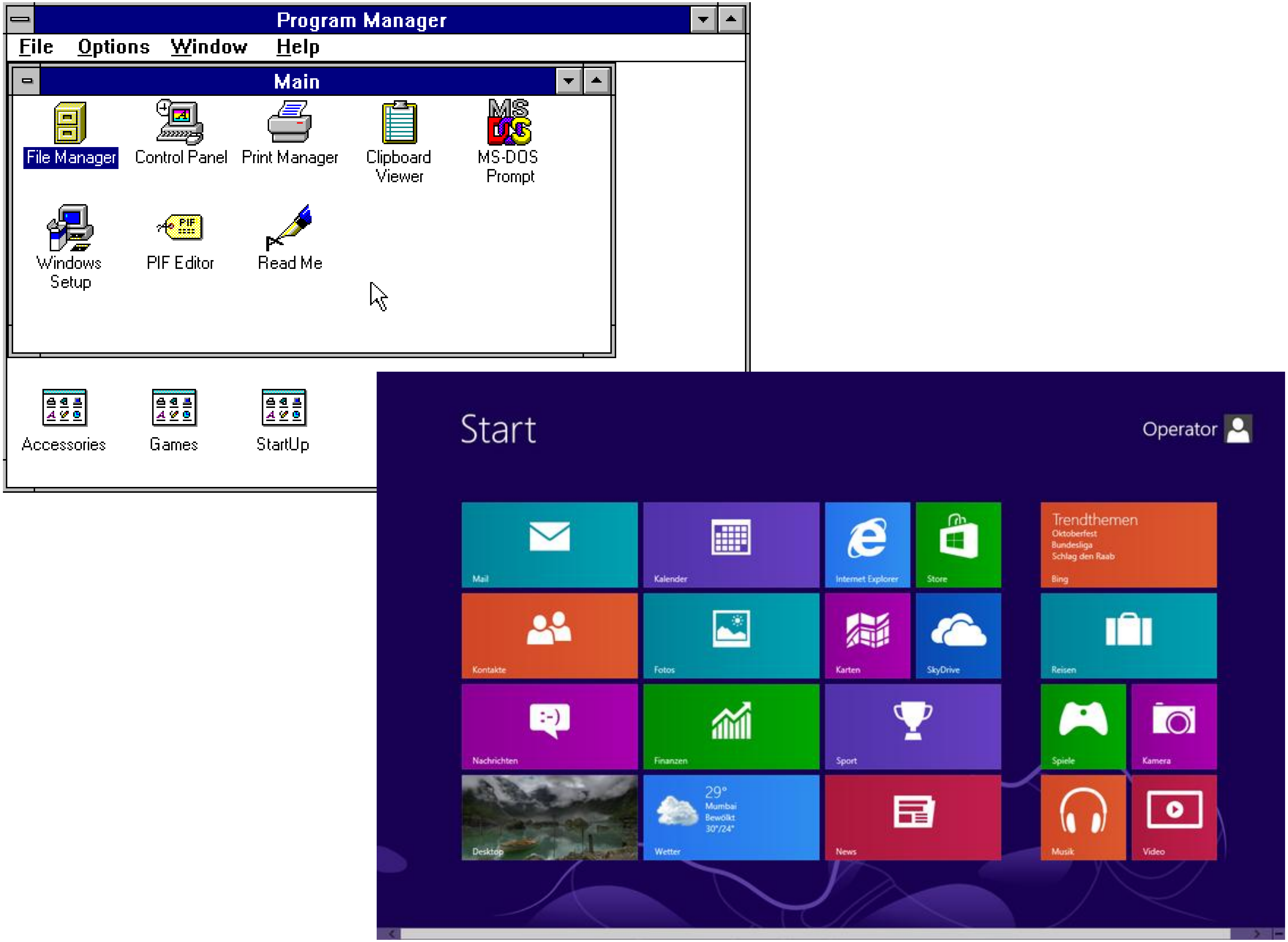
- Clone Detection
- Architekturanalyse

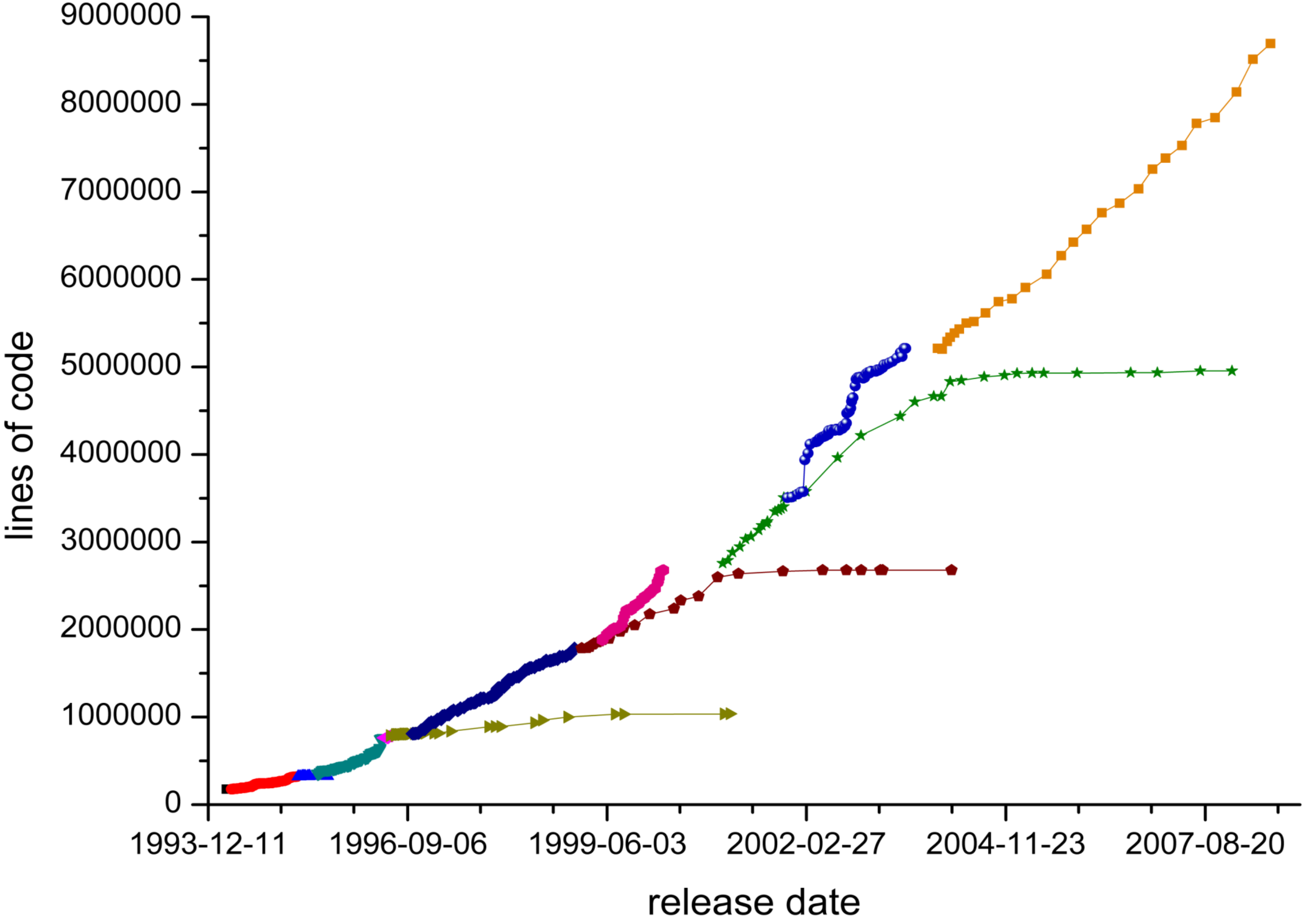
Beratung

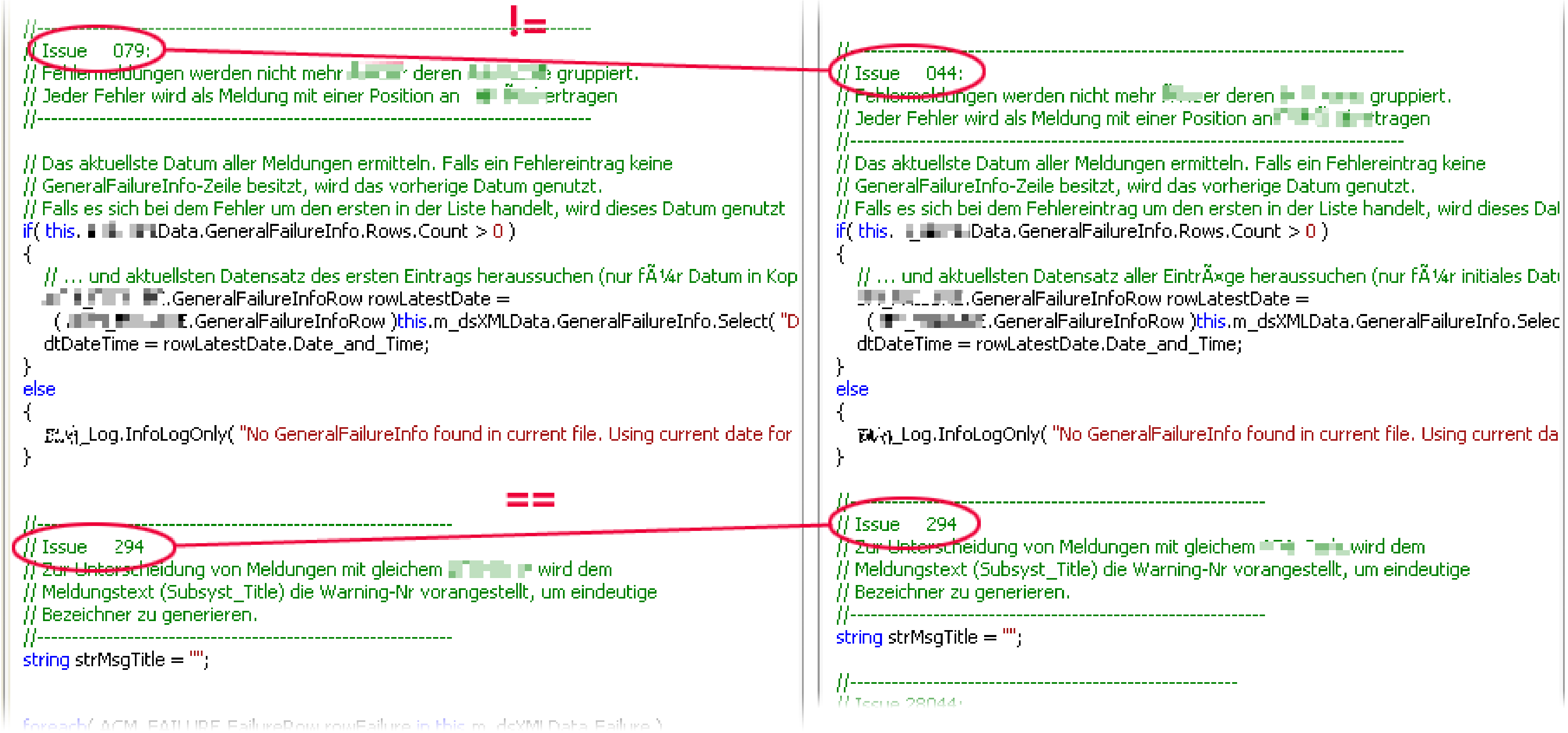
- Mitgründer
- Qualitäts-Bewertung & Qualitäts-Controlling

Entwicklung

- Continuous Quality Assessment Toolkit ConQAT
- >300 kLOC, Apache Lizenz, >20.000 Downloads

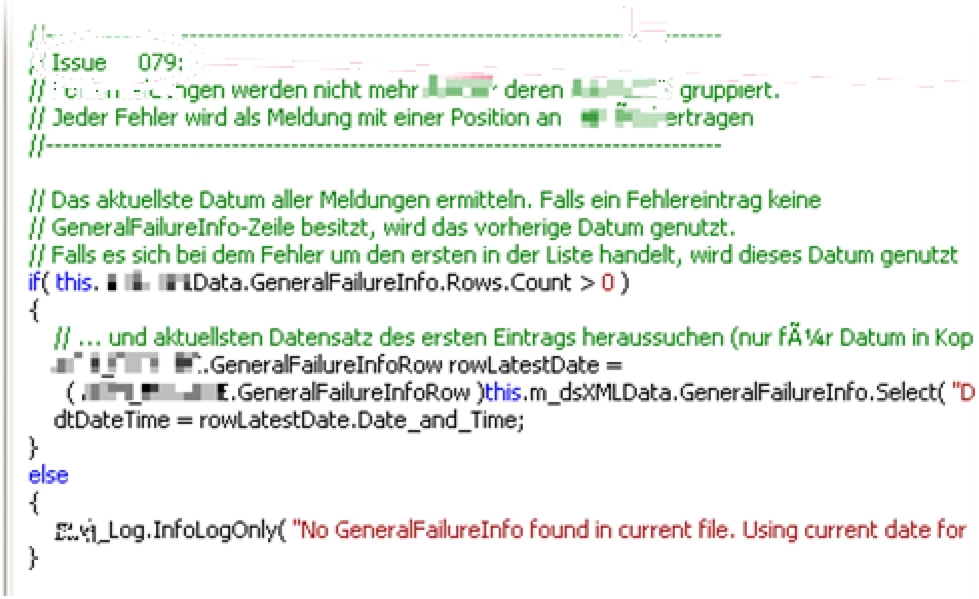

Program Manager
File
Options
Window
Help
Main
File Manager
Control Panel
Print Manager
Clipboard Viewer
MS-DOS Prompt
Windows Setup
PIF Editor
Read Me
Accessories
Games
StartUp
Start
Operator
Mail
Kalender
Internet Explorer
Store
Trendthemen
Oktoberfest
Bundesliga
Schlag den Raab
Bing
Kontakte
Fotos
Karten
SkyDrive
Reisen
Nachrichten
Finanzen
Sport
Spiele
Kamera
Desktop
29°
Mumbai
Bewölkt
30°/24°
Wetter
News
Musik
Video


9000000
8000000
7000000
6000000
5000000
4000000
3000000
2000000
1000000
0
lines of code
1993-12-11
1996-09-06
1999-06-03
2002-02-27
2004-11-23
2007-08-20
release date

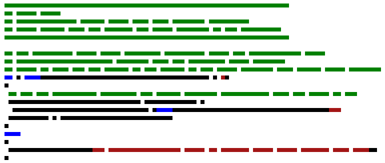


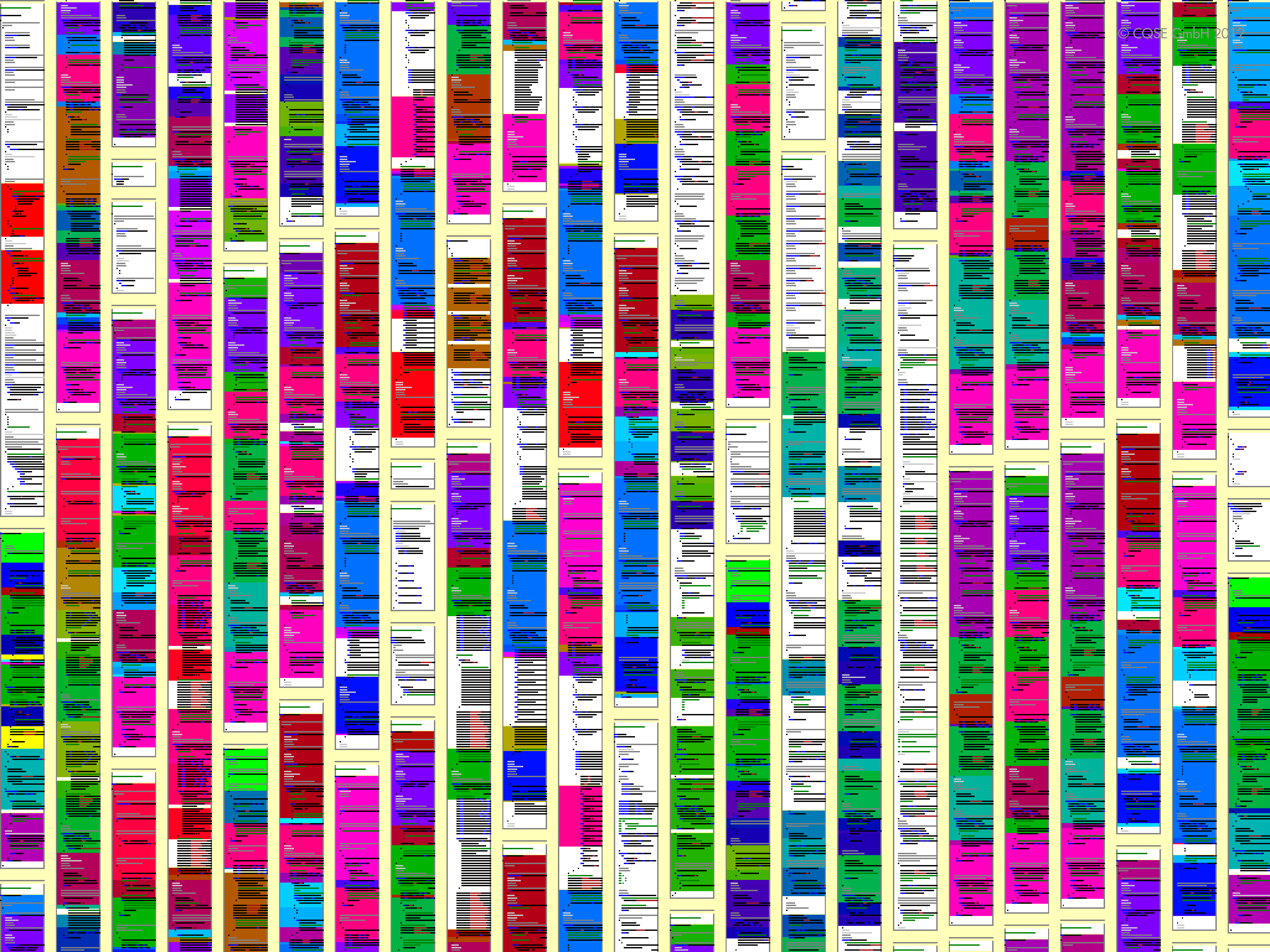
```
//----------------------------------------------------------------------------
// Issue    079:
// Fehlermeldungen werden nicht mehr [redacted] deren [redacted] gruppiert.
// Jeder Fehler wird als Meldung mit einer Position an [redacted]ertragen
//----------------------------------------------------------------------------

// Das aktuellste Datum aller Meldungen ermitteln. Falls ein Fehlereintrag keine
// GeneralFailureInfo-Zeile besitzt, wird das vorherige Datum genutzt.
// Falls es sich bei dem Fehler um den ersten in der Liste handelt, wird dieses Datum genutzt
if( this.[redacted]Data.GeneralFailureInfo.Rows.Count > 0 )
{
   // ... und aktuellsten Datensatz des ersten Eintrags heraussuchen (nur fÃ¼r Datum in Kop
   [redacted].GeneralFailureInfoRow rowLatestDate =
     ( [redacted].GeneralFailureInfoRow )this.m_dsXMLData.GeneralFailureInfo.Select( "D
   dtDateTime = rowLatestDate.Date_and_Time;
}
else
{
   [redacted]_Log.InfoLogOnly( "No GeneralFailureInfo found in current file. Using current date for
}


//------------------------------------------------------------
// Issue    294
// Zur Unterscheidung von Meldungen mit gleichem [redacted] wird dem
// Meldungstext (Subsyst_Title) die Warning-Nr vorangestellt, um eindeutige
// Bezeichner zu generieren.
//------------------------------------------------------------
string strMsgTitle = "";


foreach( ACM_FAILURE.FailureRow rowFailure in this.m_dsXMLData.Failure )
```

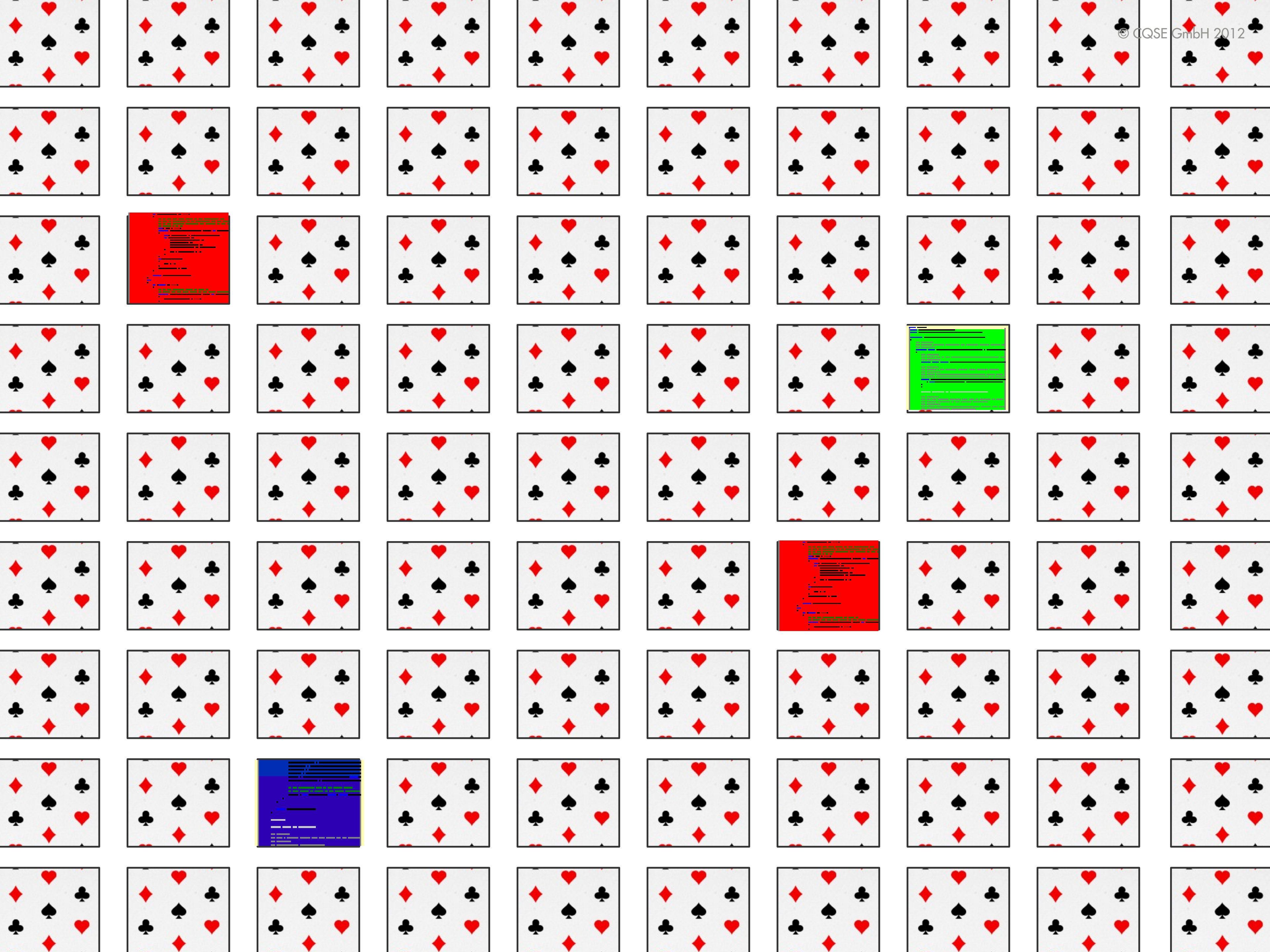
```
//----------------------------------------------------------------------------
// Issue    044:
// Fehlermeldungen werden nicht mehr [redacted]er deren [redacted] gruppiert.
// Jeder Fehler wird als Meldung mit einer Position an[redacted]tragen
//----------------------------------------------------------------------------
// Das aktuellste Datum aller Meldungen ermitteln. Falls ein Fehlereintrag keine
// GeneralFailureInfo-Zeile besitzt, wird das vorherige Datum genutzt.
// Falls es sich bei dem Fehlereintrag um den ersten in der Liste handelt, wird dieses Dat
if( this.[redacted]Data.GeneralFailureInfo.Rows.Count > 0 )
{
   // ... und aktuellsten Datensatz aller EintrÃ¤ge heraussuchen (nur fÃ¼r initiales Dat
   [redacted].GeneralFailureInfoRow rowLatestDate =
     ( [redacted].GeneralFailureInfoRow )this.m_dsXMLData.GeneralFailureInfo.Selec
   dtDateTime = rowLatestDate.Date_and_Time;
}
else
{
   [redacted]_Log.InfoLogOnly( "No GeneralFailureInfo found in current file. Using current da
}

//------------------------------------------------------------
// Issue    294
// Zur Unterscheidung von Meldungen mit gleichem [redacted] wird dem
// Meldungstext (Subsyst_Title) die Warning-Nr vorangestellt, um eindeutige
// Bezeichner zu generieren.
//------------------------------------------------------------
string strMsgTitle = "";

//------------------------------------------------------------
// Issue 28044:
```

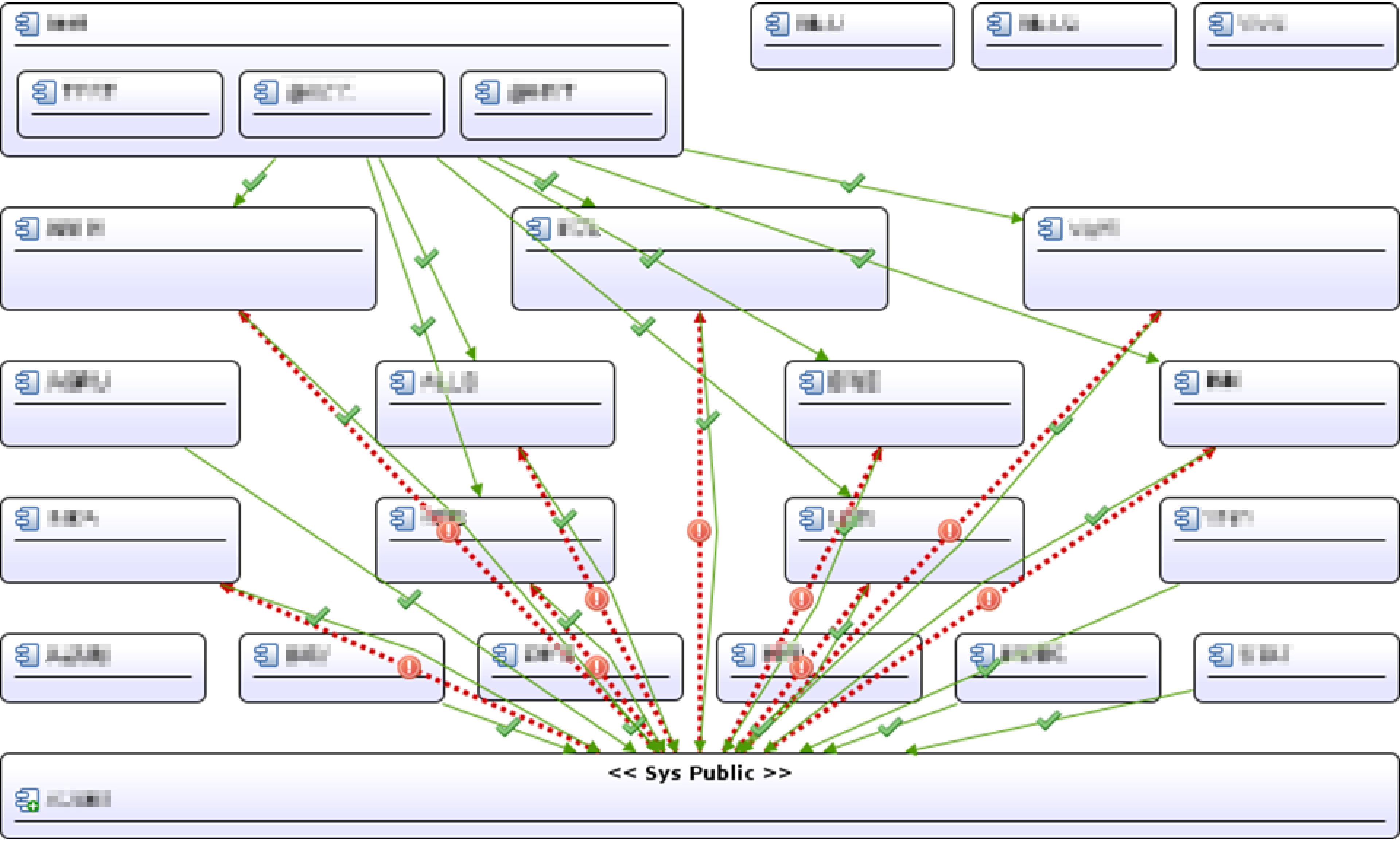


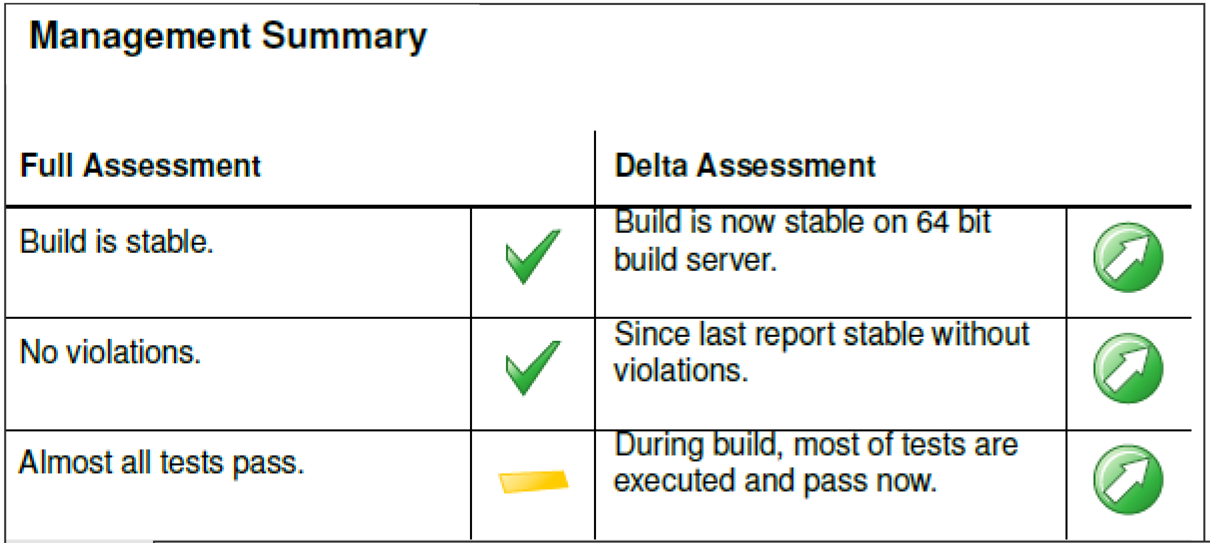
```
//[redacted]----------------------------------------------------------------------
// Issue    079:
// [redacted]ngen werden nicht mehr [redacted] deren [redacted] gruppiert.
// Jeder Fehler wird als Meldung mit einer Position an [redacted]ertragen
//--------------------------------------------------------------------------------

// Das aktuellste Datum aller Meldungen ermitteln. Falls ein Fehlereintrag keine
// GeneralFailureInfo-Zeile besitzt, wird das vorherige Datum genutzt.
// Falls es sich bei dem Fehler um den ersten in der Liste handelt, wird dieses Datum genutzt
if( this.[redacted]Data.GeneralFailureInfo.Rows.Count > 0 )
{
   // ... und aktuellsten Datensatz des ersten Eintrags heraussuchen (nur fÃ¼r Datum in Kop
   [redacted].GeneralFailureInfoRow rowLatestDate =
    ( [redacted].GeneralFailureInfoRow )this.m_dsXMLData.GeneralFailureInfo.Select( "D
   dtDateTime = rowLatestDate.Date_and_Time;
}
else
{
   [redacted]_Log.InfoLogOnly( "No GeneralFailureInfo found in current file. Using current date for
}
```





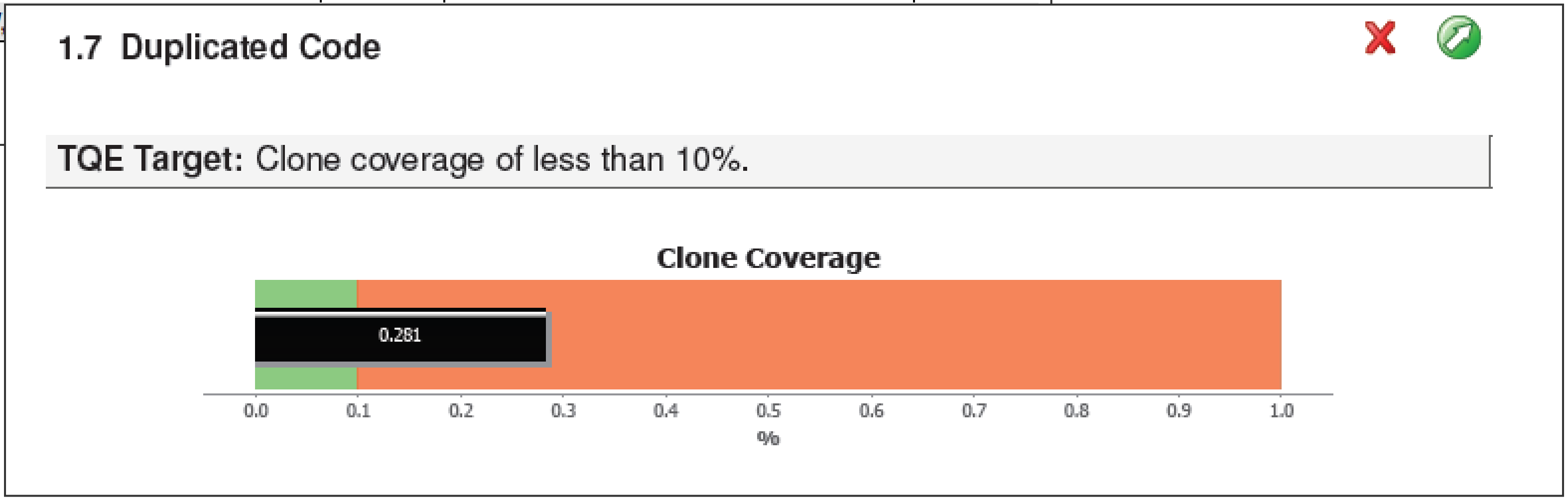



<< Sys Public >>



Management Summary

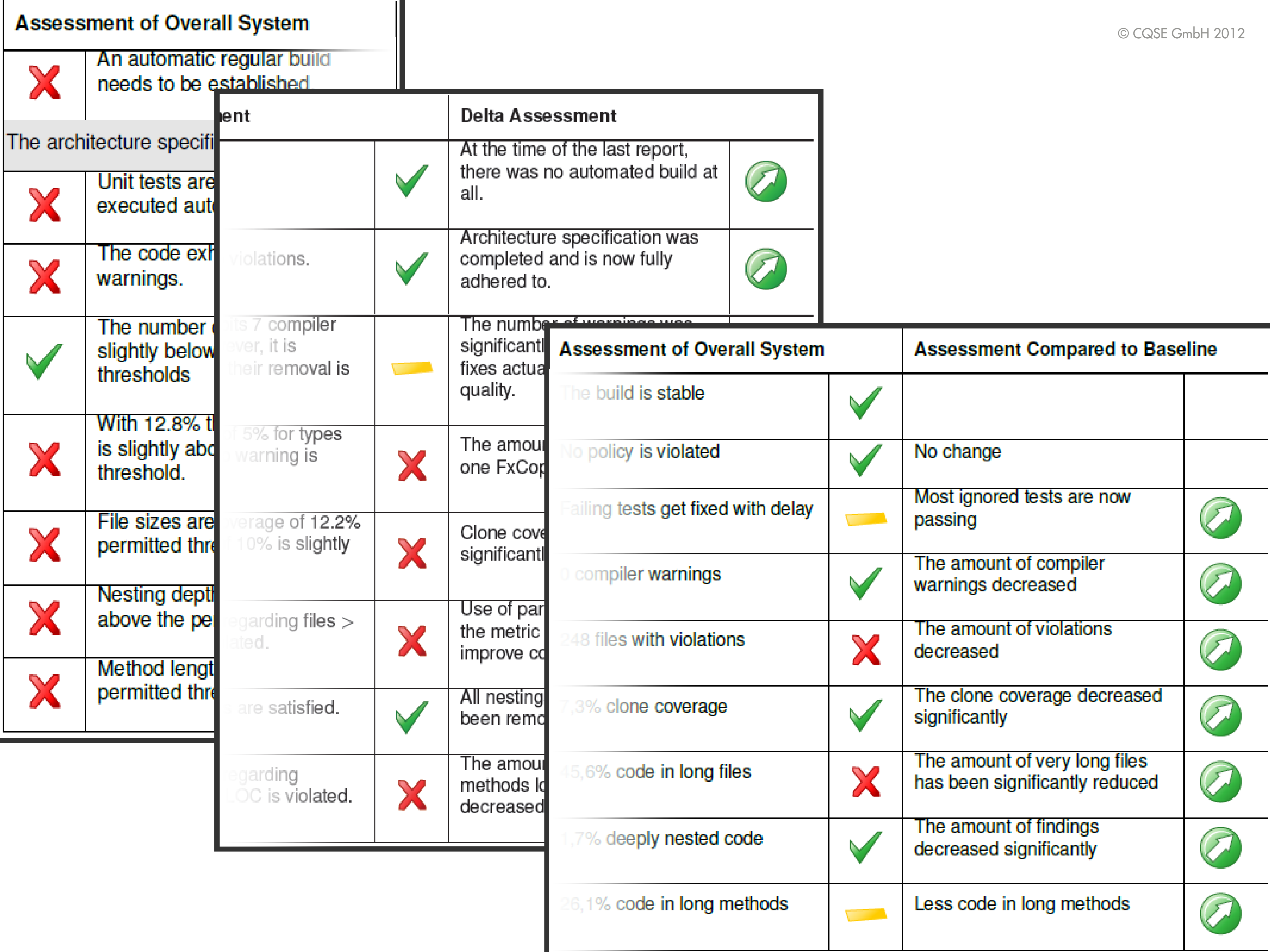
Full Assessment		Delta Assessment	
Build is stable.	✓	Build is now stable on 64 bit build server.	↗
No violations.	✓	Since last report stable without violations.	↗
Almost all tests pass.	▬	During build, most of tests are executed and pass now.	↗
Currently, Currently, occur.			



1.7 Duplicated Code

✗ ↗

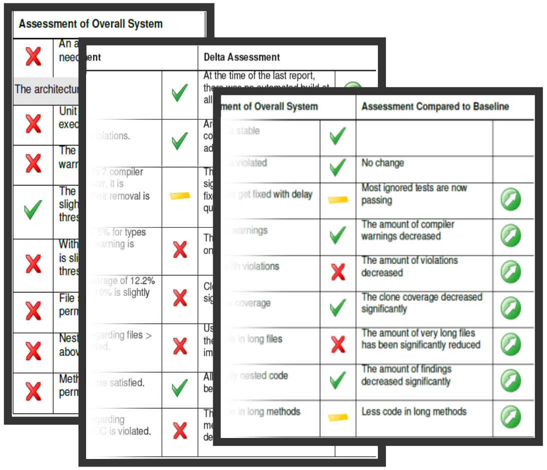
TQE Target: Clone coverage of less than 10%.



Assessment of Overall System	
✗	An automatic regular build needs to be established
The architecture specifi	
✗	Unit tests are executed aut
✗	The code exh warnings.
✓	The number slightly below thresholds
✗	With 12.8% t is slightly abo threshold.
✗	File sizes are permitted thre
✗	Nesting depth above the pe
✗	Method lengt permitted thre

ent		Delta Assessment	
	✓	At the time of the last report, there was no automated build at all.	↗
violations.	✓	Architecture specification was completed and is now fully adhered to.	↗
is 7 compiler ever, it is their removal is	▬	The number of warnings was significant fixes actua quality.	
of 5% for types warning is	✗	The amou one FxCop	
erage of 12.2% f 10% is slightly	✗	Clone cove significant	
egarding files > ated.	✗	Use of par the metric improve co	
s are satisfied.	✓	All nesting been remo	
egarding LOC is violated.	✗	The amou methods lo decreased	

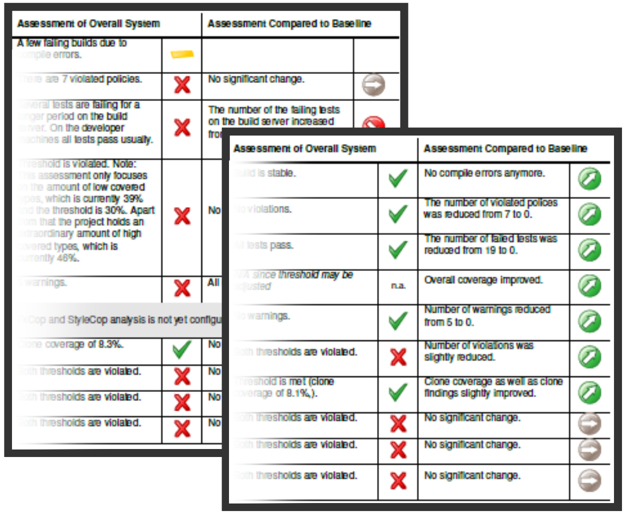
Assessment of Overall System		Assessment Compared to Baseline	
The build is stable	✓		
No policy is violated	✓	No change	
Failing tests get fixed with delay	▬	Most ignored tests are now passing	↗
0 compiler warnings	✓	The amount of compiler warnings decreased	↗
248 files with violations	✗	The amount of violations decreased	↗
7,3% clone coverage	✓	The clone coverage decreased significantly	↗
45,6% code in long files	✗	The amount of very long files has been significantly reduced	↗
1,7% deeply nested code	✓	The amount of findings decreased significantly	↗
26,1% code in long methods	▬	Less code in long methods	↗



Assessment of Overall System	
✗	An a neec
The architectur	
✗	Unit exec
✗	The warr
✓	The sligh thres
✗	With is sli thres
✗	File perm
✗	Nest abov
✗	Meth perm

ent		Delta Assessment	
	✓	At the time of the last report, there was no automated build at all	↗
violations.	✓	Ar co ad	
its 7 compiler ever, it is their removal is	▬	Th sig fix qu	
of 5% for types warning is	✗	Th on	
verage of 12.2% 10% is slightly	✗	Cl sig	
egarding files > lated.	✗	Us the im	
are satisfied.	✓	All be	
egarding LOC is violated.	✗	Th me de	

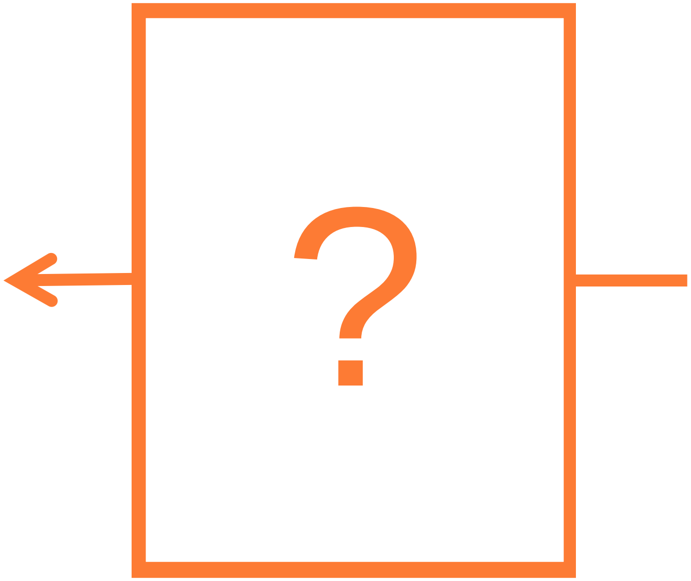
ment of Overall System		Assessment Compared to Baseline	
is stable	✓		
is violated	✓	No change	
ts get fixed with delay	▬	Most ignored tests are now passing	↗
r warnings	✓	The amount of compiler warnings decreased	↗
with violations	✗	The amount of violations decreased	↗
e coverage	✓	The clone coverage decreased significantly	↗
de in long files	✗	The amount of very long files has been significantly reduced	↗
ply nested code	✓	The amount of findings decreased significantly	↗
de in long methods	▬	Less code in long methods	↗

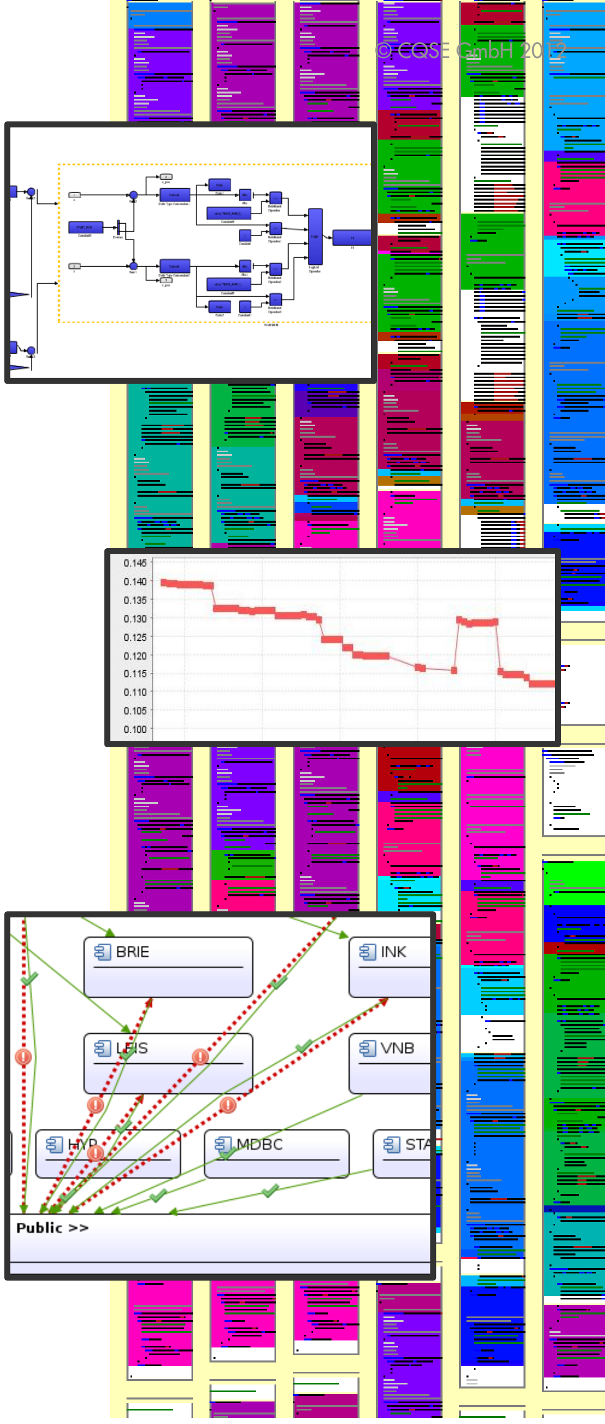


Assessment of Overall System		Assessment Compared to Baseline	
A few failing builds due to compile errors.	▬		
There are 7 violated policies.	✗	No significant change.	➡
Several tests are failing for a longer period on the build server. On the developer machines all tests pass usually.	✗	The number of the failing tests on the build server increased fro	↘
Threshold is violated. Note: This assessment only focuses on the amount of low covered types, which is currently 39% and the threshold is 30%. Apart from that the project holds an extraordinary amount of high covered types, which is currently 46%.	✗	No	
5 warnings.	✗	All	
FxCop and StyleCop analysis is not yet configu			
Clone coverage of 8.3%.	✓	No	
Both thresholds are violated.	✗	No	
Both thresholds are violated.	✗	No	
Both thresholds are violated.	✗	No	

Assessment of Overall System		Assessment Compared to Baseline	
Build is stable.	✓	No compile errors anymore.	↗
No violations.	✓	The number of violated polices was reduced from 7 to 0.	↗
All tests pass.	✓	The number of failed tests was reduced from 19 to 0.	↗
N/A since threshold may be adjusted	n.a.	Overall coverage improved.	↗
No warnings.	✓	Number of warnings reduced from 5 to 0.	↗
Both thresholds are violated.	✗	Number of violations was slightly reduced.	↗
Threshold is met (clone coverage of 8.1%,).	✓	Clone coverage as well as clone findings slightly improved.	↗
Both thresholds are violated.	✗	No significant change.	➡
Both thresholds are violated.	✗	No significant change.	➡
Both thresholds are violated.	✗	No significant change.	➡

• • •

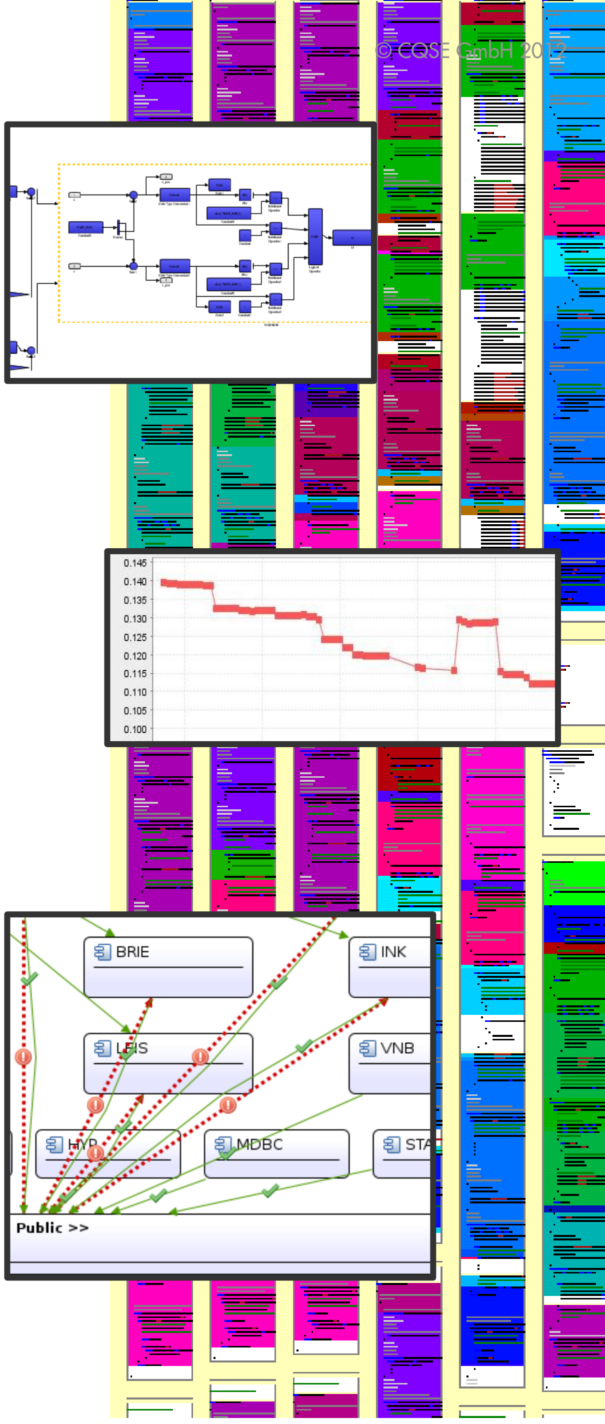

?


0.145
0.140
0.135
0.130
0.125
0.120
0.115
0.110
0.105
0.100
BRIE
INK
LEIS
VNB
HYP
MDBC
STA
Public >>

✓

▬

✗

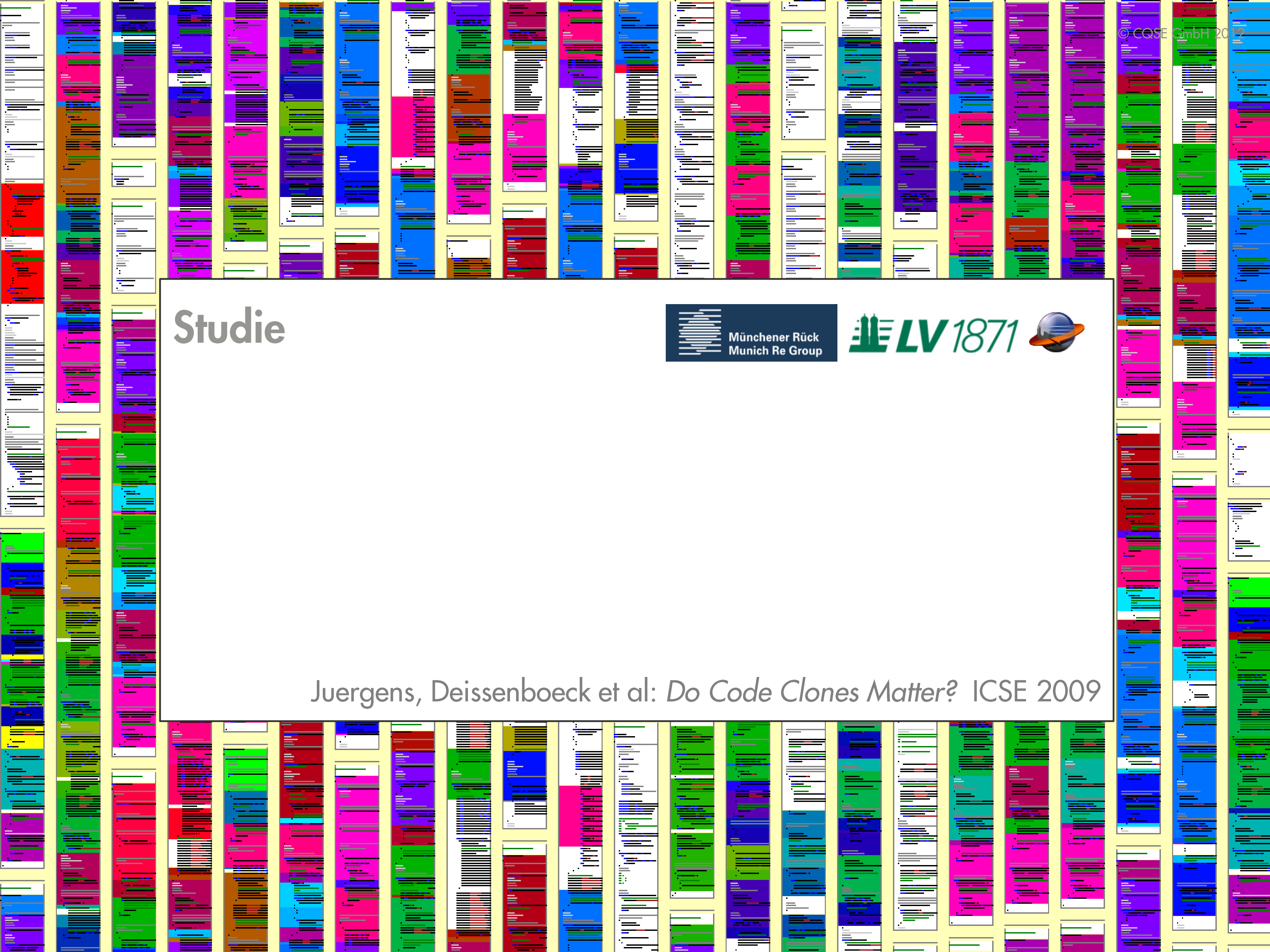

0.145
0.140
0.135
0.130
0.125
0.120
0.115
0.110
0.105
0.100
BRIE
INK
LEIS
VNB
HYP
MDBC
STA
Public >>

$$171 - 5.2 \cdot \ln(avgHV) - 0.23 \cdot avgCC(g') - 16.2 \cdot \ln(avgLOC) + 50 \cdot \sin(\mathrm{sqrt}(2.4 \cdot perCM))$$

?

HV:	Halstead Volume	CC:	Cyclomatic Complexity
LOC:	lines of code	perCM:	% Comment Lines

?

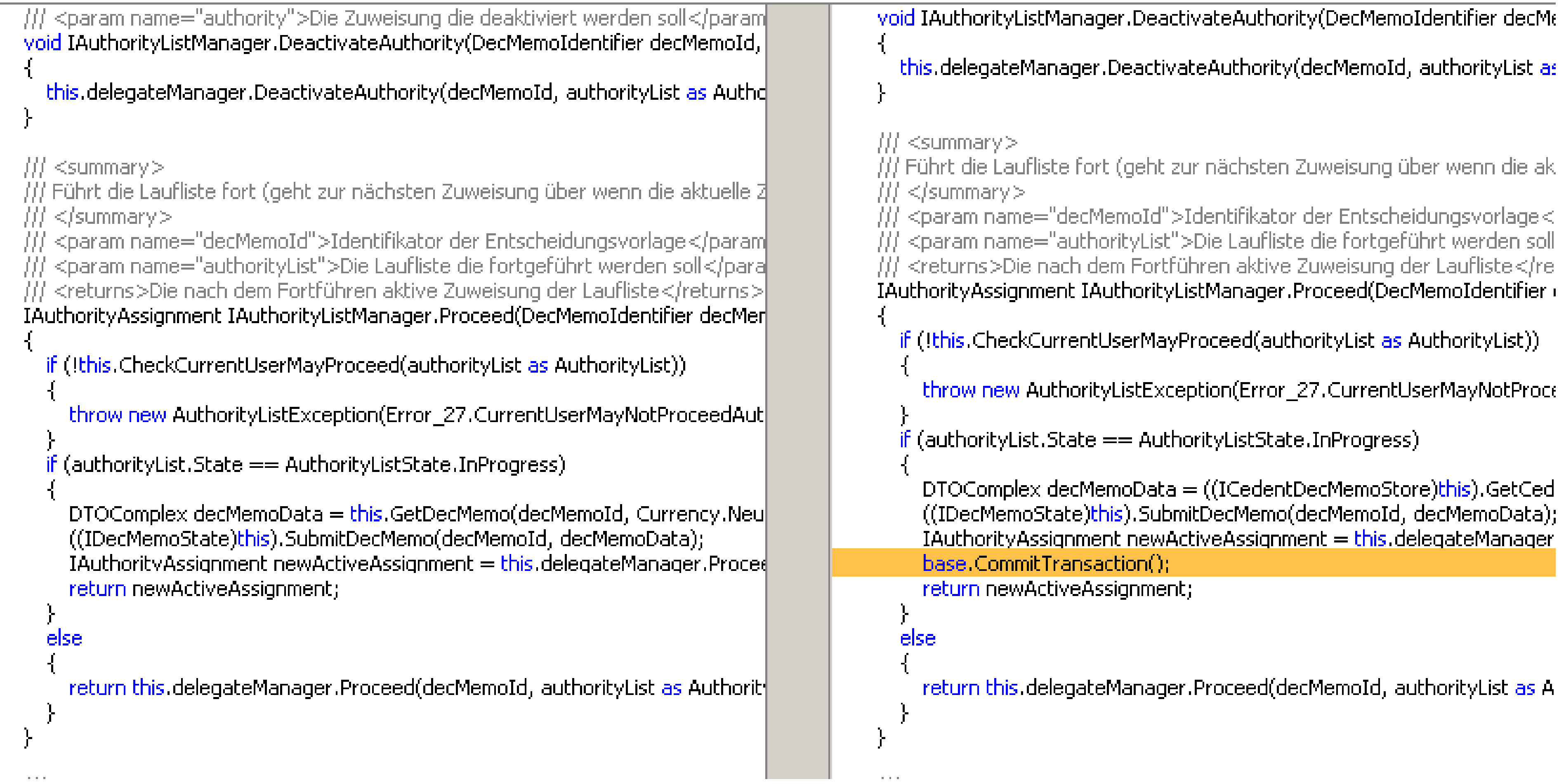


Studie

Münchener Rück
Munich Re Group

LV 1871

Juergens, Deissenboeck et al: *Do Code Clones Matter?* ICSE 2009



```
/// <param name="authority">Die Zuweisung die deaktiviert werden soll</param
void IAuthorityListManager.DeactivateAuthority(DecMemoIdentifier decMemoId,
{
    this.delegateManager.DeactivateAuthority(decMemoId, authorityList as Autho
}

/// <summary>
/// Führt die Laufliste fort (geht zur nächsten Zuweisung über wenn die aktuelle Z
/// </summary>
/// <param name="decMemoId">Identifikator der Entscheidungsvorlage</param
/// <param name="authorityList">Die Laufliste die fortgeführt werden soll</para
/// <returns>Die nach dem Fortführen aktive Zuweisung der Laufliste</returns>
IAuthorityAssignment IAuthorityListManager.Proceed(DecMemoIdentifier decMem
{
    if (!this.CheckCurrentUserMayProceed(authorityList as AuthorityList))
    {
        throw new AuthorityListException(Error_27.CurrentUserMayNotProceedAut
    }
    if (authorityList.State == AuthorityListState.InProgress)
    {
        DTOComplex decMemoData = this.GetDecMemo(decMemoId, Currency.Neu
        ((IDecMemoState)this).SubmitDecMemo(decMemoId, decMemoData);
        IAuthorityAssignment newActiveAssignment = this.delegateManager.Procee
        return newActiveAssignment;
    }
    else
    {
        return this.delegateManager.Proceed(decMemoId, authorityList as Authorit
    }
}
...
```

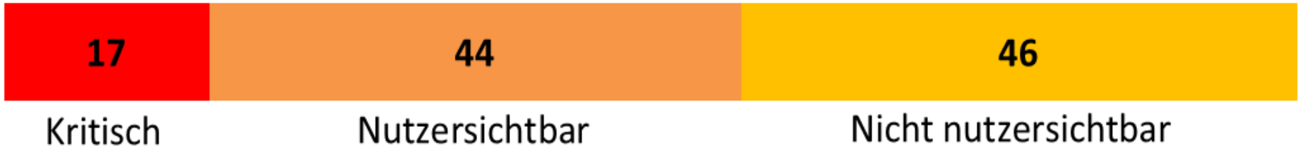
```
void IAuthorityListManager.DeactivateAuthority(DecMemoIdentifier decMe
{
    this.delegateManager.DeactivateAuthority(decMemoId, authorityList as
}

/// <summary>
/// Führt die Laufliste fort (geht zur nächsten Zuweisung über wenn die ak
/// </summary>
/// <param name="decMemoId">Identifikator der Entscheidungsvorlage<
/// <param name="authorityList">Die Laufliste die fortgeführt werden soll
/// <returns>Die nach dem Fortführen aktive Zuweisung der Laufliste</re
IAuthorityAssignment IAuthorityListManager.Proceed(DecMemoIdentifier
{
    if (!this.CheckCurrentUserMayProceed(authorityList as AuthorityList))
    {
        throw new AuthorityListException(Error_27.CurrentUserMayNotProce
    }
    if (authorityList.State == AuthorityListState.InProgress)
    {
        DTOComplex decMemoData = ((ICedentDecMemoStore)this).GetCed
        ((IDecMemoState)this).SubmitDecMemo(decMemoId, decMemoData);
        IAuthorityAssignment newActiveAssignment = this.delegateManager
        base.CommitTransaction();
        return newActiveAssignment;
    }
    else
    {
        return this.delegateManager.Proceed(decMemoId, authorityList as A
    }
}
...
```

Studie

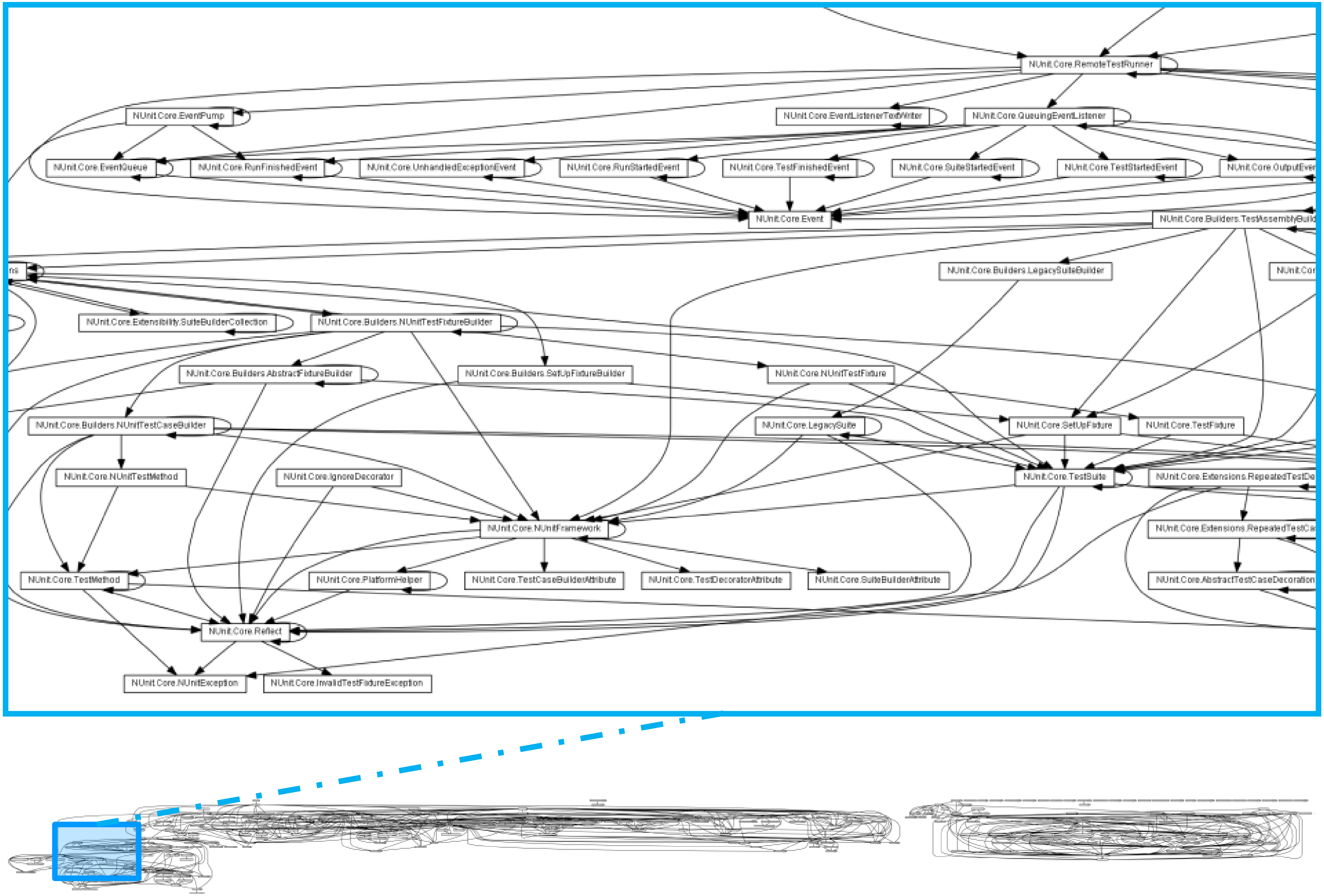
Münchener Rück
Munich Re Group

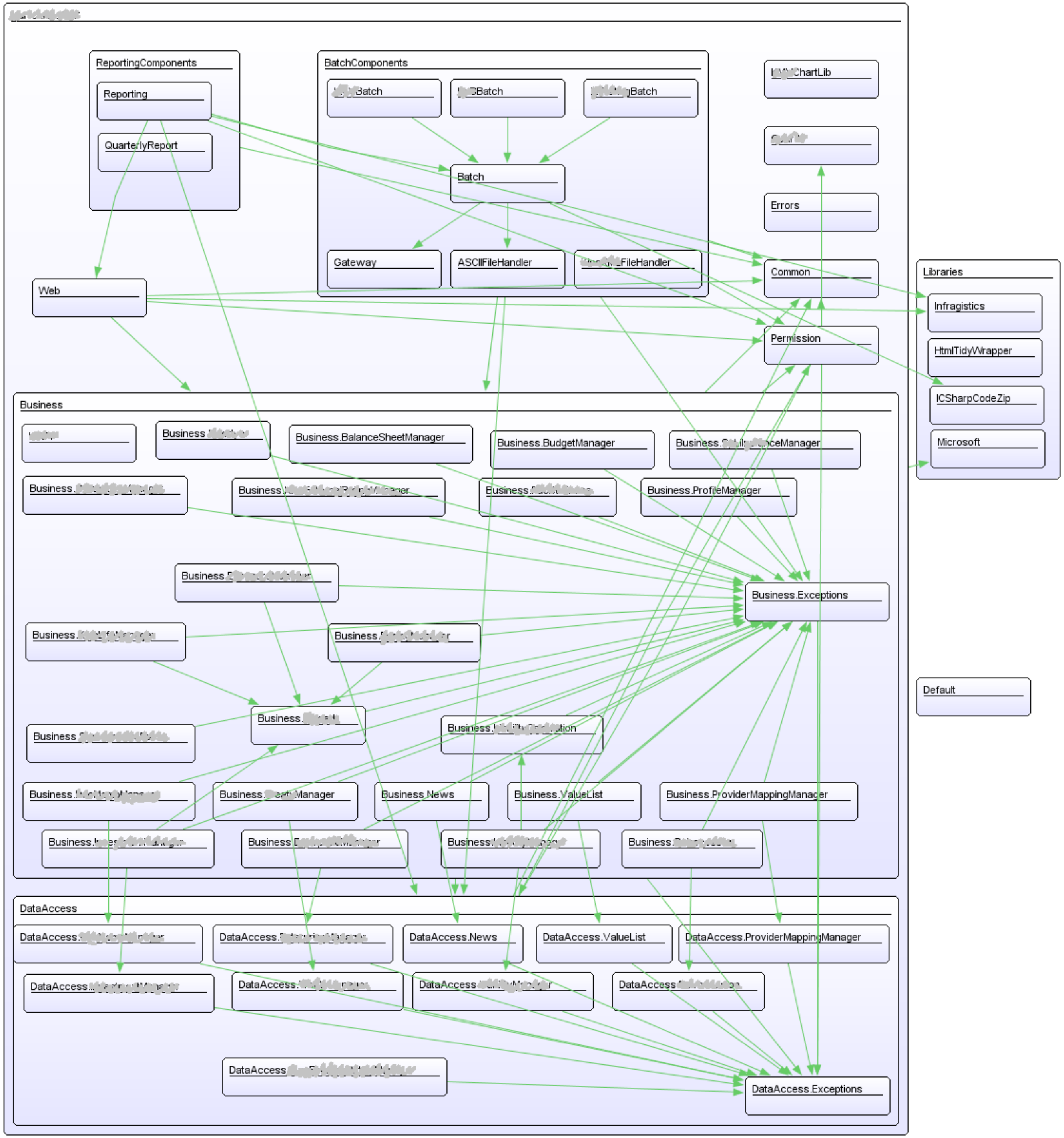
LV 1871

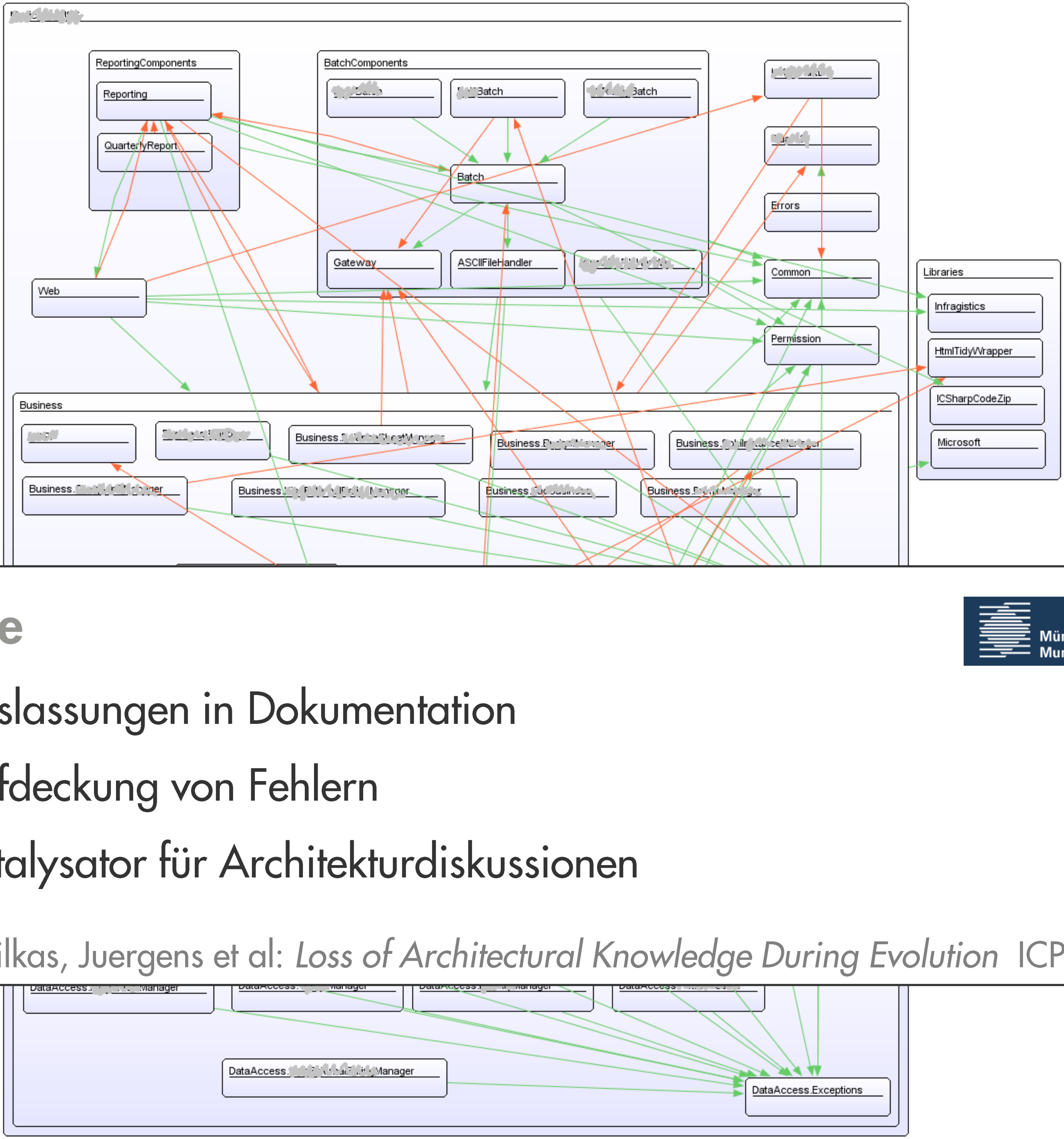
- Über 100 Fehler in produktiver Software



Juergens, Deissenboeck et al: *Do Code Clones Matter?* ICSE 2009


NUnit.Core.RemoteTestRunner
NUnit.Core.EventPump
NUnit.Core.EventListenerTextWriter
NUnit.Core.QueuingEventListener
NUnit.Core.EventQueue
NUnit.Core.RunFinishedEvent
NUnit.Core.UnhandledExceptionEvent
NUnit.Core.RunStartedEvent
NUnit.Core.TestFinishedEvent
NUnit.Core.SuiteStartedEvent
NUnit.Core.TestStartedEvent
NUnit.Core.OutputEver
NUnit.Core.Event
NUnit.Core.Builders.TestAssemblyBuild
NUnit.Core.Builders.LegacySuiteBuilder
NUnit.Cor
ns
NUnit.Core.Extensibility.SuiteBuilderCollection
NUnit.Core.Builders.NUnitTestFixtureBuilder
NUnit.Core.Builders.AbstractFixtureBuilder
NUnit.Core.Builders.SetUpFixtureBuilder
NUnit.Core.NUnitTestFixture
NUnit.Core.Builders.NUnitTestCaseBuilder
NUnit.Core.LegacySuite
NUnit.Core.SetUpFixture
NUnit.Core.TestFixture
NUnit.Core.NUnitTestMethod
NUnit.Core.IgnoreDecorator
NUnit.Core.TestSuite
NUnit.Core.Extensions.RepeatedTestDe
NUnit.Core.NUnitFramework
NUnit.Core.Extensions.RepeatedTestCa
NUnit.Core.TestMethod
NUnit.Core.PlatformHelper
NUnit.Core.TestCaseBuilderAttribute
NUnit.Core.TestDecoratorAttribute
NUnit.Core.SuiteBuilderAttribute
NUnit.Core.AbstractTestCaseDecoration
NUnit.Core.Reflect
NUnit.Core.NUnitException
NUnit.Core.InvalidTestFixtureException


[redacted]
ReportingComponents
Reporting
QuarterlyReport
BatchComponents
[redacted]Batch
[redacted]Batch
[redacted]Batch
Batch
Gateway
ASCIIFileHandler
[redacted]FileHandler
[redacted]ChartLib
[redacted]
Errors
Common
Permission
Web
Libraries
Infragistics
HtmlTidyWrapper
ICSharpCodeZip
Microsoft
Default
Business
[redacted]
Business.[redacted]
Business.BalanceSheetManager
Business.BudgetManager
Business.[redacted]ceManager
Business.[redacted]
Business.[redacted]
Business.[redacted]
Business.ProfileManager
Business.[redacted]
Business.Exceptions
Business.[redacted]
Business.[redacted]
Business.[redacted]
Business.[redacted]
Business.[redacted]tion
Business.[redacted]
Business.[redacted]Manager
Business.News
Business.ValueList
Business.ProviderMappingManager
Business.[redacted]
Business.[redacted]
Business.[redacted]
Business.[redacted]
DataAccess
DataAccess.[redacted]
DataAccess.[redacted]
DataAccess.News
DataAccess.ValueList
DataAccess.ProviderMappingManager
DataAccess.[redacted]
DataAccess.[redacted]
DataAccess.[redacted]
DataAccess.[redacted]
DataAccess.[redacted]
DataAccess.Exceptions

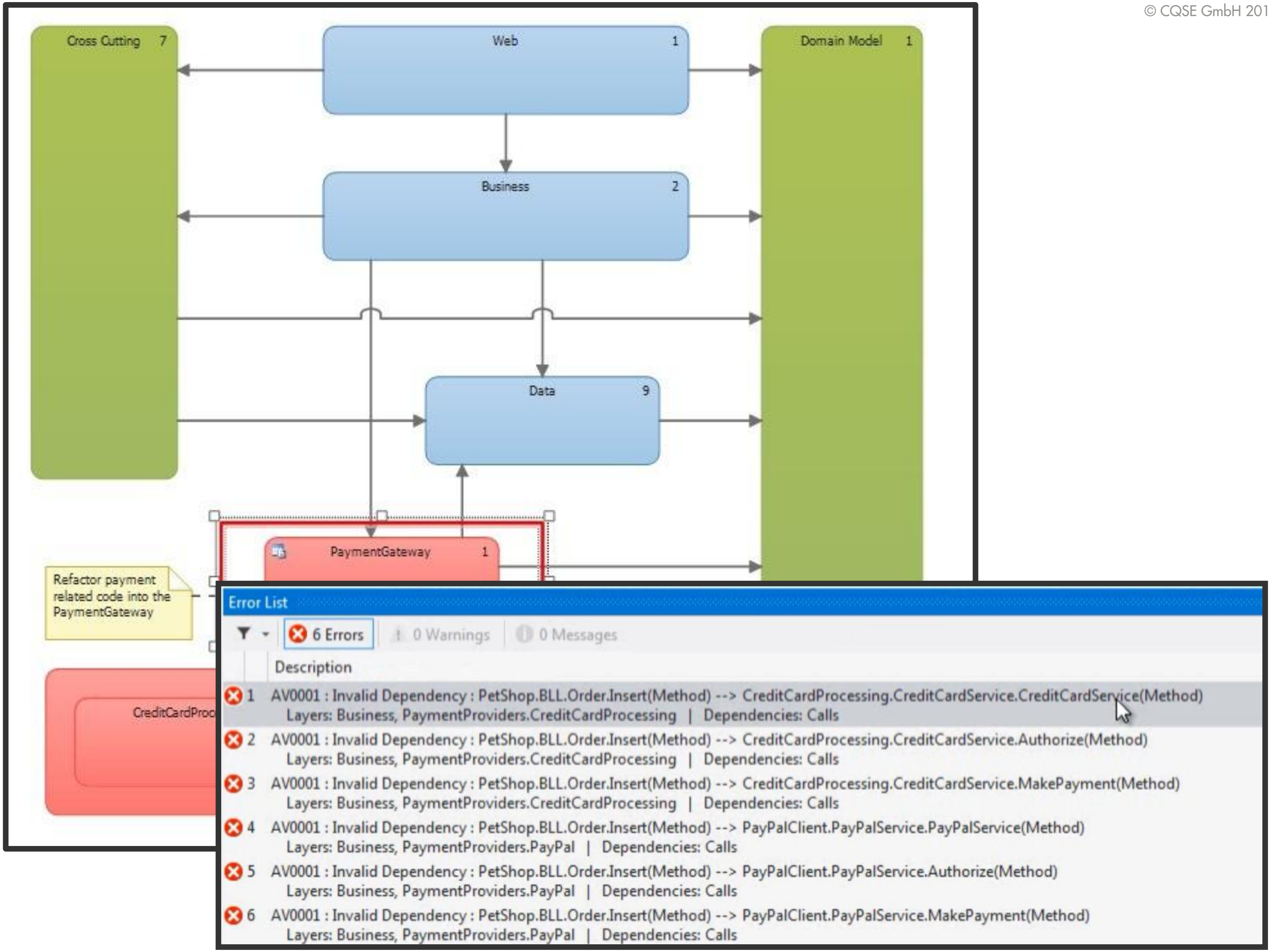


Studie

Münchener Rück
Munich Re Group

- Auslassungen in Dokumentation
- Aufdeckung von Fehlern
- Katalysator für Architekturdiskussionen

Feilkas, Juergens et al: *Loss of Architectural Knowledge During Evolution* ICPC 2009



1 AV0001 : Invalid Dependency : PetShop.BLL.Order.Insert(Method) --> CreditCardProcessing.CreditCardService.CreditCardService(Method)
Layers: Business, PaymentProviders.CreditCardProcessing | Dependencies: Calls

2 AV0001 : Invalid Dependency : PetShop.BLL.Order.Insert(Method) --> CreditCardProcessing.CreditCardService.Authorize(Method)
Layers: Business, PaymentProviders.CreditCardProcessing | Dependencies: Calls

3 AV0001 : Invalid Dependency : PetShop.BLL.Order.Insert(Method) --> CreditCardProcessing.CreditCardService.MakePayment(Method)
Layers: Business, PaymentProviders.CreditCardProcessing | Dependencies: Calls

4 AV0001 : Invalid Dependency : PetShop.BLL.Order.Insert(Method) --> PayPalClient.PayPalService.PayPalService(Method)
Layers: Business, PaymentProviders.PayPal | Dependencies: Calls

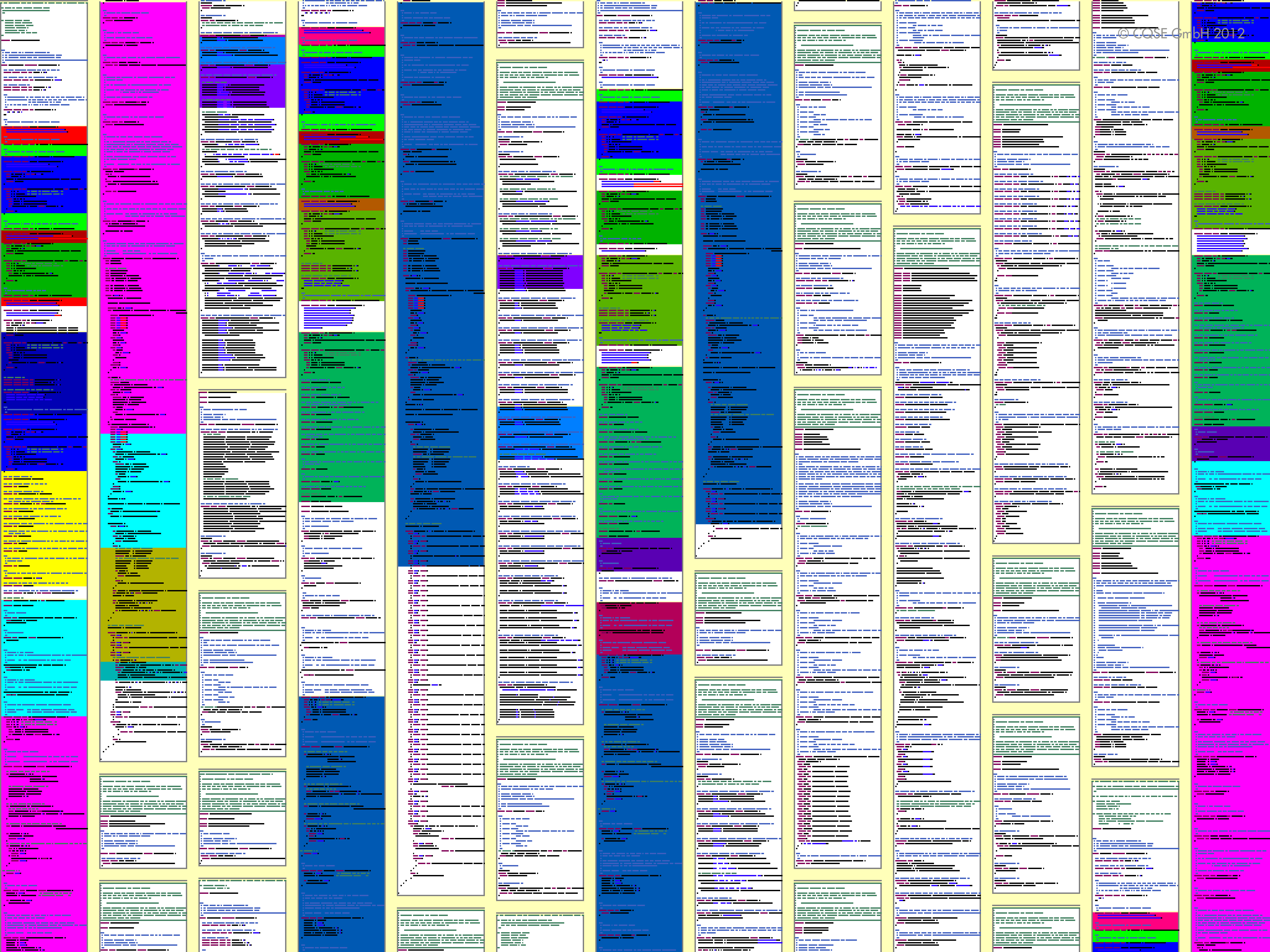
5 AV0001 : Invalid Dependency : PetShop.BLL.Order.Insert(Method) --> PayPalClient.PayPalService.Authorize(Method)
Layers: Business, PaymentProviders.PayPal | Dependencies: Calls

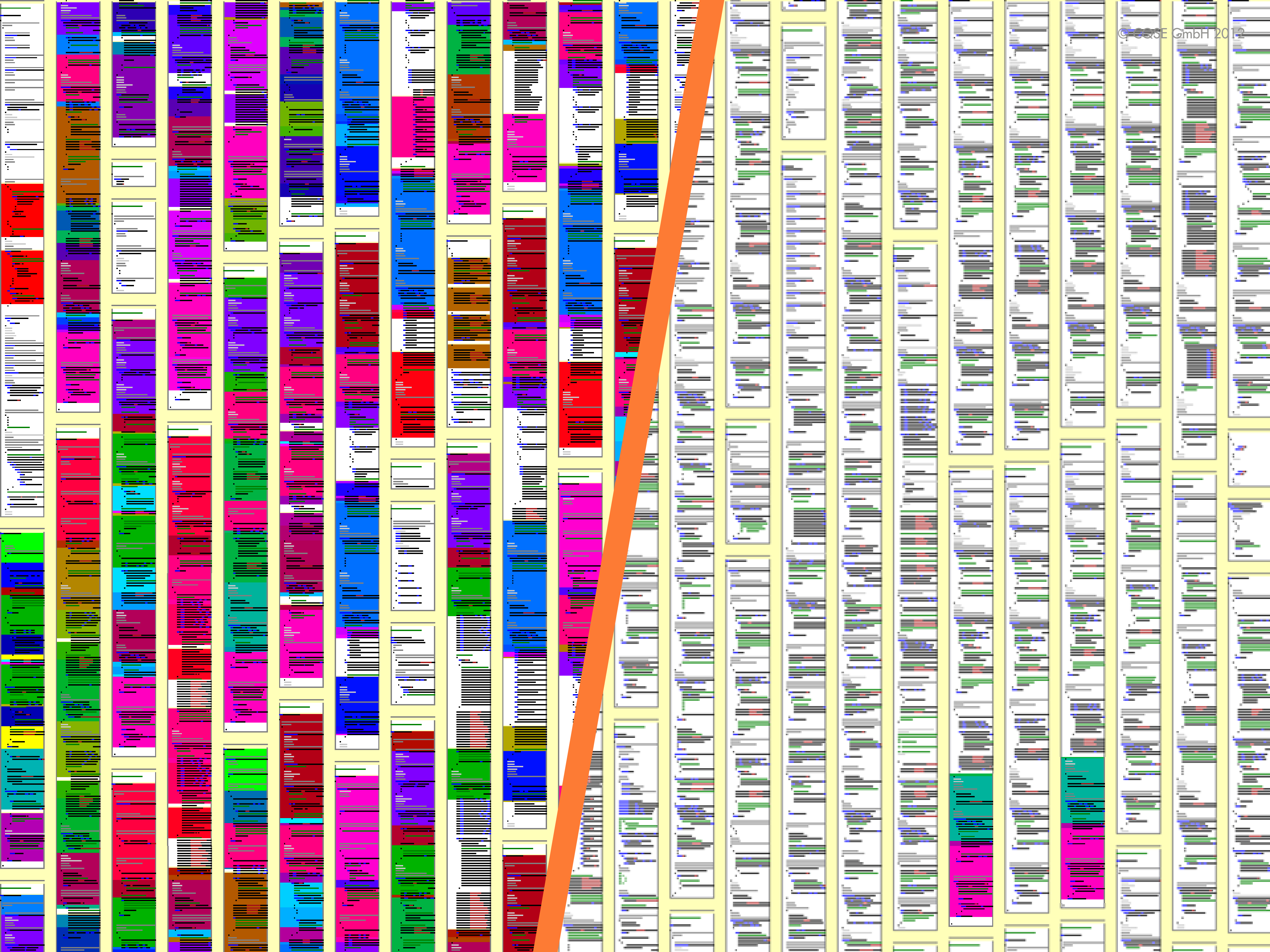
6 AV0001 : Invalid Dependency : PetShop.BLL.Order.Insert(Method) --> PayPalClient.PayPalService.MakePayment(Method)
Layers: Business, PaymentProviders.PayPal | Dependencies: Calls

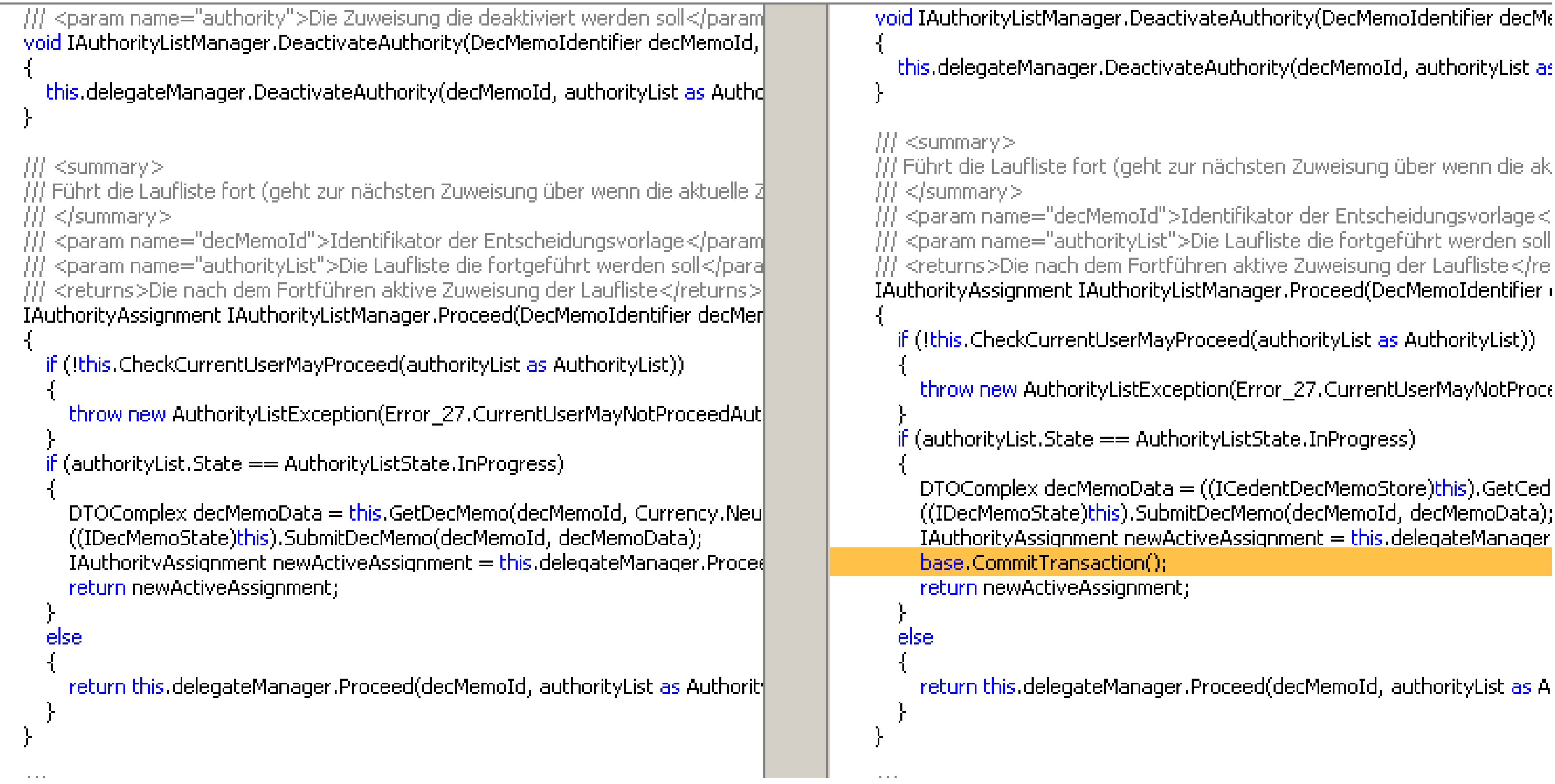
Anforderungen an zuverlässige KPIs

- Objektiv
- Auswirkungen von Code-Änderungen verständlich
- Actionable
- Nachvollziehbarer Zusammenhang zu Wartungstätigkeit

WARNING
BRIGHT LIGHT
– Do not stare into light beam.
– Do not view at close range.
Failure to do so will cause
permanent eye damage

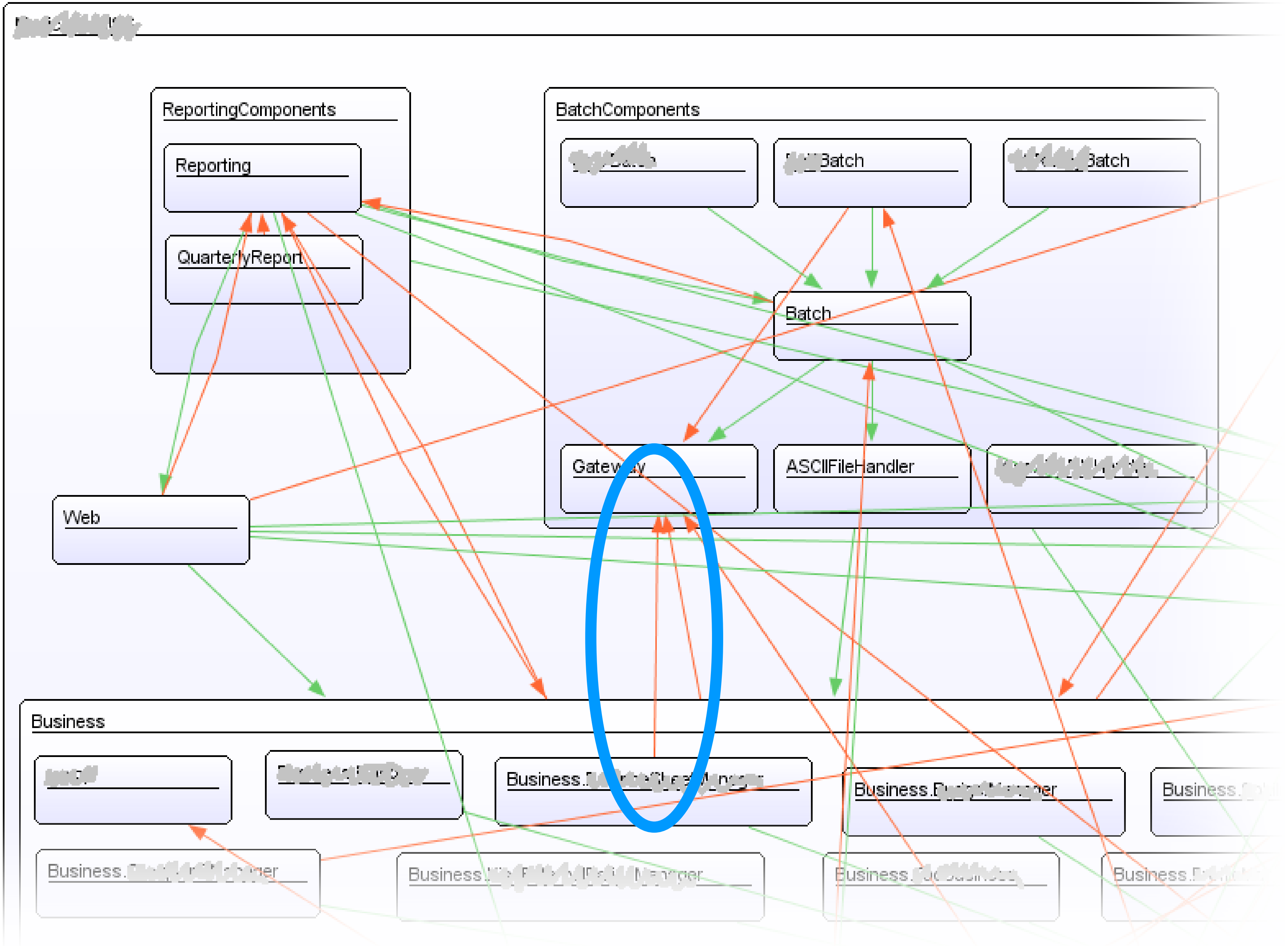


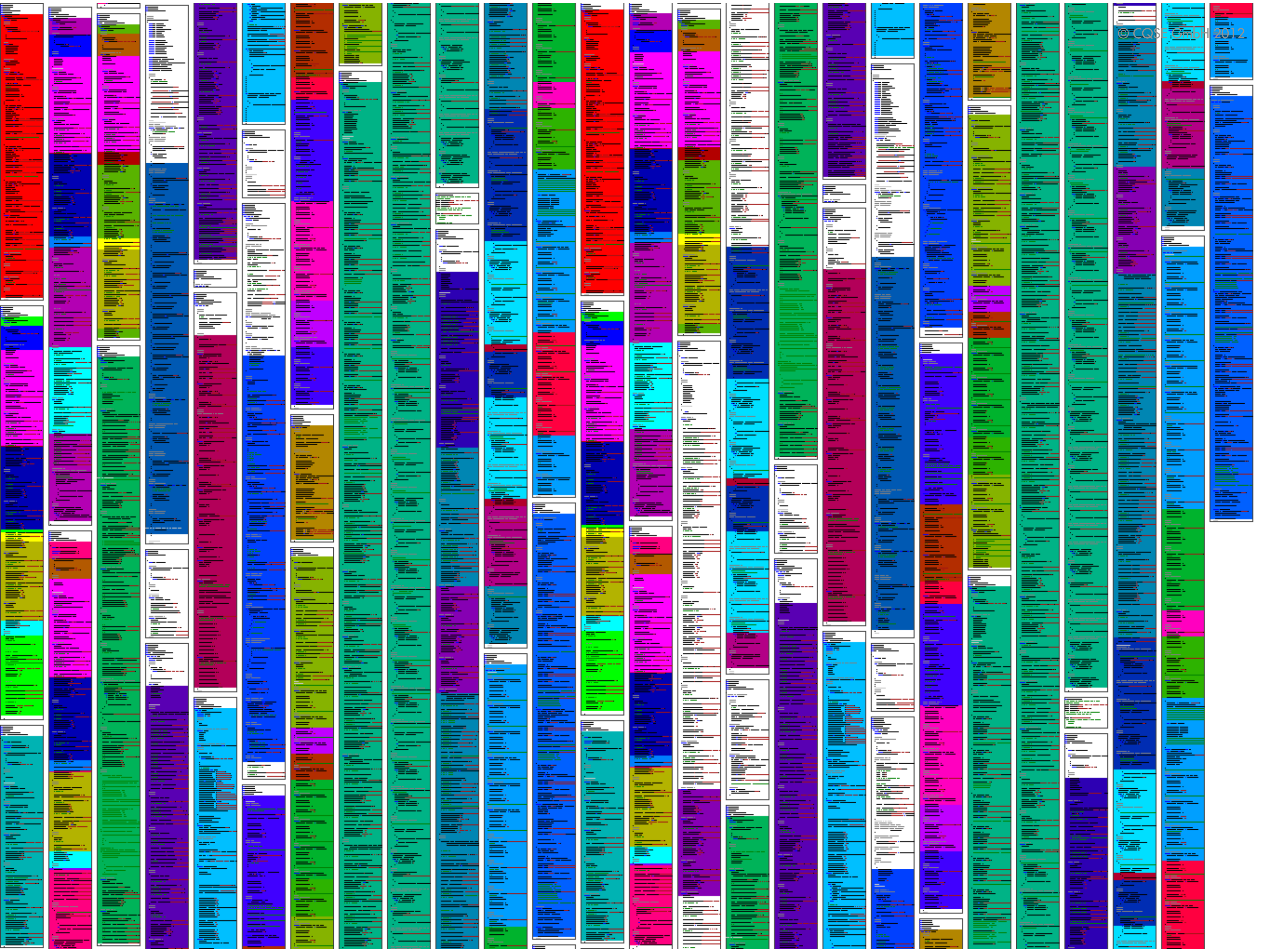


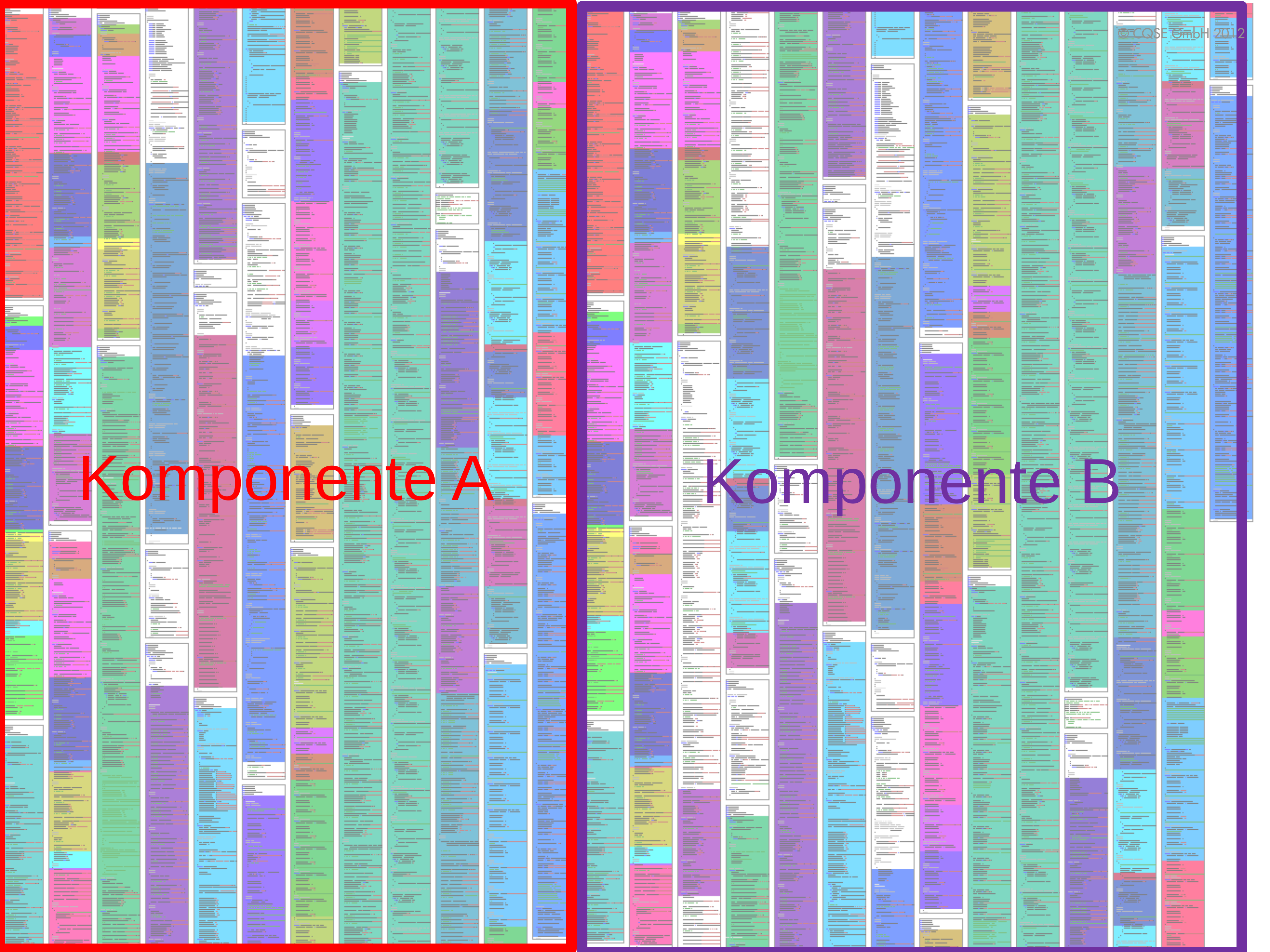


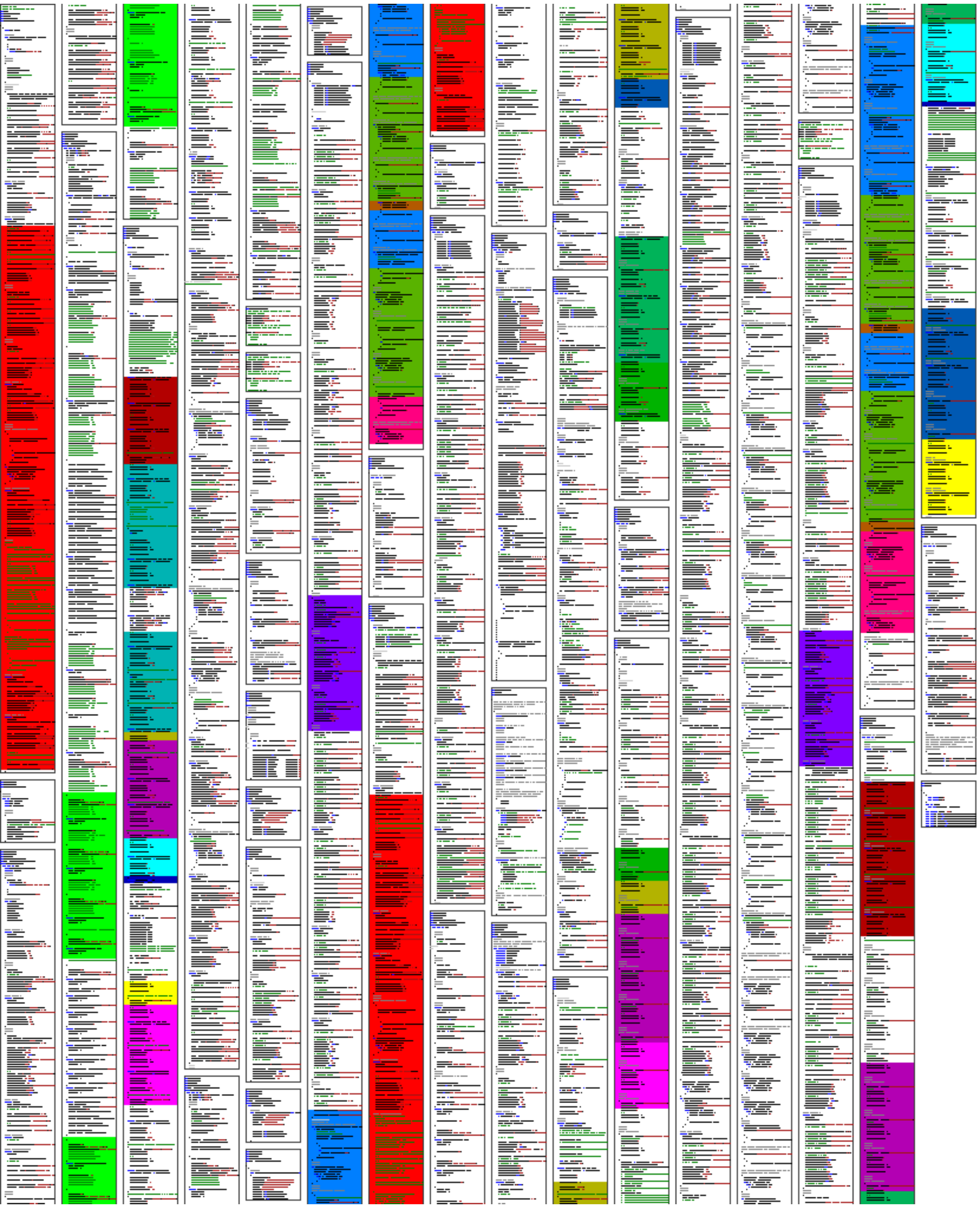
```
/// <param name="authority">Die Zuweisung die deaktiviert werden soll</param
void IAuthorityListManager.DeactivateAuthority(DecMemoIdentifier decMemoId,
{
   this.delegateManager.DeactivateAuthority(decMemoId, authorityList as Authc
}

/// <summary>
/// Führt die Laufliste fort (geht zur nächsten Zuweisung über wenn die aktuelle Z
/// </summary>
/// <param name="decMemoId">Identifikator der Entscheidungsvorlage</param
/// <param name="authorityList">Die Laufliste die fortgeführt werden soll</para
/// <returns>Die nach dem Fortführen aktive Zuweisung der Laufliste</returns>
IAuthorityAssignment IAuthorityListManager.Proceed(DecMemoIdentifier decMer
{
   if (!this.CheckCurrentUserMayProceed(authorityList as AuthorityList))
   {
      throw new AuthorityListException(Error_27.CurrentUserMayNotProceedAut
   }
   if (authorityList.State == AuthorityListState.InProgress)
   {
      DTOComplex decMemoData = this.GetDecMemo(decMemoId, Currency.Neu
      ((IDecMemoState)this).SubmitDecMemo(decMemoId, decMemoData);
      IAuthorityAssignment newActiveAssignment = this.delegateManager.Procee
      return newActiveAssignment;
   }
   else
   {
      return this.delegateManager.Proceed(decMemoId, authorityList as Authorit
   }
}
...
```

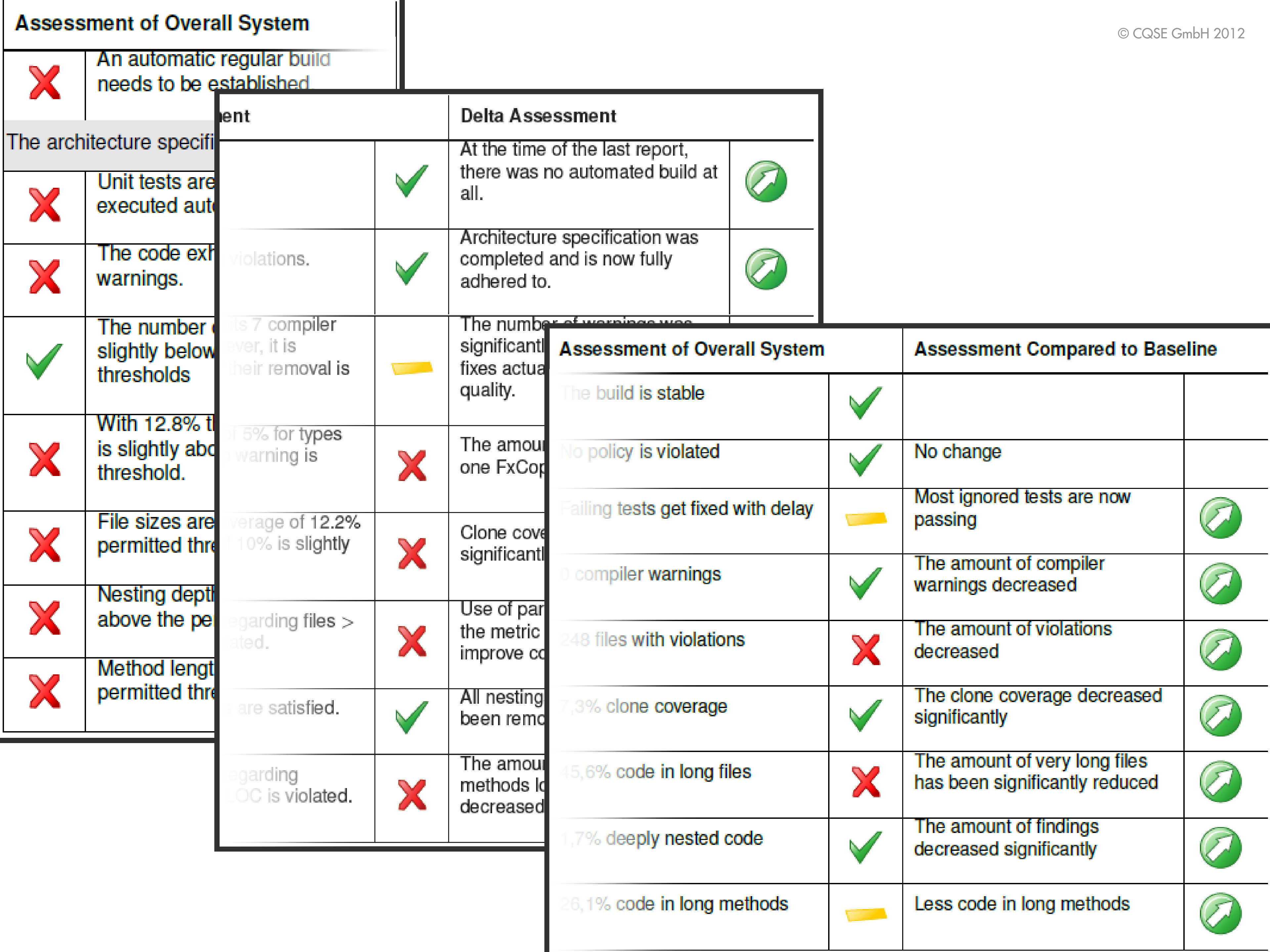
```
void IAuthorityListManager.DeactivateAuthority(DecMemoIdentifier decMe
{
   this.delegateManager.DeactivateAuthority(decMemoId, authorityList as
}

/// <summary>
/// Führt die Laufliste fort (geht zur nächsten Zuweisung über wenn die ak
/// </summary>
/// <param name="decMemoId">Identifikator der Entscheidungsvorlage<
/// <param name="authorityList">Die Laufliste die fortgeführt werden soll
/// <returns>Die nach dem Fortführen aktive Zuweisung der Laufliste</re
IAuthorityAssignment IAuthorityListManager.Proceed(DecMemoIdentifier
{
   if (!this.CheckCurrentUserMayProceed(authorityList as AuthorityList))
   {
      throw new AuthorityListException(Error_27.CurrentUserMayNotProce
   }
   if (authorityList.State == AuthorityListState.InProgress)
   {
      DTOComplex decMemoData = ((ICedentDecMemoStore)this).GetCed
      ((IDecMemoState)this).SubmitDecMemo(decMemoId, decMemoData);
      IAuthorityAssignment newActiveAssignment = this.delegateManager
      base.CommitTransaction();
      return newActiveAssignment;
   }
   else
   {
      return this.delegateManager.Proceed(decMemoId, authorityList as A
   }
}
...
```


[redacted]
ReportingComponents
Reporting
QuarterlyReport
BatchComponents
[redacted]
[redacted]Batch
[redacted]Batch
Batch
Gateway
ASCIIFileHandler
[redacted]
Web
Business
[redacted]
[redacted]
Business.[redacted]
Business.[redacted]
Business.[redacted]
Business.[redacted]
Business.[redacted]
Business.[redacted]
Business.[redacted]


© CQSE GmbH 2012


Komponente A
Komponente B

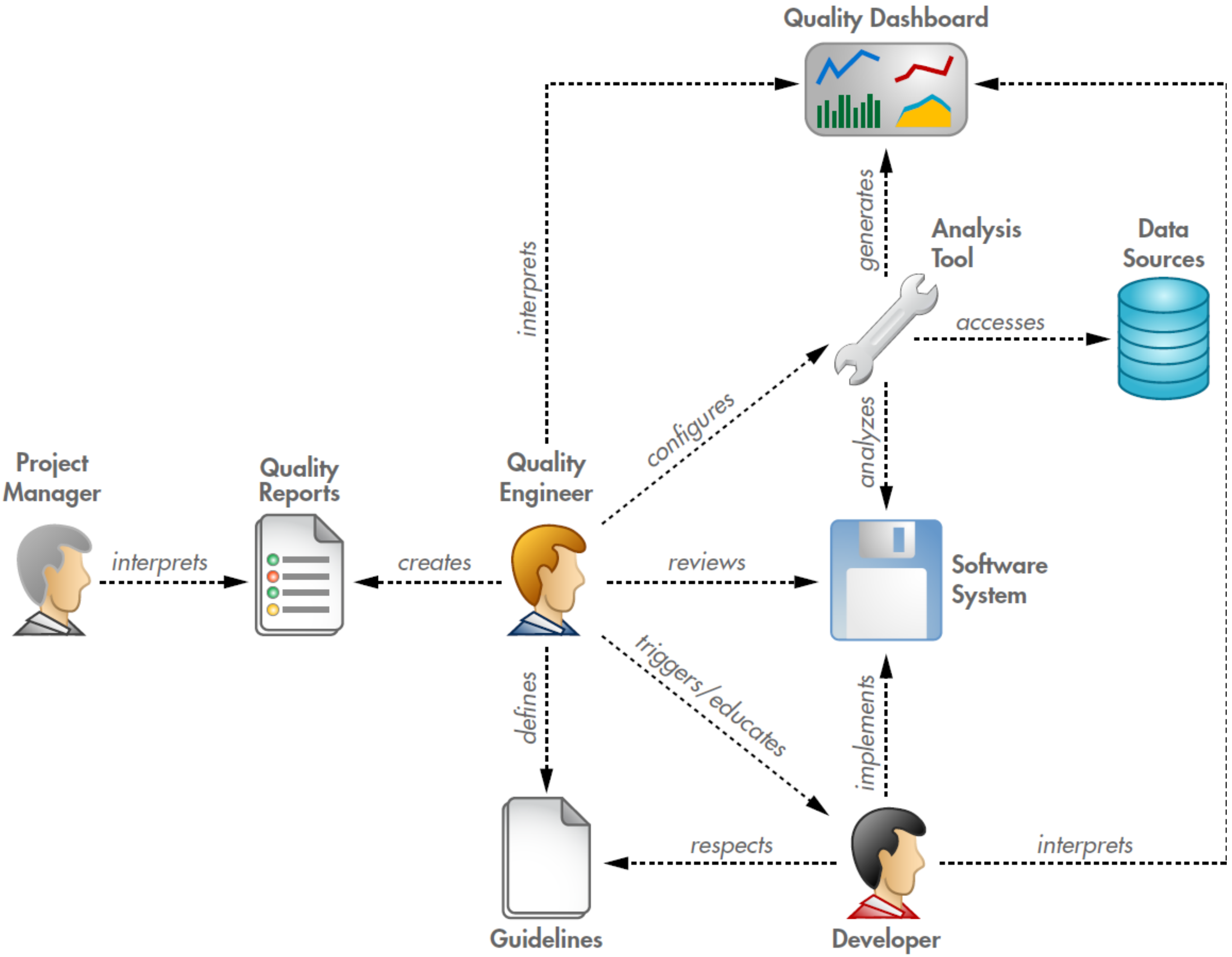




Assessment of Overall System	
✗	An automatic regular build needs to be established
The architecture specifi	
✗	Unit tests are executed aut
✗	The code exh warnings.
✓	The number slightly below thresholds
✗	With 12.8% t is slightly abo threshold.
✗	File sizes are permitted thre
✗	Nesting depth above the pe
✗	Method lengt permitted thre

ent		Delta Assessment	
	✓	At the time of the last report, there was no automated build at all.	↗
violations.	✓	Architecture specification was completed and is now fully adhered to.	↗
ls 7 compiler ver, it is their removal is	▬	The number of warnings was significant fixes actua quality.	
f 5% for types warning is	✗	The amou one FxCop	
erage of 12.2% f 10% is slightly	✗	Clone cove significant	
egarding files > ated.	✗	Use of par the metric improve co	
s are satisfied.	✓	All nesting been remo	
egarding LOC is violated.	✗	The amou methods l decreased	

Assessment of Overall System		Assessment Compared to Baseline	
The build is stable	✓		
No policy is violated	✓	No change	
Failing tests get fixed with delay	▬	Most ignored tests are now passing	↗
0 compiler warnings	✓	The amount of compiler warnings decreased	↗
248 files with violations	✗	The amount of violations decreased	↗
7,3% clone coverage	✓	The clone coverage decreased significantly	↗
45,6% code in long files	✗	The amount of very long files has been significantly reduced	↗
1,7% deeply nested code	✓	The amount of findings decreased significantly	↗
26,1% code in long methods	▬	Less code in long methods	↗


Quality Dashboard
interprets
generates
Analysis Tool
Data Sources
accesses
configures
analyzes
Project Manager
interprets
Quality Reports
creates
Quality Engineer
reviews
Software System
defines
triggers/educates
implements
respects
interprets
Guidelines
Developer

Fazit

Zuverlässige KPIs sind die Voraussetzung für aussagekräftiges, wirksames Software Quality Control.

Kontakt

Dr. Elmar Juergens · juergens@cqse.eu · +49 179 675 3863

Ich bin heute den ganzen Tag hier und freue mich auf Diskussionen.

CQSE GmbH
Lichtenbergstraße 8
85748 Garching bei München

Quellen

- E. Juergens, F. Deissenboeck, B. Hummel, S. Wagner: *„Do Code Clones Matter?"*, International Conference on Software Engineering, 2009

- M. Feilkas, D. Ratiu, E. Juergens: *„The Loss of Architectural Knowledge during System Evolution: An Industrial Case Study"*, International Conference on Program Understanding, 2009

- F. Deissenboeck: *"Continuous Quality Control of Long-Lived Software Systems"*, Doktorarbeit, Technischen Universtität München, 2009

- www.conqat.org

Quellen der Abbildungen

- Windows Screenshots: Wikipedia.
- Linux Growth Chart: Dominik Strzalka, „*Fractal Properties of Linux Kernel Maps*“, Computer Science & Engineering, 2012
- Visual Studio Layer Diagram: Microsoft Tutorial „*Beschreiben und Einsetzen der Abhängigkeiten*“
  http://www.microsoft.com/visualstudio/deu/products/visual-studio-ultimate-2012

Alle weiteren Abbildungen wurden selbst erstellt.