

QUALITY<sup>®</sup>  
made by AAREAL

# Open Source Risk Engine Praxisbericht

Burkhard Mayer und Dr. Frank Hölldorfer, Aareal Bank AG  
RISKMANAGEMENTforBANKS 2017, Köln, 7.11.17

**Aareal**

# Business segments

Aareal Bank Group

**QUALITY**<sup>®</sup>  
made by **AAREAL**

## Strukturierte Immobilienfinanzierungen

Internationale Aufstellung und Geschäftsaktivitäten auf drei Kontinenten – in Europa, den USA und Asien

Finanzierung von Gewerbeimmobilien, insbesondere Büros, Hotels, Shoppingcenter sowie Logistik- und Wohnimmobilien

Gesamtbestand: 27,9 Mrd. €

## Consulting / Dienstleistungen für die Immobilienwirtschaft

Marktführendes integriertes Zahlungsverkehrssystem für die deutsche Wohnungs- und gewerbliche Immobilienwirtschaft sowie die Energiewirtschaft

Marktführende IT-Systeme für das Management von Wohn- und Gewerbeimmobilien in Europa

Über 10 Millionen verwaltete Mieteinheiten in Europa

Internationale Präsenz: Frankreich, Niederlande, Norwegen, Großbritannien und Skandinavien

Stand: 31.12.2016

**Aareal**



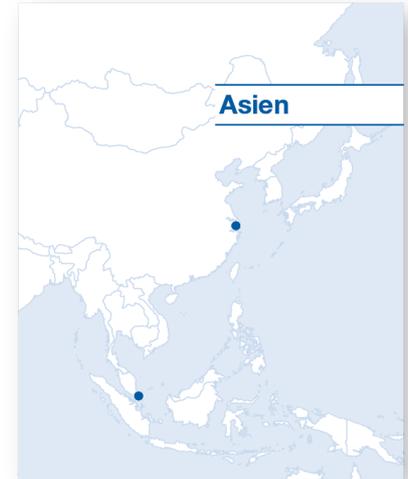
**Strukturierte  
Immobilienfinanzierungen**



**Strukturierte  
Immobilienfinanzierungen**



**Consulting/Dienstleistungen**



**Strukturierte  
Immobilienfinanzierungen**

**Aareal Bank, Strukturierte Immobilienfinanzierungen:** Brüssel, Dublin, Istanbul, London, Madrid, Moskau, New York, Paris, Rom, Schanghai, Singapur, Stockholm, Warschau, Wiesbaden |

**Aareal Estate AG:** Wiesbaden | **Aareal Valuation GmbH:** Wiesbaden |

**Westdeutsche ImmobilienBank AG:** Mainz, Münster

**Aareal Bank, Wohnungswirtschaft:** Berlin, Essen, Hamburg, Leipzig, München, Stuttgart, Wiesbaden | **Aareon AG:** Berlin, Bochum, Coventry, Dortmund, Emmen, Enschede, Gorinchem, Grathem, Hamburg, Hilversum, Hückelhoven, Karlskrona, Leipzig, Lund, Mainz, Meudon-la-Forêt, Mölndal, München, Nantes, Norrtälje, Orléans, Oslo, Piteå, Southampton, Stockholm, Stuttgart, Swansea, Toulouse | **Aareal First Financial Solutions AG:** Mainz | **Deutsche Bau- und Grundstücks-AG:** Berlin, Bonn, Dresden, Düsseldorf, Frankfurt am Main, Freiburg, Hannover, Leipzig, Moskau, München, Wuppertal

# Agenda

Einleitung: ORE, QuantLib und das Warum  
Evaluierung, Erkenntnisse und Folgen  
Aus dem Nähkästchen...

# Agenda

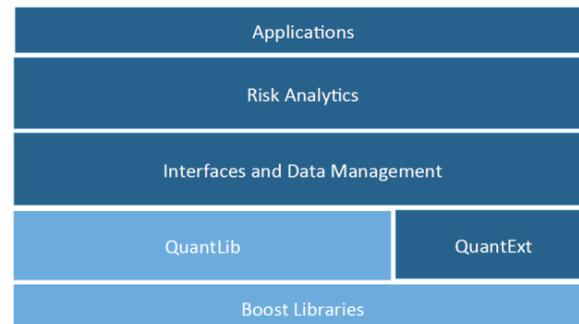
Einleitung: ORE, QuantLib und das Warum

Evaluierung, Erkenntnisse und Folgen

Aus dem Nähkästchen...

# ORE und QuantLib

## Open Source Risk Engine (ORE) ist eine auf QuantLib basierende Risikoanwendung.



- Genau wie QuantLib ist ORE eine **freie Open Source** Software (C++) mit unproblematischer “Modified BSD”-Lizenz.
- Homepage des Projektes: <http://www.opensourcerisk.org/>
- veröffentlicht unter Github: <https://github.com/OpenSourceRisk/Engine>
- In Gegensatz zur QuantLib ist die ORE eine **direkt laufende Anwendung** (Command Line, Dateigesteuert). Geschäfts- und Marktdaten sowie Konfiguration über simple Dateischnittstelle.
- Sponsor des Projektes ist Quaternion Risk Management
- Erstes Release der ORE im Oktober 2016. Kommunizierte Road-Map bzgl. Funktionsumfang.
- Aktuelles **zweites Release (Mai 2017)** auch mit kapitaltragenden Instrumenten und Zinssensitivitätsanalyse (wichtige Derivate bereits im ersten Release)
- Die von der ORE verwendete **QuantLib** ist eine wohlbekannte Bibliothek zur Bewertung von Finanzprodukten, erstes Release bereits 2000, aktive Weiterentwicklung, bekannt ist auch das Excel-Add-In quantlibxl.
- Homepage des Projektes: <http://quantlib.org>



**QuantLib**

A free/open-source library for quantitative finance

# Motivation für Nutzung der QuantLib via ORE



- Läuft sofort
- vollständig transparent und änderbar weil Quellcode offen
- hinreichend “leicht” für quantitativen Fachbereich
- praktisch keinerlei Lizenzproblematik
- “abgeschichtete Komplexität”
- externes Know-How einkaufbar
- Peer-Reviewed

Eignung als taktisches Tool ?  
(Ad-Hoc-Fragestellungen,  
Challenging)

Basis für produktive Analytics  
in einem Best-Of-Breed Ansatz?  
(Konsistenz, Preis, Flexibilität,  
Schnittstelle)



Aareal

FB will mehr als  
Excel und quantlibxl

schon gemachte Erfahrung  
des Fachbereichs mit  
QuantLib

Modernisierung und  
Modularisierung der IT

Aareal

# Agenda

Einleitung: ORE, QuantLib und das Warum

Evaluierung, Erkenntnisse und Folgen

Aus dem Nähkästchen...

# Das Experiment

## ...in Form eines kleinen Fachbereich-Projektes

Ein kleines kurzes Projekt (100 PT, 40k€) mit Beratung durch Quaternion durchgeführt um Eignung der ORE bzgl. der Basis-Analytics festzustellen.

### Kriterien:

- look-and-feel Anwendung
- look-and-feel Code und Qualität
- Produktabdeckung: Gesamtbestand der Bank
- Nachvollziehbarkeit Barwert, Sensitivität, Cashflow
- Weiterentwicklungsfähigkeit, Flexibilität, Unterstützung durch Quaternion und Community
- Performance

# gemachte Erfahrungen...

## Jung aber mächtig.

- ORE war überraschend schnell am laufen.
- Mitgelieferte Beispiele und Dokumentation erleichtern den Zugang.
- Zugang recht intuitiv und aufgeräumt: Beratungsbudget bei weitem nicht ausgeschöpft
- Stattdessen Sponsoring von fehlenden Features (z.B. Tilgungsstrukturen), wurde schnell umgesetzt.
- Fast vollständige Produktabdeckung.
- Bugs waren schnell gefunden und schnell beseitigt, kleinere „Wünsche“ schnell selbst umgesetzt.
- Markt- und Simulationsbeschreibung ist mächtig; führt zu entspr. Konfigurationsaufwand
- Schnittstellen für Geschäfte und Marktdaten sind auf ORE-Seite recht transparent; überraschend schwierig war der Download aus den Quellsystemen
- Die Community könnte größer sein – aber die ORE ist ja noch jung.

Die Folgen:

- ✓ Als taktisches Instrument im Risikocontrolling nicht mehr wegzudenken.
- ✓ IT-Projekt zur Ablösung von SEM-RA (SAP) durch ORE gestartet.
- ✓ Weiteres quantitatives Projekt auf ORE als Kernlösung umgeschwenkt.

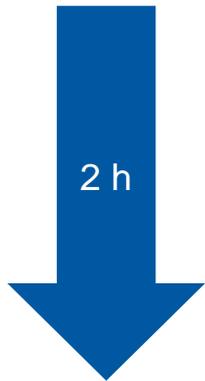
# Agenda

Einleitung: ORE, QuantLib und das Warum  
Evaluierung, Erkenntnisse und Folgen

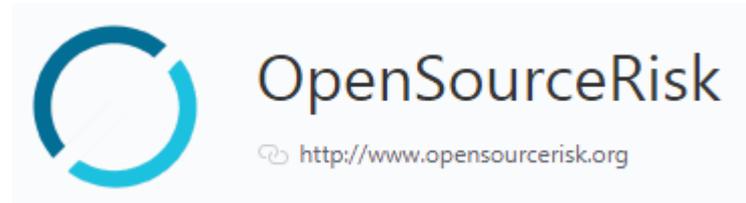
Aus dem Nähkästchen...

# Der Einstieg

## Wie bekommt man die ORE?



- **Downloaden**  
[www.opensourcerisk.org](http://www.opensourcerisk.org)
- **Kompilieren**  
Anleitung: UserGuide.pdf  
Benötigte Bibliotheken: boost, QuantLib
- **Fertig**



## Anwendung

### Konsolenanwendung

Konfiguration, Input: xml, txt  
Output: csv

### Kein Frontend

Excel/Calc Demosheets

# Zum warm werden

## Mitgelieferte Beispiele

Example\_1  
Example\_2  
Example\_3  
Example\_4  
Example\_5  
Example\_6  
Example\_7  
Example\_8  
Example\_9  
Example\_10  
Example\_11  
Example\_12  
Example\_13  
Example\_14  
Example\_15  
Example\_16  
Example\_17  
Example\_18  
Example\_19  
Example\_20  
Input

conventions.xml  
curveconfig.xml  
fixings\_20160205.txt  
market\_20160205.txt  
market\_20160205\_flat.txt  
market\_20160205\_flat\_2.txt  
pricingengine.xml  
todaysmarket.xml  
netting.xml  
ore.xml  
portfolio.xml  
sensitivity.xml  
simulation.xml  
stresstest.xml

## Output

crossgamma.csv  
curves.csv  
flows.csv  
log.txt  
npv.csv  
scenario.csv  
sensitivity.csv

# Erweiterung

- Opensource
- Gut strukturiert
- Veränderung im Code
- Veröffentlichen im Repository
- Entwickler Feedback

## Added: Flowtyp to Cashflowoutput #14

New issue

Merged QuaternionRisk merged 7 commits into `openSourceRisk:master` from `bmay:master` on 7 Jul

Conversation 7 Commits 7 Files changed 2

Changes from all commits Jump to... +19 -0

Unified Split

```
10 OREAnalytics/orea/app/reportwriter.cpp
... @@ -1,6 +1,8 @@
1 1
2 2 /*
3 3 Copyright (C) 2017 Quaternion Risk Management Ltd
4 4 + Copyright (C) 2017 Aareal Bank AG
5 5 +
6 6 All rights reserved.
7 7
8 8 This file is part of ORE, a free-software/open-source library
@@ @@ -93,6 +95,7 @@ void ReportWriter::writeCashFlow(ore::data::Report& report, boost::shared_ptr<Po
93 95 .addColumn("Type", string())
94 96 .addColumn("LegNo", Size())
95 97 .addColumn("PayDate", Date())
98 98 + .addColumn("FlowType", string())
96 99 .addColumn("Amount", double(), 4)
97 100 .addColumn("Currency", string())
98 101 .addColumn("Coupon", double(), 10)
@@ @@ -118,6 +121,7 @@ void ReportWriter::writeCashFlow(ore::data::Report& report, boost::shared_ptr<Po
118 121 Date payDate = ptrFlow->date();
119 122 if (payDate >= asof) {
120 123 Real amount = ptrFlow->amount();
124 124 + string flowType = "";
121 125 if (payer)
122 126 amount *= -1.0;
123 127 std::string ccy = trades[k]->legCurrencies()[i];
@@ @@ -128,9 +132,11 @@ void ReportWriter::writeCashFlow(ore::data::Report& report, boost::shared_ptr<Po
128 132 if (ptrCoupon) {
129 133 coupon = ptrCoupon->rate();
130 134 accrual = ptrCoupon->accrualPeriod();
135 135 + flowType = "Interest";
131 136 } else {
```

# Eigene Beispiele

- Aufgebaut anhand der existierenden Beispiele
- Abgleich von CF, BW und Sensi

Auffällig:

Bond mit Floor ohne Sensi auf FloorVols

```
9 OREData/ored/portfolio/bond.cpp View
... .. @@ -1,5 +1,7 @@
1 1 /*
2 2 Copyright (C) 2016 Quaternion Risk Management Ltd
3 3 + Copyright (C) 2017 Aareal Bank AG
4 4 +
3 5 All rights reserved.
4 6
5 7 This file is part of ORE, a free-software/open-source library
⚡ @@ -53,6 +55,8 @@ void Bond::build(const boost::shared_ptr<EngineFactory>& engineFactory) {
53 55 currency_ = coupons_.currency();
54 56 Leg leg;
55 57 Handle<IborIndex> hIndex;
58 58 + Handle<OptionletVolatilityStructure> ovs;
59 59 +
56 60 if (coupons_.legType() == "Fixed")
57 61 leg = makeFixedLeg(coupons_);
58 62 else if (coupons_.legType() == "Floating") {
⚡ @@ -62,6 +66,9 @@ void Bond::build(const boost::shared_ptr<EngineFactory>& engineFactory) {
62 66 QL_REQUIRE(!hIndex.empty(), "Could not find ibor index " << indexName << " in market.");
63 67 boost::shared_ptr<IborIndex> index = hIndex.currentLink();
64 68 leg = makeIborLeg(coupons_, index, engineFactory);
69 69 + if (!coupons_.floatingLegData().floors().empty() || !coupons_.floatingLegData().caps().empty())
70 70 + ovs = engineFactory->market()->capFloorVol(currency_, builder->configuration(MarketContext::pricing));
71 71 +
65 72 } else {
66 73 QL_FAIL("Unknown leg type " << coupons_.legType());
67 74 }
⚡ @@ -71,6 +78,8 @@ void Bond::build(const boost::shared_ptr<EngineFactory>& engineFactory) {
71 78 // workaround, QL doesn't register a bond with its leg's cashflows
72 79 if(!hIndex.empty())
73 80 bond->registerWith(hIndex);
81 81 + if (!ovs.empty())
82 82 + bond->registerWith(ovs);
74 83 }
75 84
76 85 Currency currency = parseCurrency(currency_);
```

# Massentest

- 23.532 Produkte
- CF, BW, Sensi
- Laufzeit 10 Minuten
- 90% davon für Sensiberechnung
- Arbeitsaufwand ca. 2,5 Tage

Schwierigkeit:

Datenextraktion aus dem Bestandssystem

```
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. Alle Rechte vorbehalten.

C:\Users\bmayer>cd \US-Projects\ORE\Examples\Example_OWN_04\

C:\US-Projects\ORE\Examples\Example_OWN_04>..\..\App\bin\x64\Release\ore.exe ore.xml
ORE starting
Conventions... OK
Market... OK
Market data loader... OK
Curve configuration... OK
OK
Engine factory... OK
Portfolio... OK
Write Reports... SKIP
Curve Report... OK
NPU Report... OK
Cashflow Report... OK
OK
Sensitivity Report... OK
Stress testing... SKIP
Simulation... SKIP
Aggregation and XUA Reports... SKIP
run time: 4.9e+02 sec
ORE done.

C:\US-Projects\ORE\Examples\Example_OWN_04>_
```

# Vielen Dank für Ihre Aufmerksamkeit!

Haben Sie Fragen oder Anmerkungen?



**Diese Präsentation enthält die Meinung des Vortragenden und gibt nicht zwingend auch die Sichtweise der Aareal Bank Group wieder. Darüber hinaus können Sachverhalte aus Verständniszwecken verkürzt oder vereinfacht dargestellt worden sein.**