DEUTSCHE BÖRSE GROUP

Architecture Review

Risk Product

April 26th, 2018

Plan for Today

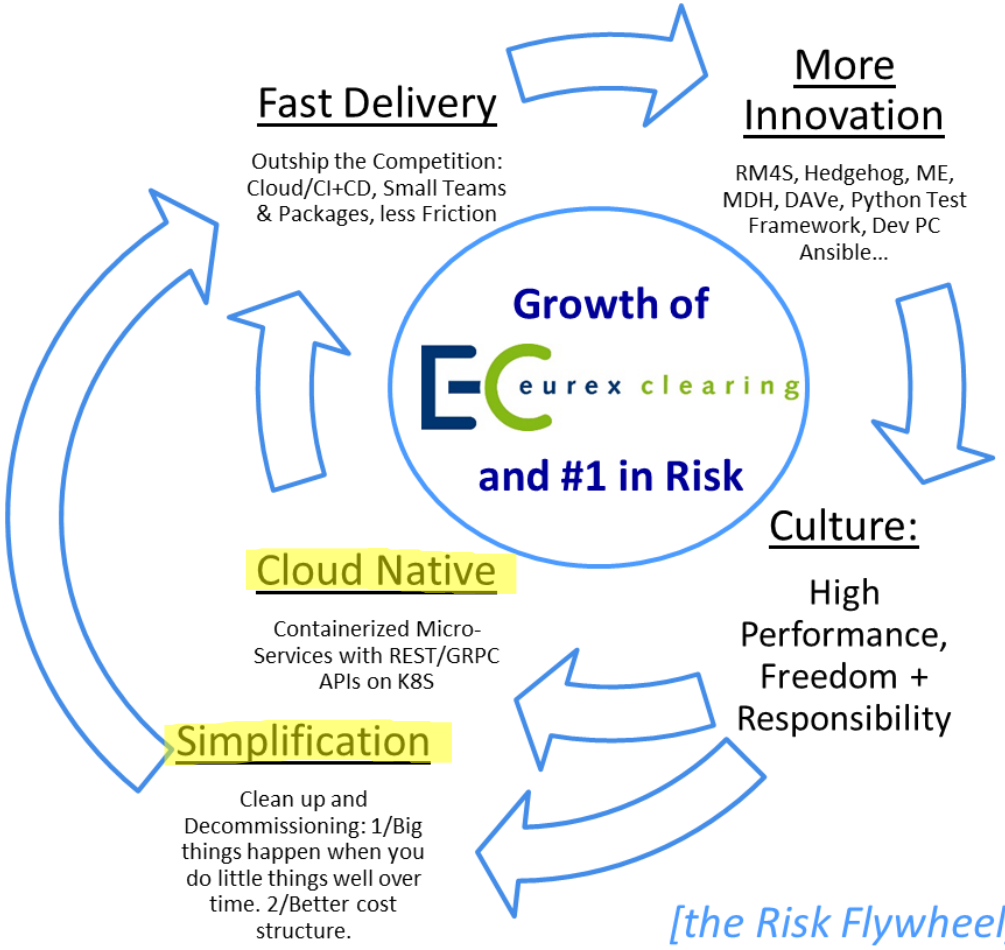
1)	Getting to ½ the Code and scaling Cloud Native	10’
2)	Risk vs Zombies 1: EDRE Decommissioning	5’
3)	Risk Management For Settlements	10’
4)	Demo 1: Market Data Hub (Michal)	35’
5)	Risk vs Zombies 2: MDH as a Zombie Killer	10’
6)	Update on Cloud Readiness and Next Steps (Martin)	10’
7)	Risk vs Zombies 3: Prisma GUI on Rest/JavaScript	10’
8)	Demo 2: Feature Baselines and AWS Builds in Prisma (Dusan)	25’
9)	Demo 3: Specification / Documentation Clean-Up (Martin)	10’
10)	Security Management Overview	10’
11)	Risk vs Zombies 4: RICA and RITA tools	5’
12)	Miscellaneous & Discussion	40’

Getting to ½ the Code and scaling Cloud Native

Our core values are ***high performance*** (standards), ***freedom*** and ***responsibility***.

We exist to make Eurex Clearing the ***industry leader in risk services*** (CCP risk management) and ***support the growth of Eurex*** into new products and markets.

And our goal over the next couple of years is to ***simplify our landscape*** and ***operate along cloud native services run by small and independent teams with end-to-end responsibility.***



Risk vs Zombies 1: EDRE Decommissioning

EDRE was switched off on 26.3.:

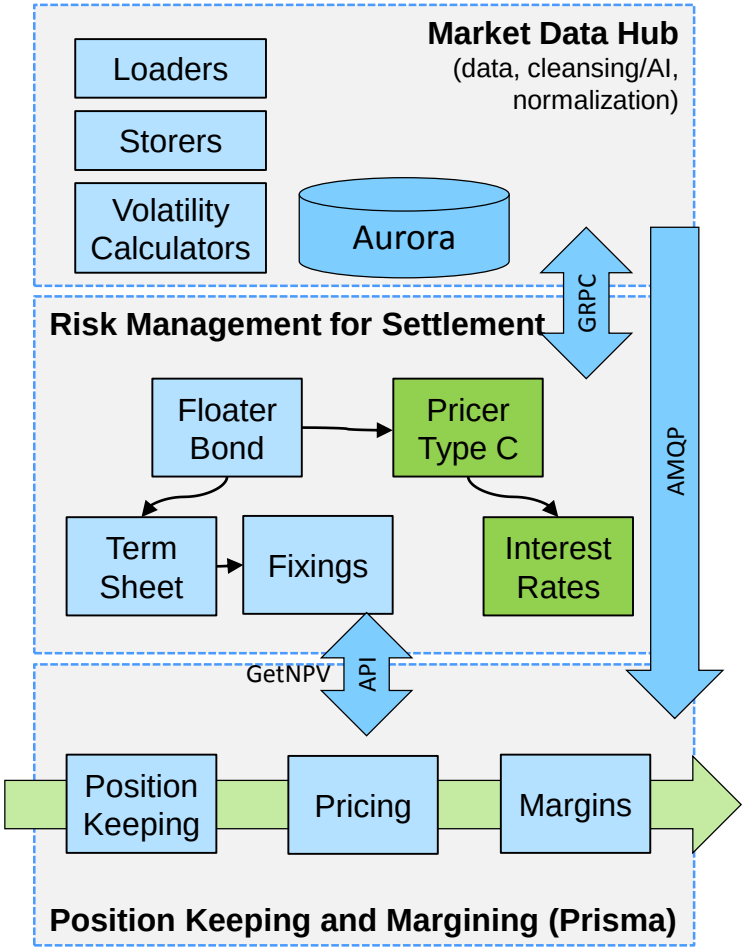
- 850k LOC / 12% of the code (fewer risks/vulnerabilities, easier to build new features)
- 6 on premise servers being decommissioned; DEV servers stay as shared resources for other projects.
- Switching off more difficult than anticipated - backwards compatibility, communication overhead, learning for users at the ECC.

Next Steps:

- Split functionalities along Products (settlement prices, interest on cash, collateral reports…)
- Replace Cash Market Pricing and Margining (more on that in a few minutes)
- Derivatives Clean-Up on RE / PND (let RE/PND run independently from Eurex, just processing Cash)
- Cash Clean-Up of RE / PND

Risk Management For Settlements

- **Problems:** RBM = outdated methodology and technology. Unavailable market data -> new product launches require workarounds.
- **MDH:** Eurex Ref. Data, SOL3 Market Supervision Data implemented, CEF, WSS in progress.
- **RM4S - Objectives:** Plug in different Pricers, diverse risk factors and their behavior, new asset classes pricing based on data that is available.
- **RM4S - Status:** Agile development (team of 4) of first use case (bullet bond) and interpretation with MDH. Focus on flexible instantiation of object trees based on configuration and data discovery.
- **Skills and Capacity: I**T to build deeper know how in Quantitative Finance; capacity to Run will increase.
- **Multi-Year Vision:** Structured Product Launches will become easier; methodology more defensible (reg. compliance), and consistent. Rollout first for Cash later also for Listed and OTC.



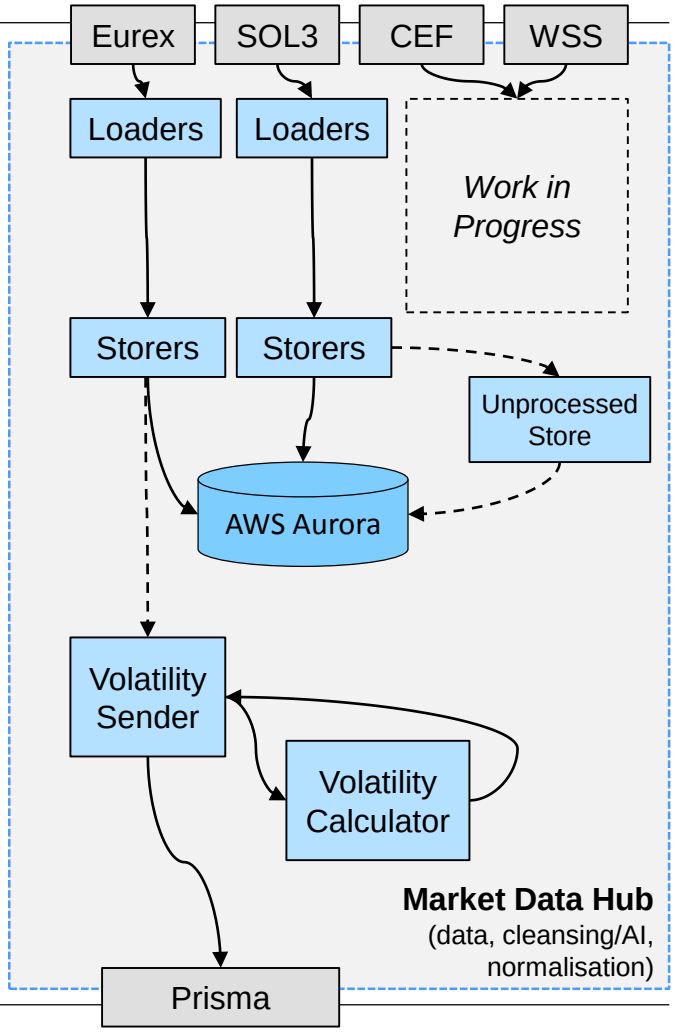
Demo 1: Market Data Hub (Michal)

Technology Stack:

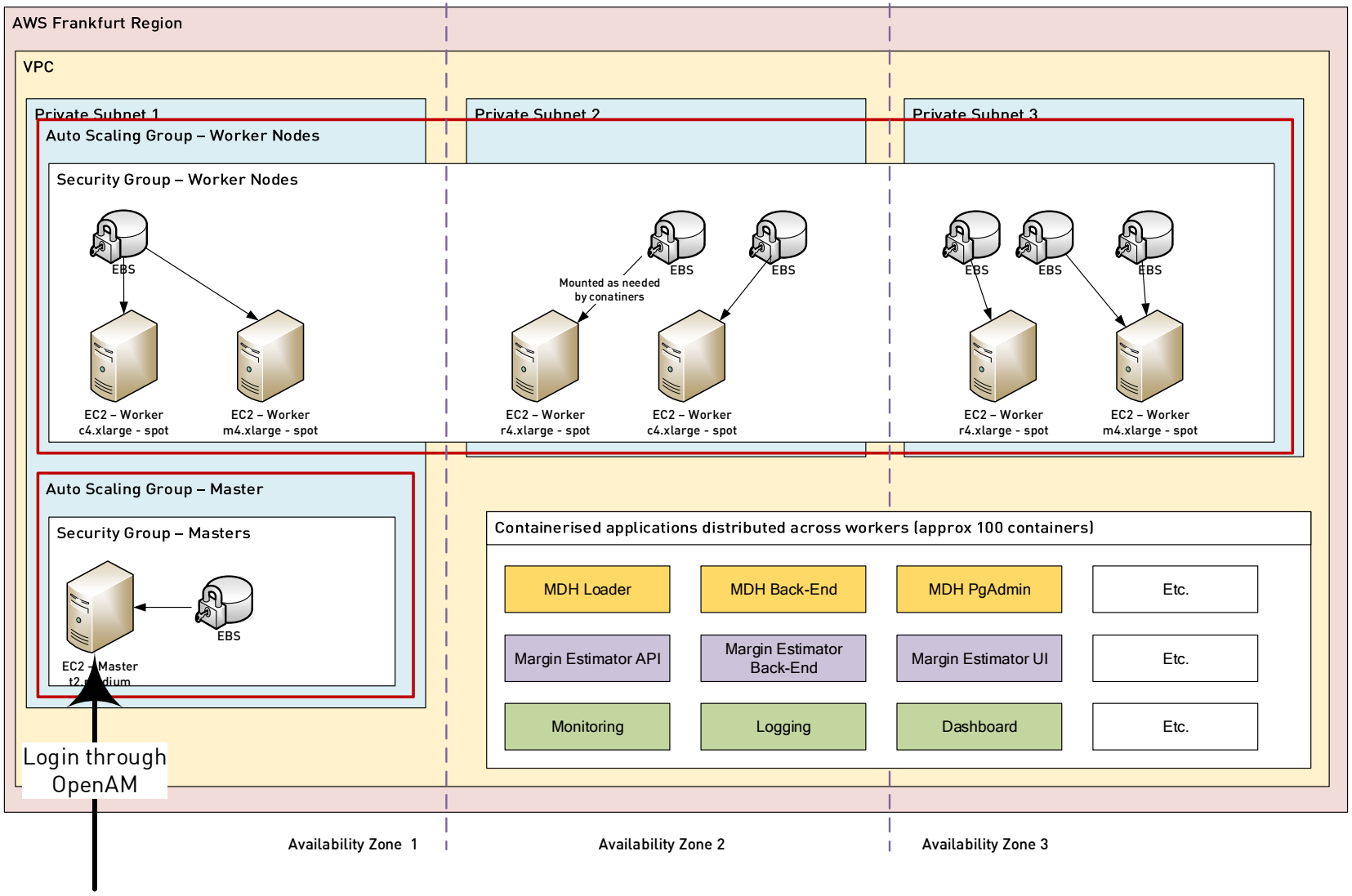
- Containerized Services on Kubernetes
- Reactive Programming / VertX Framework
- AWS Aurora / PostgreSQL Database -> managed services are extremely easy, fast and so far cheap (compared to doing it ourselves)

DEMO:

- Build cloud infrastructure from scratch (15-20min)
- *Security compliance* of our Kubernetes (OpenAM integration for Privileged Access Management)
- Same CI/CD pipeline; *good hygiene*: integration test coverage, code quality (SonarQube)
- Logging via *Elasticsearch* / *Kibana* and monitoring in *Grafana*
- A view of the data / pgAdmin.



Demo 1: Market Data Hub (Michal); Cloud Infrastructure



Risk vs Zombies 2: MDH as a Zombie Killer

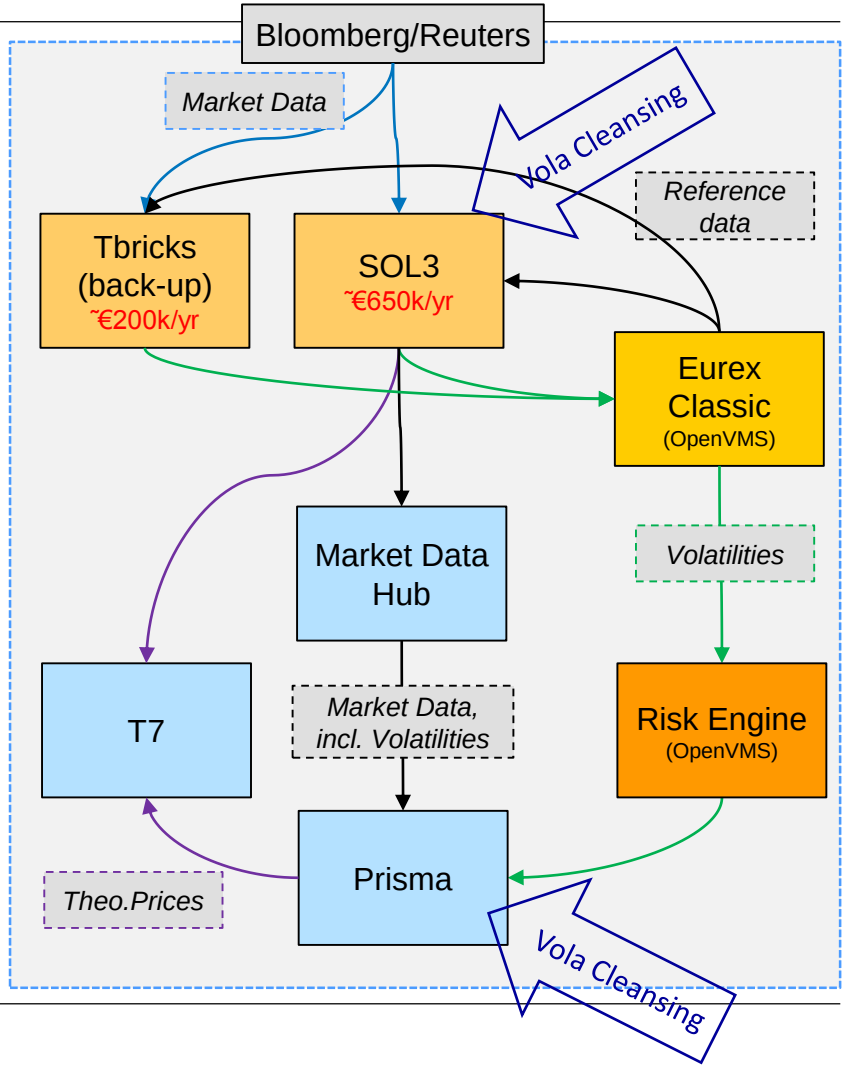
MDH is an enabler for RM4S: collection, cleansing and normalisation of market data that pricing / margining can build upon.

MDH helps us decommission Eurex and Risk Engine: routing data from Market Supervision systems to Risk systems.

MDH UI will decommission TBricks (ORC):

- Volatility maintenance
- Settlement Price Confirmation (and distribution which cannot stay on Eurex Classic forever)
- Reuters/Xetra underlying price feeds
- Theoretical price calculation

Ultimately, MDH should replace SOL3 as well (where we pay for changes, pay for licenses, and are dependent on 1 person in run/ops).



Update on Cloud Readiness and Next Steps (Martin)

	ME	MDH
Go-live	Q2	Q2/3
Acceptance Test	✓	in progress
Info Classification	✓	✓
IS Risk Assessment	in progress	
Data Protection	✓	
Outsourcing	✓	✓
Purchasing	✓	✓
BCM		
Internal Audit	✓	✓
Group Risk	✓	
Legal		
NPC	✓20.4.	

Margin Estimator:

- Security Concept under review (IS Risk Assessment); risks (if any remain) will be reviewed, understood and accepted by T. Laux and M. Matusza.
- 24.4. BaFin informed
- 3.5. ECAG board approval
- 16.5. for information Bundesbank + BaFin / DBG Cloud Group
- ✓ List API on DBP (staging environment)
- Launch in Production (contingent on 3.5. decision)

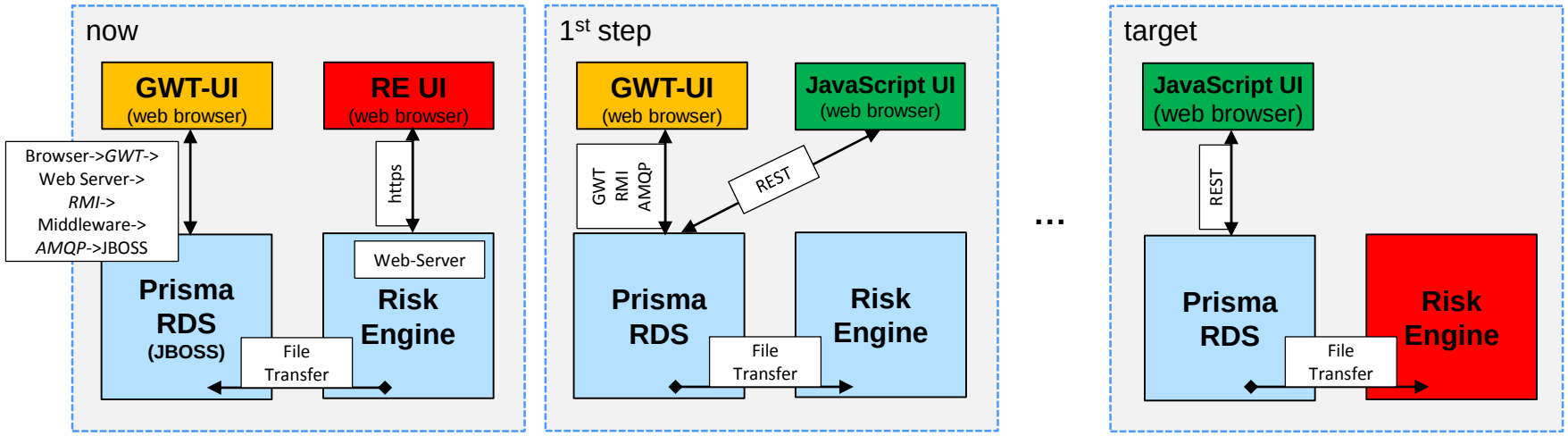
Market Data Hub:

- Risk Assessments and approvals
- Go-live with/after Prisma 7.1 (June 18th)

Afterwards:

- Learn from go-live, OPS upskilling
- Improve deployment / approval pipeline
- Launch **DAVe** and **Hedgehog** (confidential data)
- **On-premise Docker (RHEL 7.4)** / Microsoft Azure

Risk vs Zombies 3: Prisma GUI on Rest/JavaScript



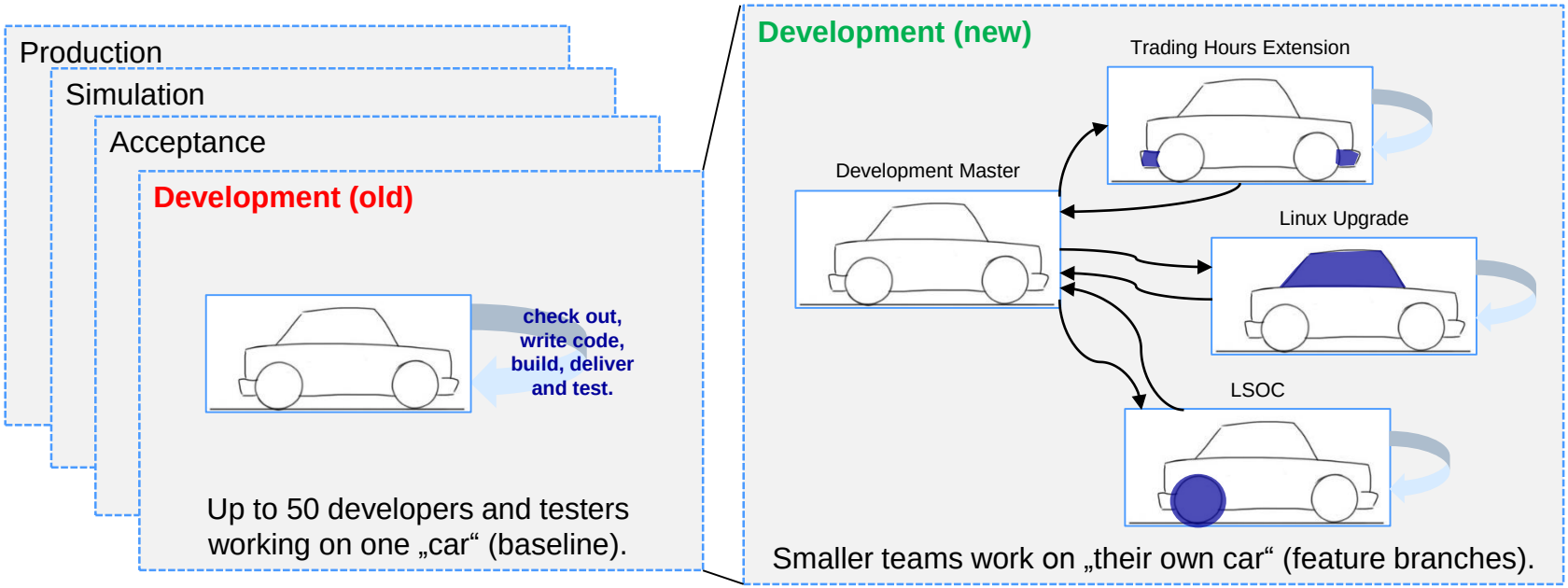
Approach:

- Expose REST in RDS
- Decommission Risk Engine Screens with a JavaScript UI on REST
- Step by step move what is needed from the Google Web Toolkit UI to the JavaScript UI and onto REST

Benefits:

- One step closer to switch off Risk Engine.
- Simplify RDS; harmonize across RISK (REST/JavaScript); standard communication & fewer security issues
- Re-energize the team (working with new technology sparked RICA)

Feature Baselines and AWS Builds for Prisma (Dusan)



- The “car” (baseline) gets broken for everyone as it is improved.
- Features running late cannot be easily put into the next release.
- Cont. integration on premise (Jenkins)

- More stable master; more flexibility in deliveries; less risk in release cycles.
- Pre-requisites: ***Automation of tools for cloning/merging*** and ***cloud capacity*** for continuous integration, builds and tests (*>>DEMO*).

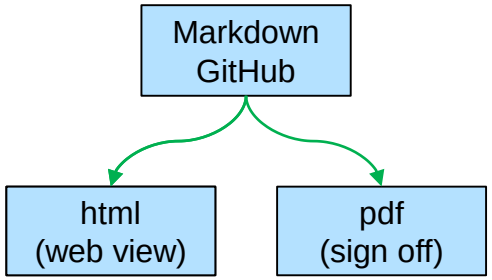
Demo 3: Specification / Documentation Clean-Up (Martin)

Problem to solve: Prisma System Documentation contains information which is not useful to anyone. Updates are cumbersome, it needs to be structured better, shortened and cleaned up.

Idea: Use new technology, and trigger a change of process / mind-set with this exercise.

First Steps: Combined Markdown + LaTeX for RICA and Hedgehog specifications. Difference Claims Tool Development.

GitHub native approach (branches, pull requests, reviews)



Learnings (demo):

- Branches = features, same as code approach
- Sharing repo with code is preferred = network effects
- New GitHubEE greatly improved diffs for markdown

Still to figure out:

- Versions / releases?
  -> Kubernetes approach is branches
- Scaling and on boarding
- Sign-off handling
- Beautification

Security Management Overview

Established:

- **Security Concepts** up to date -> process runs well; assessments and sign-offs mostly completed. (*Prisma, Risk Engine + PND/DWH, RICA, RITA (new), SOL3 + ORC (new), Kubernetes (new) und Margin Estimator (new)*).
- **Management of the Risk Register** - actively addressing old and new findings (˜27), communication and resolution tracking.
- **CyberArk used by Ops** team.
- **HP Fortify Scanning** - Scans established where possible (no Go, Javascript and cobol, etc), issues resolution in progress.

Still need to work on:

- **Patching process** – Strategy and process in draft (joint server ops with Derivatives). The vulnerabilities to be patched should come from Vulnerability Scanning and CISO defined baseline (minimum requirements per OS), or alternatively as Emergency Patch requests.
- **Vulnerability scanning** – Automatic ticket submission a ticket per finding and server, sorting, batch processing, prioritisation not really possible. Working with IS to receive more actionable scan results.
- **Penetration Testing** – Mandatory for all Critical Systems not only external facing ones. First Pen. Test will be Prisma after 7.1 launch (June 18th) on RHEL 7.4.

Risk vs Zombies 4: deprecated RICA tools; Matlab and Oracle

RICA

- Stopping 3 entire tools: Decay Factor, Bid Ask Spread / BAMC, Liquidity Add-On.
- No more testing or operating them, GitHub Repos archived, and generally less code (and Matlab) to run.
- Next victims: Vola Floors (R7.1), Robust Factor,…
- Auction Tool: possibly to be insourced after a recent fire-drill.

RITA

- Re-write of Matlab into Python and migration from Oracle to PostgreSQL starting (in Q1 we were helping out in RRH).

Business/Prototyping

- new work moved to C++/Python (QuantLib)

In August we will extend Matlab for 10 licenses to keep what we have (and need) running (this is down from 30).

Miscellaneous & Discussion

1. Python Testing framework finished for ETD, used in system/acceptance test – better maintainability and better performance → GitHub Repo

2. SRE/DEVOPS – progress is slow; SRE Training @ Google to be scheduled. We will co-run first cloud apps; and we will keep upskilling and practise, practise, practise. Points to work on:

- Better transparency → dashboards and monitoring open to all
- Process / habits → post-mortems; avoid problems from happening again and improve
- Get Coders into Ops → <50% of time spent on ops topics; Dev+Ops is one staffing pool (left over Ops work falls into Dev).
- Error Budgets / SLAs → let teams launch releases as quickly as they want as long as they keep to their SLA
- Collaboration with Server Ops: docker on premise / web-server security baseline

3. Kubernetes in 60 Minutes (video link) – popular on PTI; recorded by Innovation Lab.

+ Collected points / questions during this meeting.


DEUTSCHE BÖRSE
GROUP

Notes from the last Risk Architecture Review – 19.12.2017

Data Lake –> we would like to store historical data more flexibly and cheaply with the use of cloud. A data lake in the cloud (S3 Buckets, SQL or NoSQL) will allow us to use cloud native APIs for analytics, automation, AI, and Machine Learning. This ecosystem is growing rapidly, and therefore is preferable to a Hadoop/Cloudera type solution. Hadoop/Cloudera offer out of the box analytics and portability between cloud providers – but this is not the latest/most powerful technology. Content Lab/Data are looking at the same, Juergen/StatistiX also have a concept for what they want to do – and would like to introduce this to us soon. General direction: Should we have one "data lake" with a centralized team in the future – or do we shift responsibility + costs for historical data, and the ability to offer APIs and Services on top of them to individual products? → unchanged

React + Go -> make sure we are pushing hard enough. Re-writing code and upskilling people are important, can we train broader – important to have readily available work in React/Go for a higher % of the team. → JavaScript upskilling for front end and RDS developers, new RDS UI

We tend to set **arbitrary timelines for software and product launches** – the market does not need the features at that time, and to meet the timelines we make shortcuts and don't take enough time to re-engineer the Applications. Push back more, in order to do the right thing. → with hindsight we should have probably tested market fit/necessity of LSOC more

RE/EDRE/DWH should be on GitHub – at least as an archive of working software, where searches and viewing of the code will be possible even after the systems are decommissioned. (Done – code is being synced once per week, EDRE is archived: https://github.deutsche-boerse.de/dev/dbg.openvms.edre, https://github.deutsche-boerse.de/dev/dbg.openvms.riskengine, https://github.deutsche-boerse.de/dev/dbg.openvms.datawarehouse)

Interesting for next time: How do Developers work? (Process – JIRA, GitHub Issues…)

Enterprise GitHub in AWS – follow up for Oliver/Frank/Thomas Curran.

Launching Risk Cloud Apps (DAVe, Hedgehog, Market Data Hub, Margin Estimator) as non-material informing BaFin on 26.1. We were not fully in agreement on the Apps using member positions. Next Step: A meeting is set up for 4.1. to align with Data Protection, Group Risk, ECAG, Security, Cloud Team and Risk IT. → update presented today

API Mandate – difficult to track and measure if it is made mandatory by order from the top. Lars and Product Leads to look for ways of adopting it from within the Product Leads circle. → unchanged