DICE: Developing Data-Intensive Cloud Applications with Iterative Quality Enhancements

Giuliano Casale
Imperial College London
Project Coordinator

DICE Horizon 2020 Project
Grant Agreement no. 644869
http://www.dice-h2020.eu

OPEN CLOUD FORUM By OMG
LONDON 2015

Funded by the Horizon 2020
Framework Programme of the European Union

DICE Project

- Horizon 2020 Research & Innovation Action
  - Quality-Aware Development for Big Data applications
  - Feb 2015 - Jan 2018, 4M Euros budget
  - 9 partners (Academia & SMEs), 7 EU countries

Imperial College London

XLAB NOT IDLE

IeAT

Universidad Zaragoza

NETFECTIVE Creative Technology

ATC ATHENS TECHNOLOGY CENTER

POLITECNICO DI MILANO

prodevelop Integración de tecnologías

flexiant™ your cloud simplified

Motivation

- Software market rapidly shifting to Big Data
  - 32% compound annual growth rate in EU through 2016
  - 35% Big data projects are successful [CapGemini 2015]
- European call for software quality assurance (QA)
  - ISTAG: call to define environments "*for understanding the consequences of different implementation alternatives (e.g. quality, robustness, performance, maintenance, evolvability, ...)*"
- QA evolving too slowly compared to the trends in software development (Big data, Cloud, DevOps ...)
  - Still crucial for competiveness!

Quality-Aware MDE Today

Platform-Indep. Model

Architecture Model

Platform-Specific Model

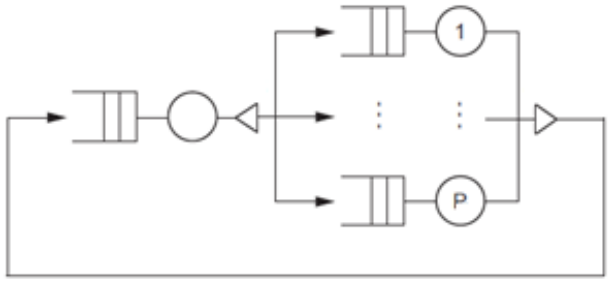
Domain Models

QA Models

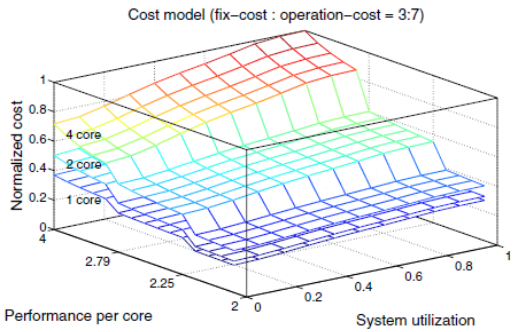
Platform Description

Code generation

Analytical Models



Cost-Quality Models



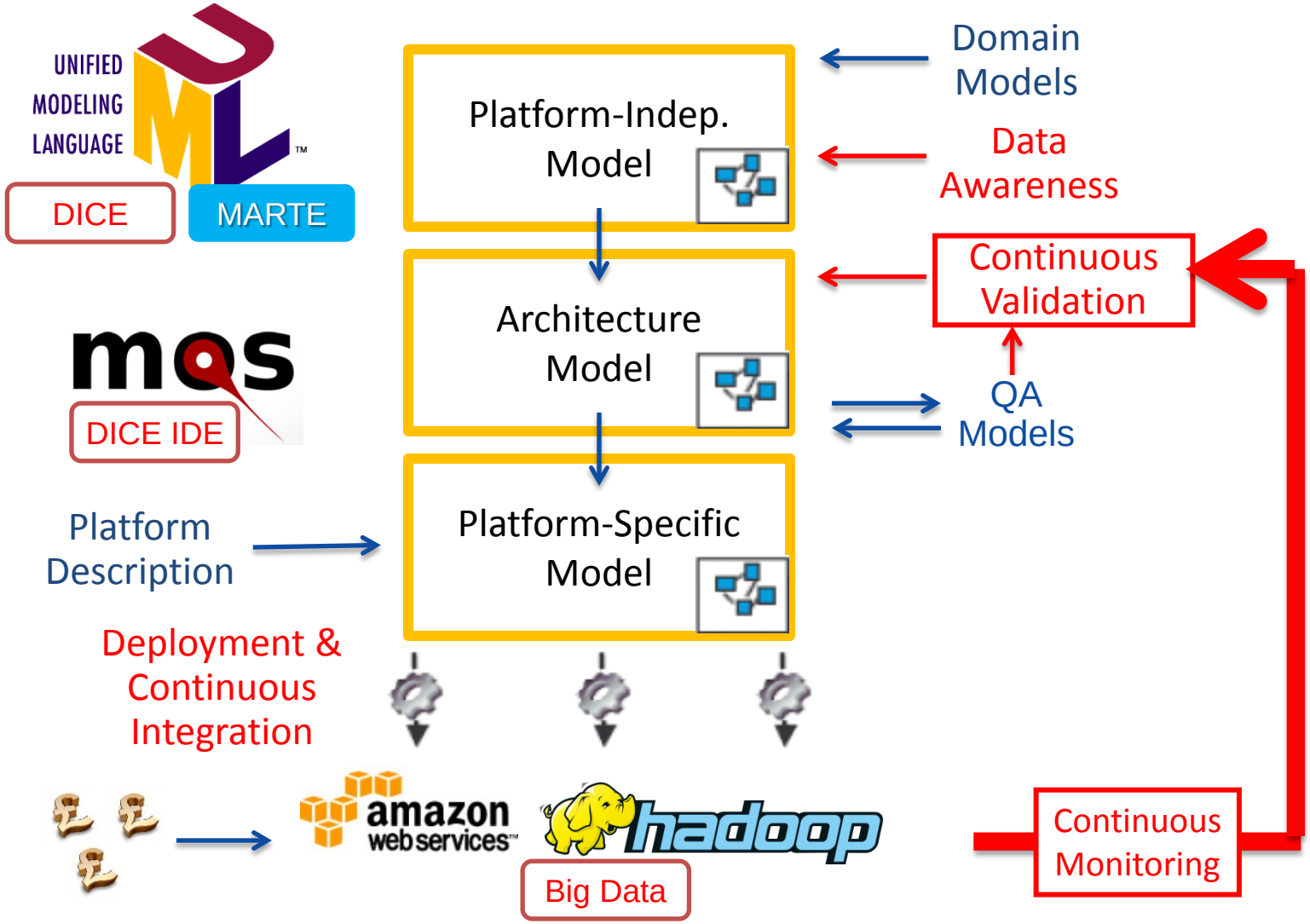
Challenge 1: QA for Big Data

- 5Vs:
  - Volume,
  - Velocity,
  - Variety,
  - Veracity,
  - Value
- Problem: today no QA toolchain can reason on the quality of complex Big Data applications
- Heteregeous Big Data Technologies
  - NoSQL, Spark, Hadoop/MapReduce, Storm, CEP, ...
- Cloud infrastructure adds complexity
  - Cloud storage, auto-scaling, private/public/hybrid, ...

Challenge 2: Embracing DevOps

- QA must become lean as well
  - Continuous quality checks and model versioning
- Modelling of the operations
  - Dev needs awareness of infrastructure and costs
- Continuous feedback
  - Forward and backward model synchronisation
  - Tracking of self-adaptation events (e.g. auto-scaling)
- Big data coming from continuous monitoring
  - QA has its own Big data, use machine learning?

A Holistic Approach: DICE



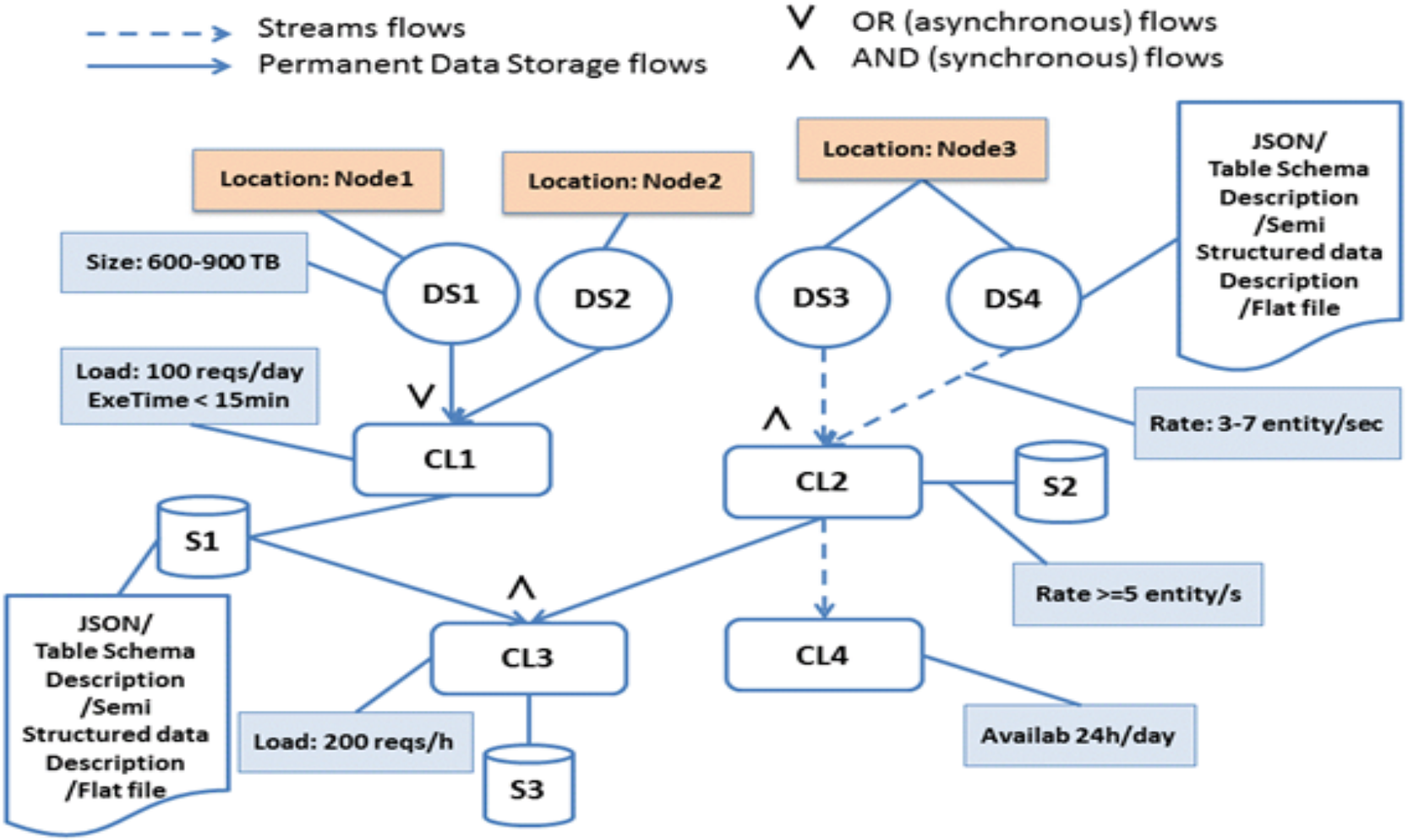
Benefits

- Tackling skill shortage and steep learning curves
  - Data-aware methods, models, and OSS tools
- Shorter time to market for Big Data applications
  - Cost reduction, without sacrificing product quality
- Decrease development and testing costs
  - Select optimal architectures that can meet SLAs
- Reduce number and severity of quality incidents
  - Iterative refinement of application design

DICE QA: Quality Dimensions

- Reliability
  - Availability
  - Fault-tolerance
- Efficiency
  - Performance
  - Time behaviour
  - Costs
- Safety & Privacy
  - Risk of harm
  - Privacy & data protection

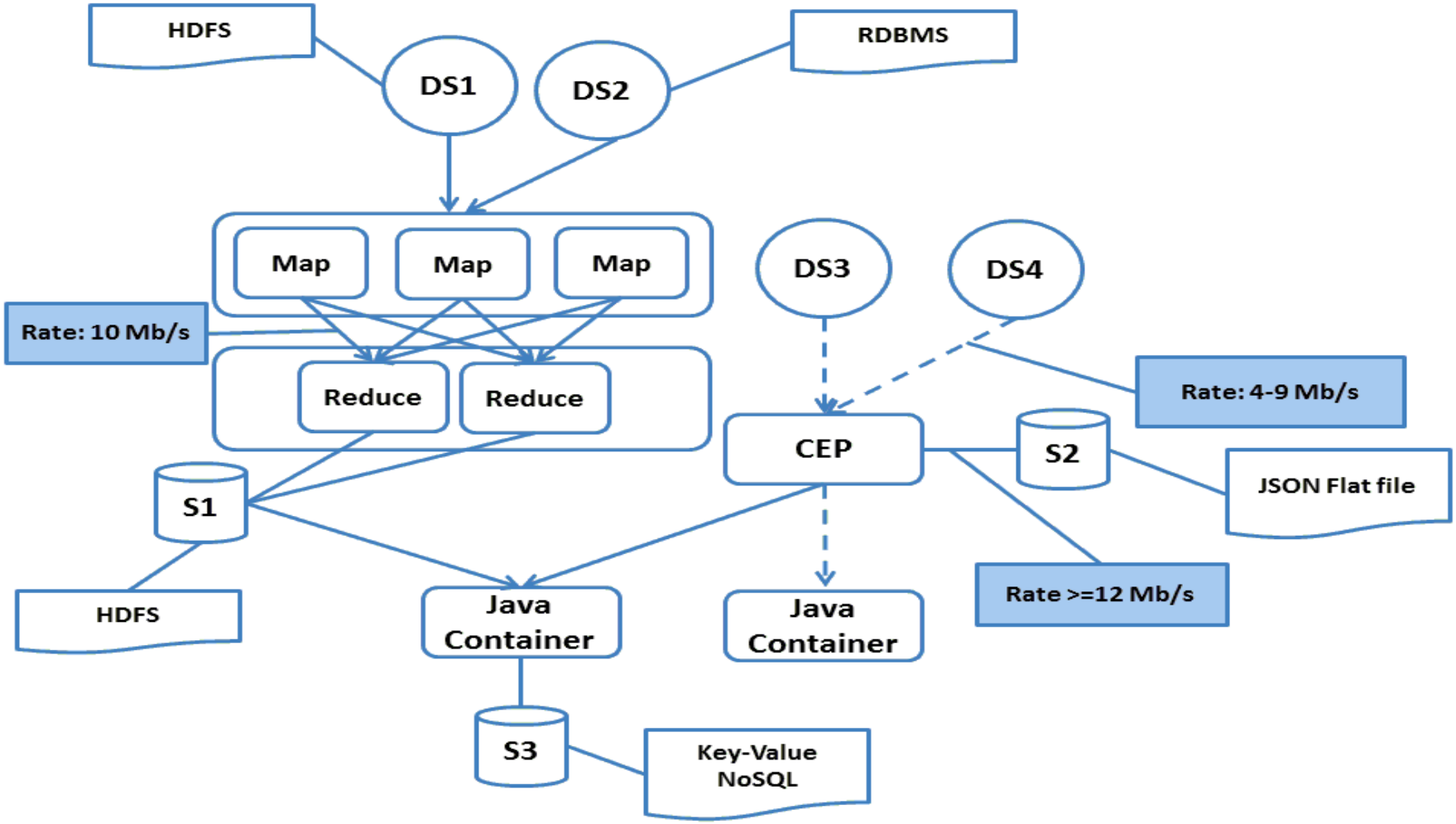
DICE Platform Independent Model (DPIM)



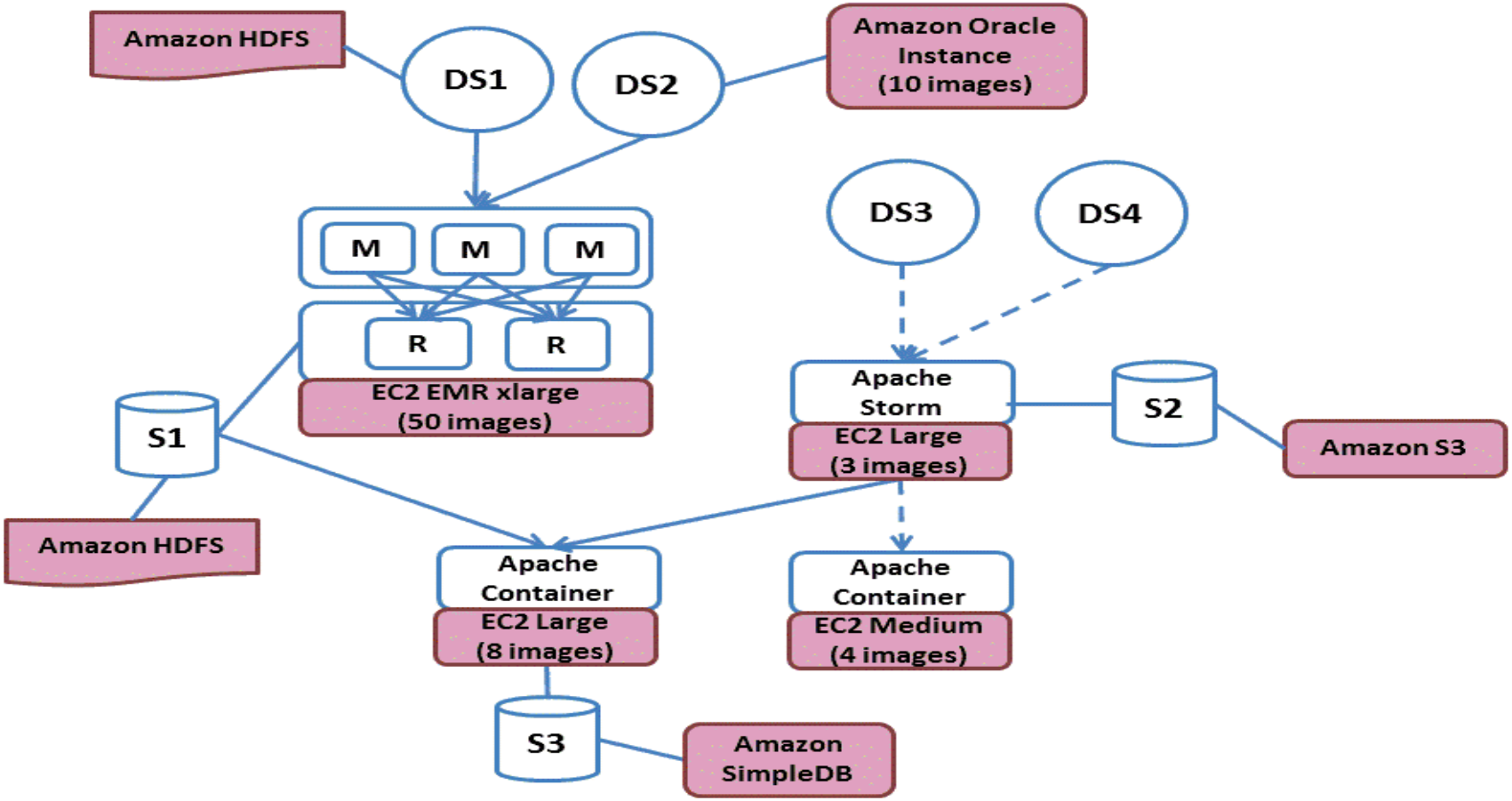
DICE Profile: PIM Level

- Functional approach to data to be expanded
  - Data dependencies
    - graph relationships between data, archives and streams
- QA focuses on quantitative aspects of data
  - Static characteristics of data
    - volumes, value, storage location, replication pattern, consistency policies, data access costs, known schedules of data transfers, data access control / privacy, ...
  - Dynamic characteristics of data
    - cache hit/miss probabilities, read/write/update rates, burstiness, ...

DICE Platform and Technology Specific Model (DTSM)



DICE Platform, Technology and Deployment Specific Model (DDSM)



DICE Profile: PSM Level

- Need for technology-specific abstractions
  - Hadoop: Number of mappers and reducers , ...
  - In-memory DBs: Peak memory and variable threading
  - Streaming: merge/split/operators, networking, ...
  - Storage: Supported operations, cost/byte , ...
  - NoSQL: Consistency policies , ...
- Generation of deployment plan
  - Proposed Chef + TOSCA extension
- Interest is both on private and public clouds
  - Private clouds more relevant for batch processing
  - Public clouds more relevant for streaming

Demonstrators

Case study	Domain	Features & Challenges
Distributed data-intensive media system (ATC)	• **News & Media** • **Social media**	• Large-scale software • Data velocities • Data volumes • Data granularity • Multiple data sources and channels • Privacy
Big Data for e-Government (Netfective)	• **E-Gov application**	• Data volumes • Legacy data • Data consolidation • Data stores • Privacy • Forecasting and data analysis
Geo-fencing (Prodevelop)	• **Maritime sector**	• Vessels movements • Safety requirements • Streaming & CEP • Geographical information

Thanks!

www.dice-h2020.eu

Challenge 2: Embracing DevOps

- Software development process is evolving
  - Developer: “I want to change my code”
  - Operator: “I want systems to be stable”
    - ...but code changes are the cause of most instabilities!
- DevOps closes the gap between Dev and Ops
  - Lean release cycles with automated tests and tools
  - Deep modelling of systems is the key to automation

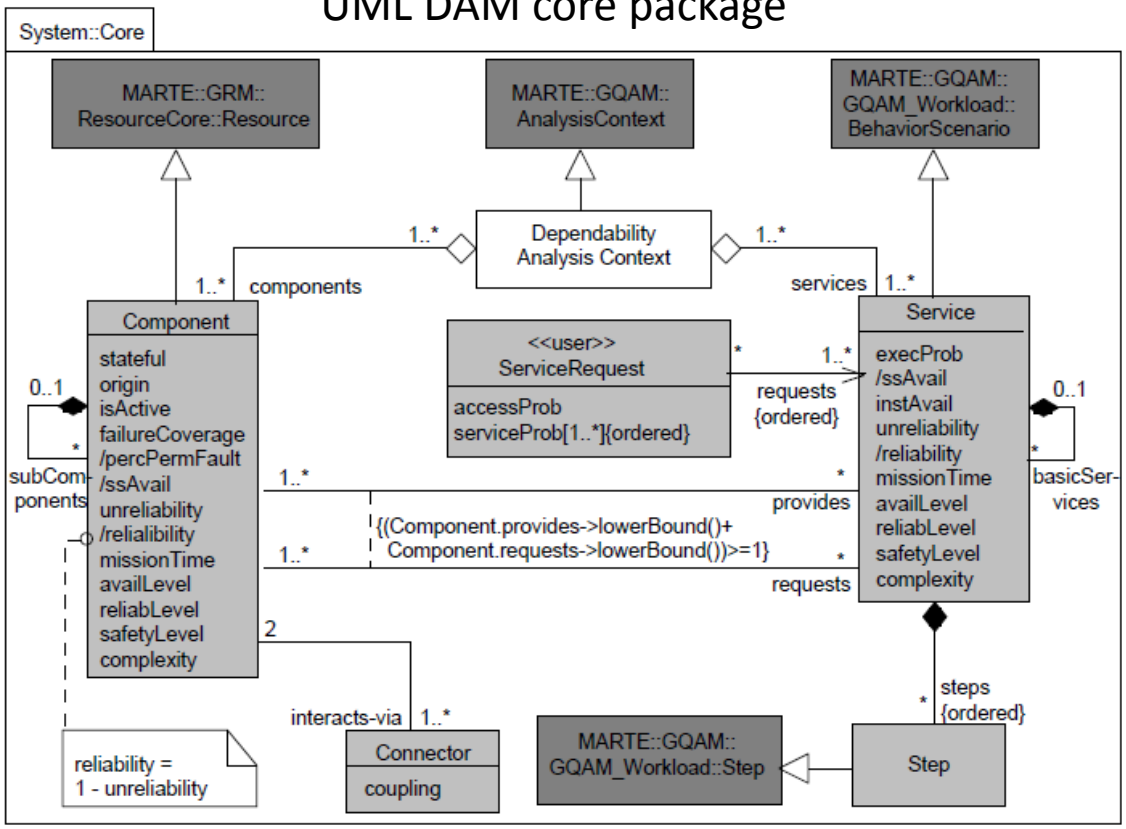
Main Technical Outputs

1. DICE Profile (WP2)
   - New UML profile to characterize data location, processing, transformation, and usage
   - Data-aware quality annotations
   - Deployment models (output to TOSCA)
2. QA Tools (WP3/WP4)
   - OSS tools (analysis, simulation, verification, feedback)
3. Integrated Development Environment (WP1)
   - Guides through the DICE methodology
4. Delivery Tools (WP5)
   - Deployment, continuous integration, testing

DICE QA: Possible Baselines

- UML MARTE
  - Performance
  - Timing
  - Verification
- MODACloudML
  - Cloud/PMI
  - Not UML
- UML DAM
  - Dependability/ZAR, covers our quality dimensions

UML DAM core package



Year 1 - Expected Achievements

Milestone	*Deliverables*
Baseline and Requirements - July 2015	• State of the art analysis • Requirement specification • Dissemination, communication, collaboration and standardisation report • Data management plan
Architecture Definition - January 2016	• Design and quality abstractions • DICE simulation tools • DICE verification tools • Monitoring and data warehousing tools • DICE delivery tools • Architecture definition and integration plan • Exploitation plan