



# D<sup>4</sup>

## Rationale, Concept & Architecture of a Distributed MCDA application Designer

R. Bisdorff (UniLu)  
Gilles Dodinet & Michel Zam (KarmicSoft, Lamsade)

5<sup>th</sup> Decision Deck Workshop  
Brest, September 2009



## Rationale

- Common requirements for any Decision Deck software
  - A. Problem data input, computation and visualization of results
  - B. Role oriented user management and collaboration
  - C. Extensibility: adding and enhancing MCDA methods



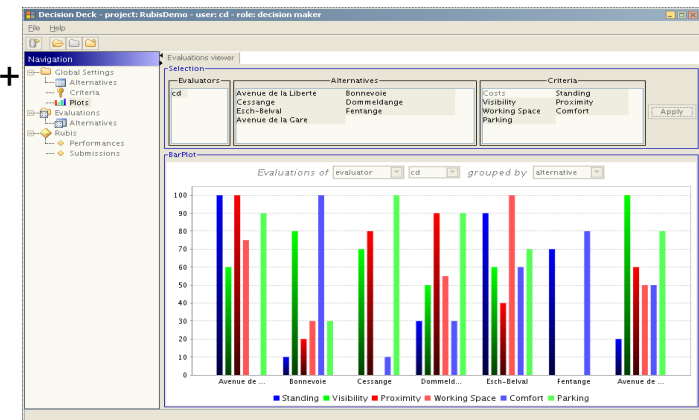
## Agenda

- Rationale
- DecisionDeck software short story
- High hopes and limitations
- D4 architecture / bricks
- D2/D3/D4/Dz interchange overview
- Q&A



## Decision Deck software story

- D2 : Desktop + plugins
- CL/SV, runs on LAN



## Decision Deck software story

- D3 : *Distributed + WS*

Welcome to the Distributed Decision Deck (D<sup>3</sup>) Server  
« ernst-schroeder.uni.lu » at the University of Luxembourg

**D<sup>3</sup>: a distributed, collaborative approach to**

The Decision Deck project aims at developir These software components implement the c methods. If you already have developed an in order to fit the Decision Deck plugin program

This **Distributed Decision Deck (D<sup>3</sup>)** server r services exposing MCDA methods. Data excf A demonstration of the D<sup>3</sup> server providing th

**Documentation**

- Concerning the Decision Deck Project:
- D<sup>3</sup> user guide: <http://decision-deck.sou>
- The D<sup>3</sup>-Rubis http resources: <http://>

## Decision Deck software story

- diviz : *desktop designer + workflow composition server*

## Decision Deck High Hopes and limitations

- Spreading MCDA methods
  - Deserving but slow growing community ...
- Objective reasons & walls
  - MCDA Method designers : requirement specification / books, algorithms
  - Java/XML developers : implementation => **skills difficult to find**
  - MCDA tools users : ....
  - Data standardization : promising => stabilization & **evolution issues**
  - Data and process **traceability** => Missing link (feedback, trust)
- But it's not MCDA's community fault !
  - Engineering limitation** => Computer science responsibility

## Expectations

D2 + D3	D4
A. Problem Data input, computation and visualization of results D2 heavy client, D3 no data input	RIA/Distributed/Cloud → deployment & ergonomy
B. Role oriented user management (D2) versus distributed workflow (D3)	Configurable user roles & processes
C. Extensibility: adding & enhancing MCDA methods D2 + plugins, D3-WS	<b>Declarative designer</b> (no Java, no XML) + MDE
	Auditing & Traceability
	Online Viral Community (Moodle-like)

**(feature) inheritance + interface (XMCA) = continuous platform**

## D4 Architecture

- Unified portal
  - “Distributed (Declarative) Designer for Decision Deck”
  - Online webapp designer/launcher for MCDA webapps
  - Experimental prototype and ongoing project
- Bricks
  - Data
  - GUI
  - Computation
  - Process
  - Auditing



9

## D2 - remastered



11

## Global entry point

### D4: Distributed MCDA webapp Designer and Launcher

user1    
Remember me

#### Description

- Create the domain model
- Design the user pages
- Execute the application

**No coding skills required**  
**No installation needed**

#### Create an account \*

Login   
First name   
Last name   
Email   
Password



10

## Data Brick

- Persistency
- Standard architecture JEE/JPA/DBMS
- Fine coarse (atomic) granularity & traceability
- Generic storage schema
- Concurrent data and schema evolution
- XSD/XML I/O (XMCD++)
- Declarative designer



12

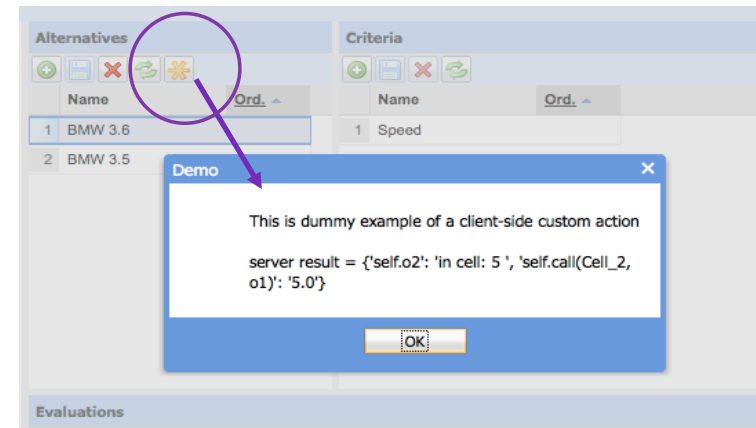
## GUI Brick

- RIA (D3 like) data centered GUI : ExtJS
- High level templates & dynamic factories (Mydraft)
- Distributed Declarative Designer

## Computation Brick

- (d2 / d3 : java plugins & WS)
- Black Box (WS + XMCD++ I/O)
- White Box (scripting languages Python, JS, ...)
  - Scripts : stored as persistent data
  - Script Editor
- Advantages
  - No Java/XML skills required, no admin deployment effort required, radical shorter application life cycle, better dissemination, open source algorithmic implementations

## Custom actions



Custom action → client code → server code → ...

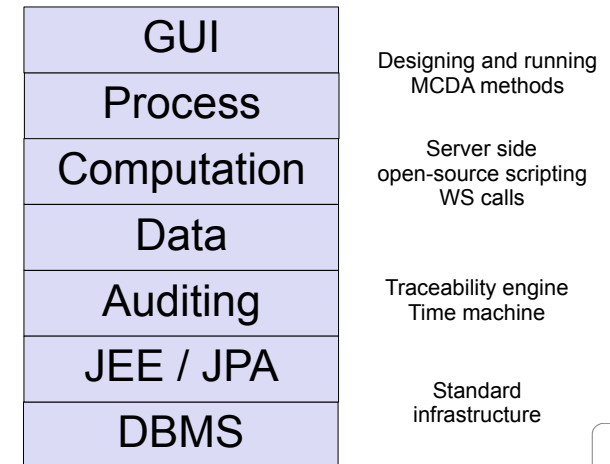
## Process Brick

- Configurable State Machines
- User roles and grants management system
- MCDA process modelling
- Abstract bricks chaining language
- Declarative Designer

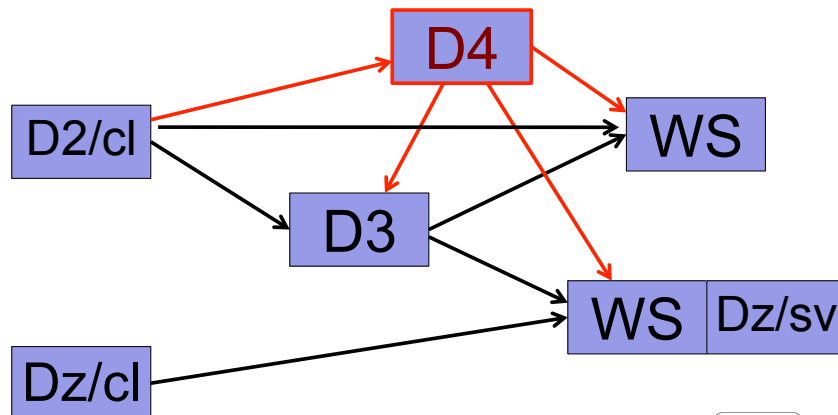
## Auditing Brick

- Fine coarse historization of data and data structure evolutions
- Process execution logs
- Karmicsoft traceability engine
- Still using the RDBMS

## D4 vision



## Global interaction



## Conclusion

### D4

- « Distributed Designer for Decision Deck »
- High-level (methodologist level) online tool
- Pre-alpha version of an experimental prototype
- Hosted : <http://leopold-loewenheim.uni.lu>
- More to come soon. Please stay tuned