Your Connection to ICT Research

SimQRi - A Query-oriented Tool for the Efficient Simulation and Analysis of Process Models

Christophe Ponsard, Quentin Boucher, Gustavo Ospina

EXE 2016, Saint-Malo (France), 3rd October 2016

PLUS HAUT
ET PLUS PROCHE
LE FONDS EUROPÉEN DE DÉVELOPPEMENT RÉGIONAL
ET LA WALLONIE INVESTISSENT DANS VOTRE AVENIR

FEDER
UNION EUROPEENNE

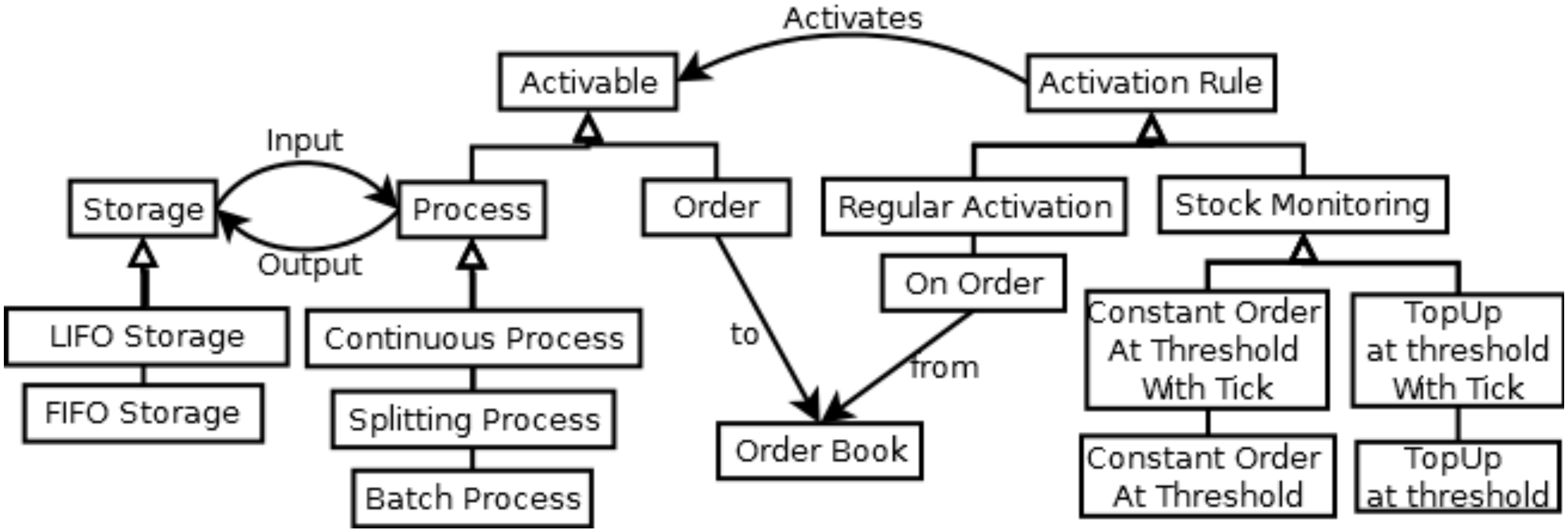
Wallonie

Centre d'Excellence en **Technologies** de l'**Information** et de la **Communication**
www.cetic.be

SimQRi Project Context

- Our context: SimQRi CORNET project - http://simqri.cetic.be
    - Quantitative Risk Management in Supply Chains
    - Main goal:
        - Develop a practical tool-supported methodology to help SMEs assessing the risks and mitigating their impact in the production process
    - Main output: modelling and simulation toolbox
        - Modelling supply chains
        - Identifying delay, quantity, bad quality,…
        - Simulate alternative designs to reduce risk

SimQRi Simulation Framework: meta-model



Tool Demo

Modelling with SimQRi tools

(The Eclipse one)

SimQRi Simulation Framework: Query Language

- *Queries for Processes:* atomic operators extracting basic metrics from processes
  - completedBatchCount(*p*) the total number of batches performed by *p*.
  - startedBatchCount(*p*) the number of (all) batches started by *p*.
  - totalWaitDuration(*p*) the total duration where *p* was not running until time *t (on all lines)*
  - meanLoad(*p*) is the ratio between of the completed batches and the total capacity
  - anyBatchStarted(*p*) true if a batch was started by the process at time *t*
  - …
- *Queries for Storages:* are atomic operators that extract basic metrics from stores
  - empty(*s*) true if *s* is empty at time *t*, false otherwise.
  - content(*s*) the number of items in *s* at time *t*.
  - capacity(*s*) the (fixed) maximal capacity of *s*.
  - totalPut(*s*) the number of items put into *s* until time *t*, not counting the initial ones.
  - totalFetch(*s*) the number of items fetched from *s* until time *t*.
  - totalLostByOverflow(*s*) the number of items lost by overflow from *s* until time *t*.
  - …
- *Operators:* Complex queries can be built using following operators
  - logical: true*,*false*,not*(!)*,and*(&)*,or*(*k*)*,<,>,...*
  - temporal logic: hasAlwaysBeen*,*hasBeen*,*since*,...*
  - arithmetic: +*,* −*,* ∗*,/,sum,...*
  - temporal arithmetic: time*,*min*,*max*,*avg*,*integral…

Tool Demo

Queries in SimQRi tools

(The Eclipse one)

Global Cost Optimisation Example

- Given the process we have presented earlier
  - Each % of element in trash after rework costs 200 PU
  - Each time unit costs 10 PU
- What is the optimum recycling rate of « Disassembly » process
  - Demo in the tool + Excel sheet

Order Treshold Optimisation Example

- Given the process we have presented earlier where each supplier has a delay of 10 days (mean value of Gaussian law, 2 days of std dev)
- What is the optimum order treshold for the different storages?

Order Treshold Optimisation Example

- Given the process we have presented earlier where each supplier has a delay of 10 days (mean value of Gaussian law, 2 days of std dev)
- What is the optimum order treshold for the different storages?

Value	Idle time of production process
10 parts	26%
20 parts	4%
30 parts	0%

Behind the Scenes

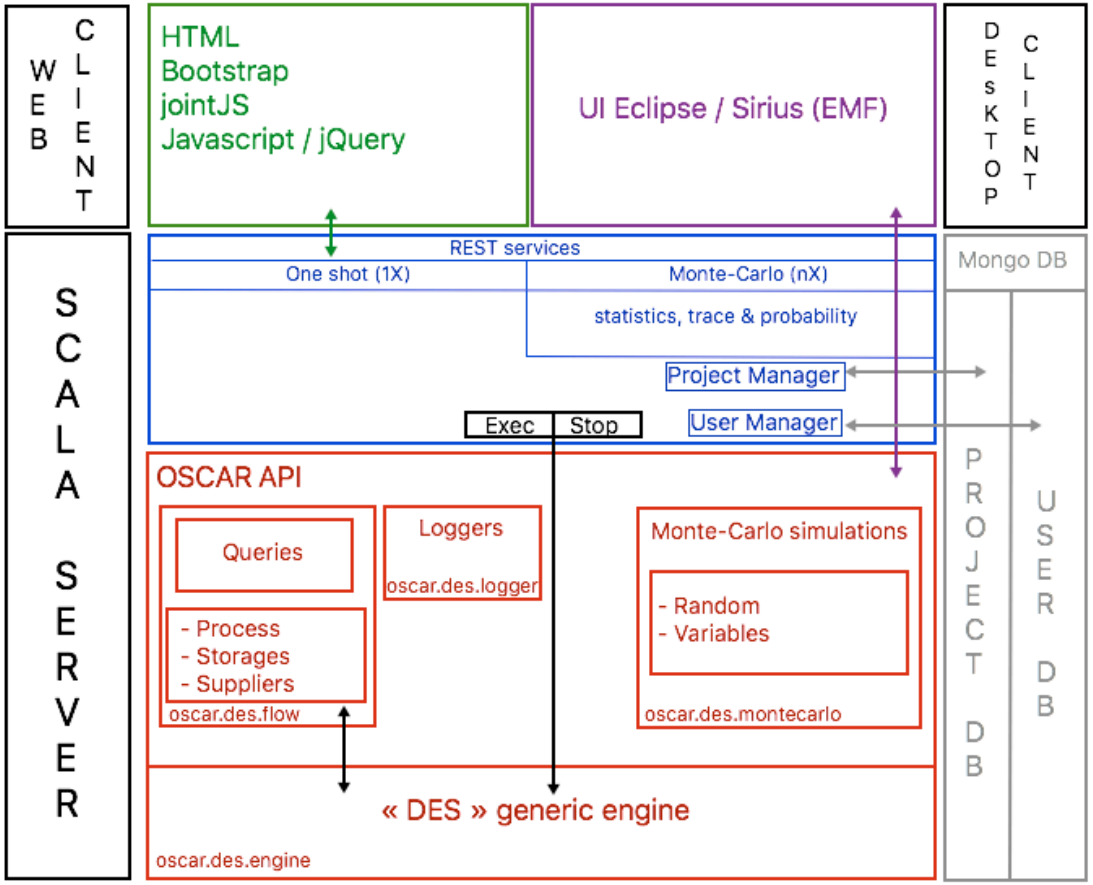
- Monte Carlo techniques for aggregating results
- Efficient query evaluations
    - Incremental evaluation throughout the simulation run
    - Minimal updates
        - Update only the relevant fragment of queries
            - Non accumulating expressions (e.g. +): evaluated only at the end
            - Accumulating expressions: evaluated at each step
    - Bottom-up updates to allow the sharing of sub-queries

Behind the Scenes (cont'd)

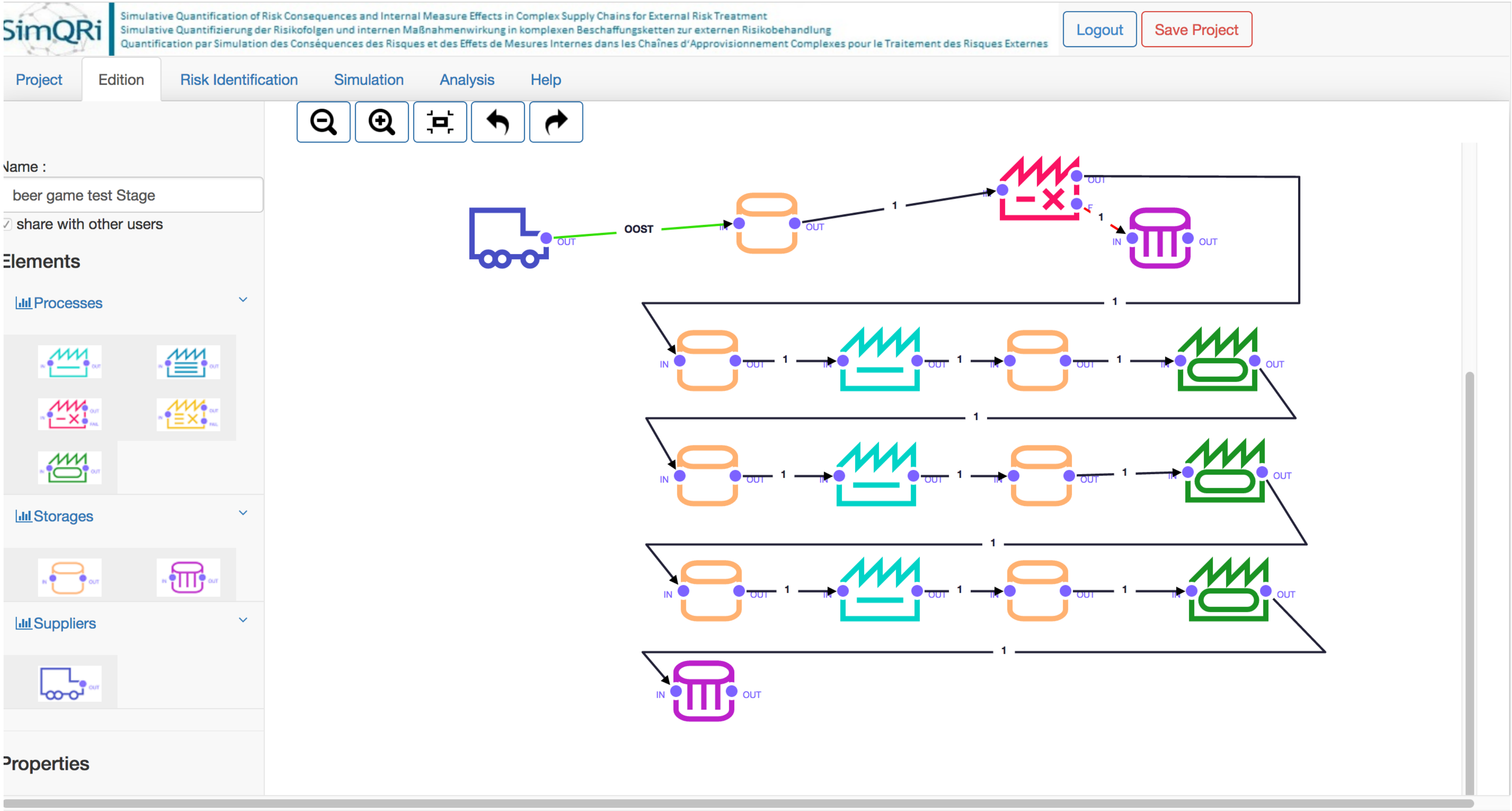
- Algorithm overview
  1. Instantiate the model and all the queries into the engine
  2. Perform one-shot analysis on the structure of the queries in order to label each sub-expression as accumulating or non-accumulating
  3. Put accumulating expressions an evaluation list (also captures the order to follow when updating them at each step)
  4. At each step of the simulation, update the accumulating expressions are updated following the evaluation list
  5. At the end of the simulation run, update the non-accumulating expressions, following a similar bottom-up process

Our Tools

- Web-based version: https://simqri.cetic.be/
- Desktop version: Eclipse plugin based on Sirius
- Shared architecture



Web-based Version

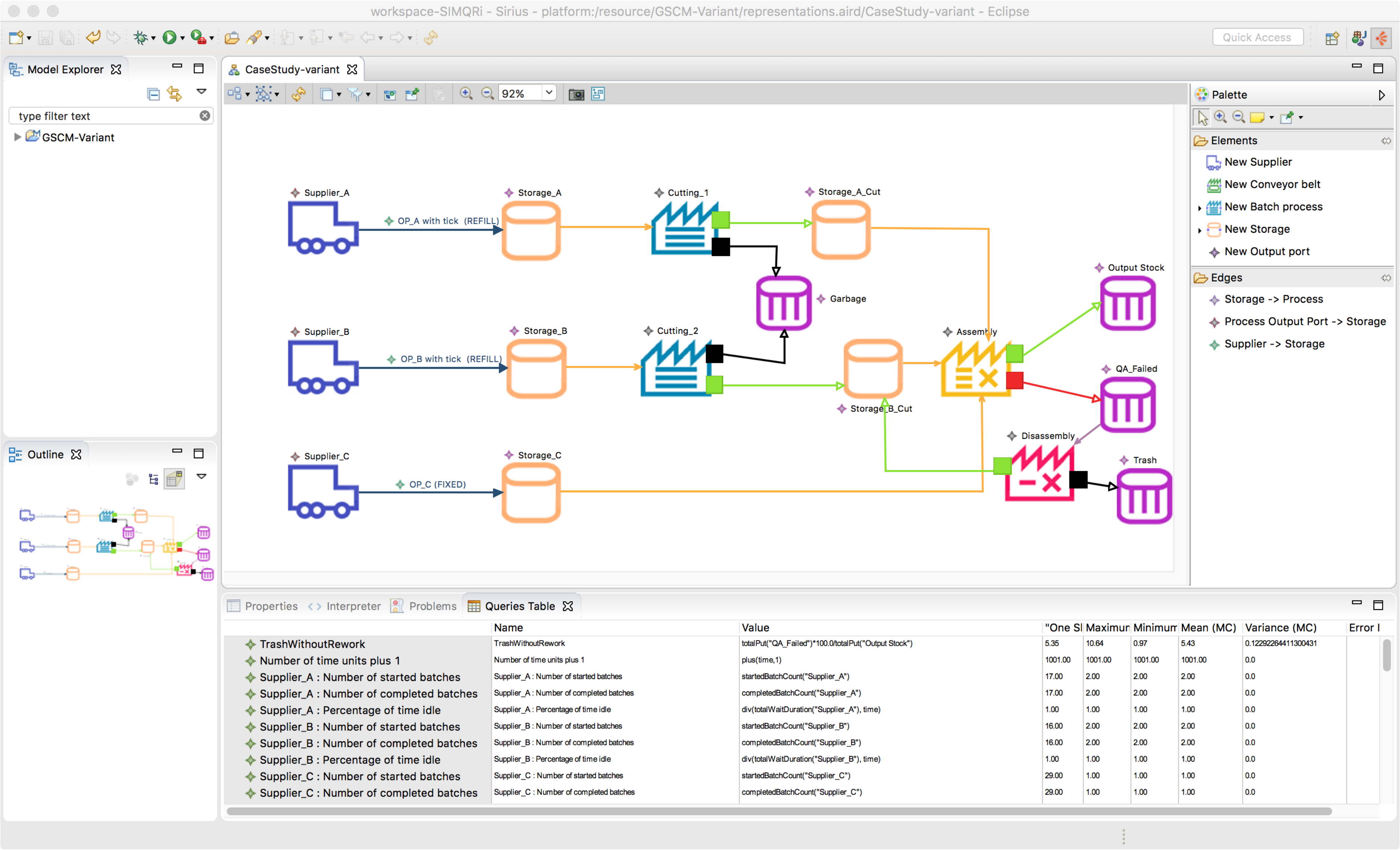


Simulative Quantification of Risk Consequences and Internal Measure Effects in Complex Supply Chains for External Risk Treatment
Simulative Quantifizierung der Risikofolgen und internen Maßnahmenwirkung in komplexen Beschaffungsketten zur externen Risikobehandlung
Quantification par Simulation des Conséquences des Risques et des Effets de Mesures Internes dans les Chaînes d'Approvisionnement Complexes pour le Traitement des Risques Externes

Name : beer game test Stage

☑ share with other users

Eclipse Version



workspace-SIMQRi - Sirius - platform:/resource/GSCM-Variant/representations.aird/CaseStudy-variant - Eclipse

	Name	Value	"One Sl	Maximur	Minimur	Mean (MC)	Variance (MC)	Error I
TrashWithoutRework	TrashWithoutRework	totalPut("QA_Failed")*100.0/totalPut("Output Stock")	5.35	10.64	0.97	5.43	0.12292264411300431	
Number of time units plus 1	Number of time units plus 1	plus(time,1)	1001.00	1001.00	1001.00	1001.00	0.0	
Supplier_A : Number of started batches	Supplier_A : Number of started batches	startedBatchCount("Supplier_A")	17.00	2.00	2.00	2.00	0.0	
Supplier_A : Number of completed batches	Supplier_A : Number of completed batches	completedBatchCount("Supplier_A")	17.00	2.00	2.00	2.00	0.0	
Supplier_A : Percentage of time idle	Supplier_A : Percentage of time idle	div(totalWaitDuration("Supplier_A"), time)	1.00	1.00	1.00	1.00	0.0	
Supplier_B : Number of started batches	Supplier_B : Number of started batches	startedBatchCount("Supplier_B")	16.00	2.00	2.00	2.00	0.0	
Supplier_B : Number of completed batches	Supplier_B : Number of completed batches	completedBatchCount("Supplier_B")	16.00	2.00	2.00	2.00	0.0	
Supplier_B : Percentage of time idle	Supplier_B : Percentage of time idle	div(totalWaitDuration("Supplier_B"), time)	1.00	1.00	1.00	1.00	0.0	
Supplier_C : Number of started batches	Supplier_C : Number of started batches	startedBatchCount("Supplier_C")	29.00	1.00	1.00	1.00	0.0	
Supplier_C : Number of completed batches	Supplier_C : Number of completed batches	completedBatchCount("Supplier_C")	29.00	1.00	1.00	1.00	0.0	

Conclusion

- We defined a meta-model for manufacturing processes based on Discrete Event Simulation with stochastic behaviour.

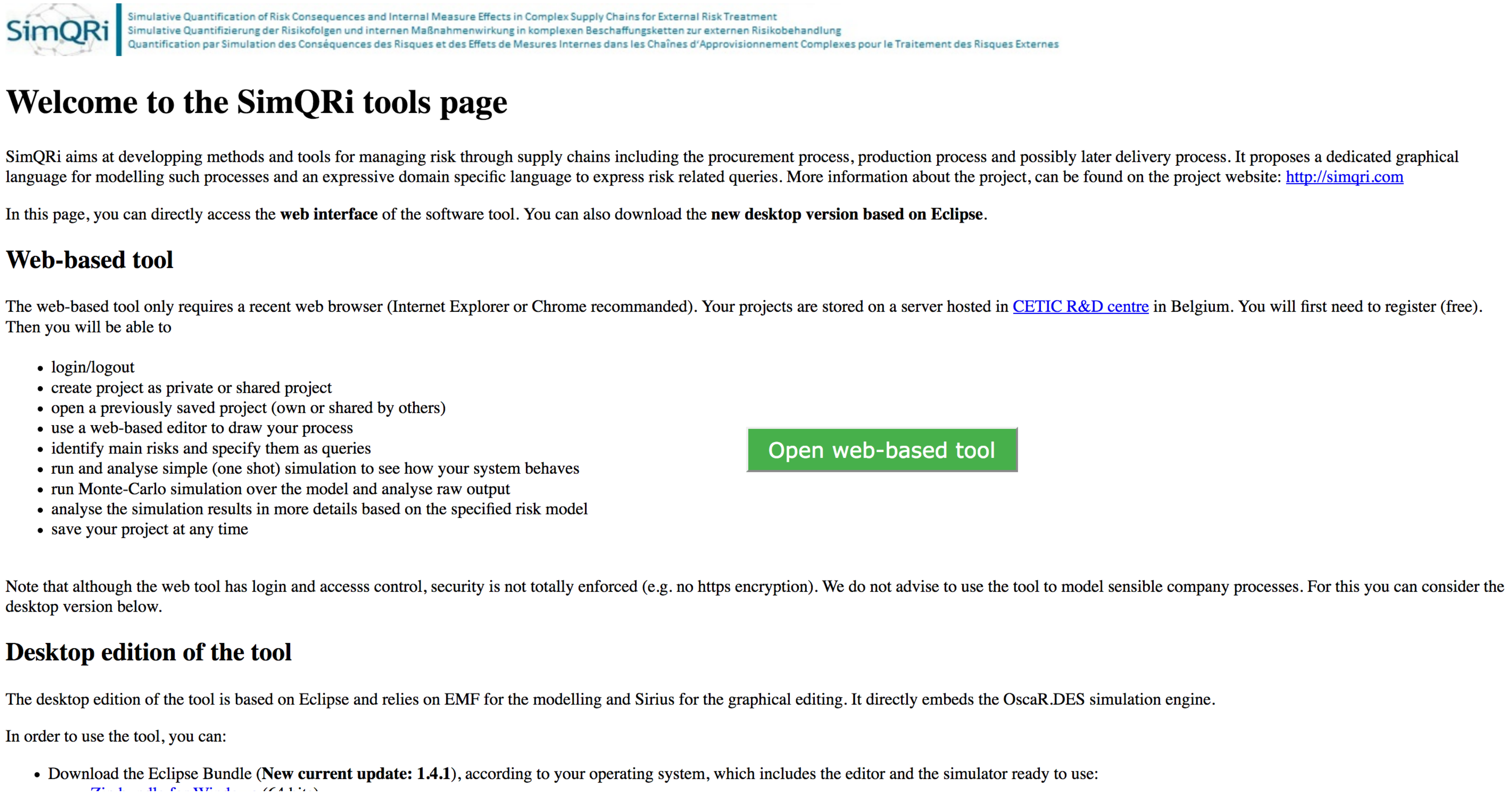
- Based on it, we developed a query language for expressing supply risks
    - Rich, extensible, efficient to evaluate

- Monte Carlo techniques for aggregating results of simulations, including queries.

- Two implementations
    - Web-based using OscaR-DES, Scalatra, MongoDB, Bootstrap, JQuery and JointJS
    - Desktop-based using OscaR-DES, Eclipse Modelling Framework and Sirius

- Available at https://simqri.cetic.be/

Tool homepage: https://simqri.cetic.be



SimQRi | Simulative Quantification of Risk Consequences and Internal Measure Effects in Complex Supply Chains for External Risk Treatment
Simulative Quantifizierung der Risikofolgen und internen Maßnahmenwirkung in komplexen Beschaffungsketten zur externen Risikobehandlung
Quantification par Simulation des Conséquences des Risques et des Effets de Mesures Internes dans les Chaînes d'Approvisionnement Complexes pour le Traitement des Risques Externes

Welcome to the SimQRi tools page

SimQRi aims at developping methods and tools for managing risk through supply chains including the procurement process, production process and possibly later delivery process. It proposes a dedicated graphical language for modelling such processes and an expressive domain specific language to express risk related queries. More information about the project, can be found on the project website: http://simqri.com

In this page, you can directly access the **web interface** of the software tool. You can also download the **new desktop version based on Eclipse**.

Web-based tool

The web-based tool only requires a recent web browser (Internet Explorer or Chrome recommanded). Your projects are stored on a server hosted in CETIC R&D centre in Belgium. You will first need to register (free). Then you will be able to

- login/logout
- create project as private or shared project
- open a previously saved project (own or shared by others)
- use a web-based editor to draw your process
- identify main risks and specify them as queries
- run and analyse simple (one shot) simulation to see how your system behaves
- run Monte-Carlo simulation over the model and analyse raw output
- analyse the simulation results in more details based on the specified risk model
- save your project at any time

Note that although the web tool has login and accesss control, security is not totally enforced (e.g. no https encryption). We do not advise to use the tool to model sensible company processes. For this you can consider the desktop version below.

Desktop edition of the tool

The desktop edition of the tool is based on Eclipse and relies on EMF for the modelling and Sirius for the graphical editing. It directly embeds the OscaR.DES simulation engine.

In order to use the tool, you can:

- Download the Eclipse Bundle (**New current update: 1.4.1**), according to your operating system, which includes the editor and the simulator ready to use:

cetic

Your Connection to ICT Research

Aéropole de Charleroi-Gosselies
Avenue Jean Mermoz 28
B-6041 Charleroi - Belgique

twitter.com/@CETIC 🇫🇷
twitter.com/@CETIC_be 🇬🇧

linkedin.com/company/cetic

info@cetic.be

+32 71 490 700

www.cetic.be

Thank you!

Christophe Ponsard

Department Manager

christophe.ponsard@cetic.be

Quentin Boucher

Senior Research Engineer

quentin.boucher@cetic.be

Gustavo Ospina

Senior Research ENgineer

gustavo.ospina@cetic.be