Goal-aware Analysis of Software License Risks

FitsumKifetew, Mirko Morandini, Denisse Munante, Anna Perini, Alberto Siena, and Angelo Susi

FBK - Fondazione Bruno Kessler
Center for **I**nformation **T**echnology
Software Engineering Group
TRENTO, Italy

iStar'17, Essen, Germany, 12.06.2017

Overview

- Introduction:
  - "Licences Risks in adoption of Open Source Software (OSS)"
- Risk Analysis Framework:
  - RiskML (Risk Modelling Language)
- Goal-aware license risk analysis
  - SUPERSEDE Case
  - Preliminary Results
- Conclusion

Introduction: OSS adoption

- Adopters' goals to adopt OSS:
  - reduction of cost and time to market
  - standards alignment
  - independence from producers

- In spite of these advantages: *"**Insufficient risk management** is one of the five topmost mistakes to avoid when implementing OSS-based solutions"* (Gartner 2011).

Introduction: License risks

- OSS projects retain several different (missing) licenses. If it is not correctly managed, several license risks can be raised
  - licenses violations
  - potential legal issues

- It affects adopters' goals:
  - possible forms of free and commercial redistribution
  - compatibility with other licences (forms of attribution, license modifiability, …)
  - market penetration
  - reputation

Objective: Prevent these risks

How can we prevent or warn of these risks?

Objective: Prevent these risks

How can we prevent or warn these risks?

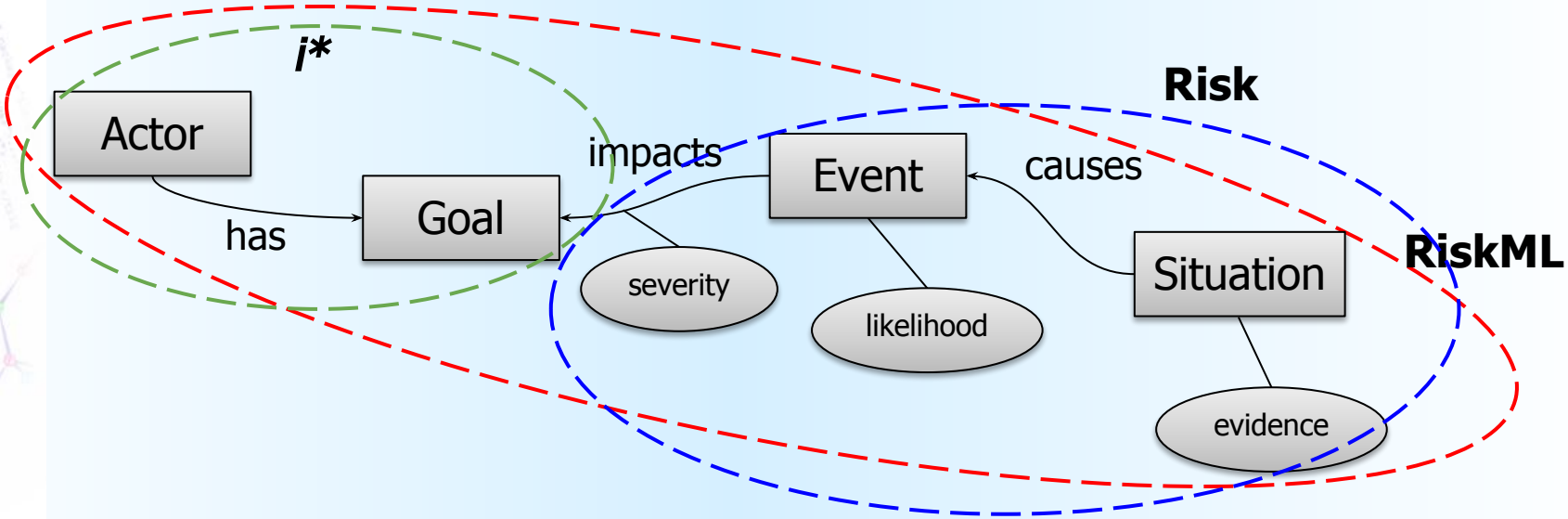
Performing a OSS licensing analysis!

But how?

- Using a risk analysis framework
  - "**RiskML+i***" is a framework to model and analyse risk exposure, and how it harms the adopters' goals.

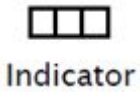
What is "Risk"?

- Risk is the effect of **uncertainty** on objectives [ISO31000:2009]
- Risk is a combined measure representing :
  (i) the adverse impacts that would arise if an event occurs &
  (ii) the likelihood of its occurrence. [NIST 2012,CORAS]



RiskML: a modelling language that implements the notion of risk and binds it to OSS data

RiskML: language concepts



Indicator: abstract representation of a measure that gives

Situation: a state of affairs which allow a certain event to happen.

- *sat(φ): satisfaction of being in this state*

Event: a change in the state of affairs, with a potential negative impact on goals.

- *lik(φ): likelihood* of the event.
- *sev(φ): severity* for a stakeholder's goals

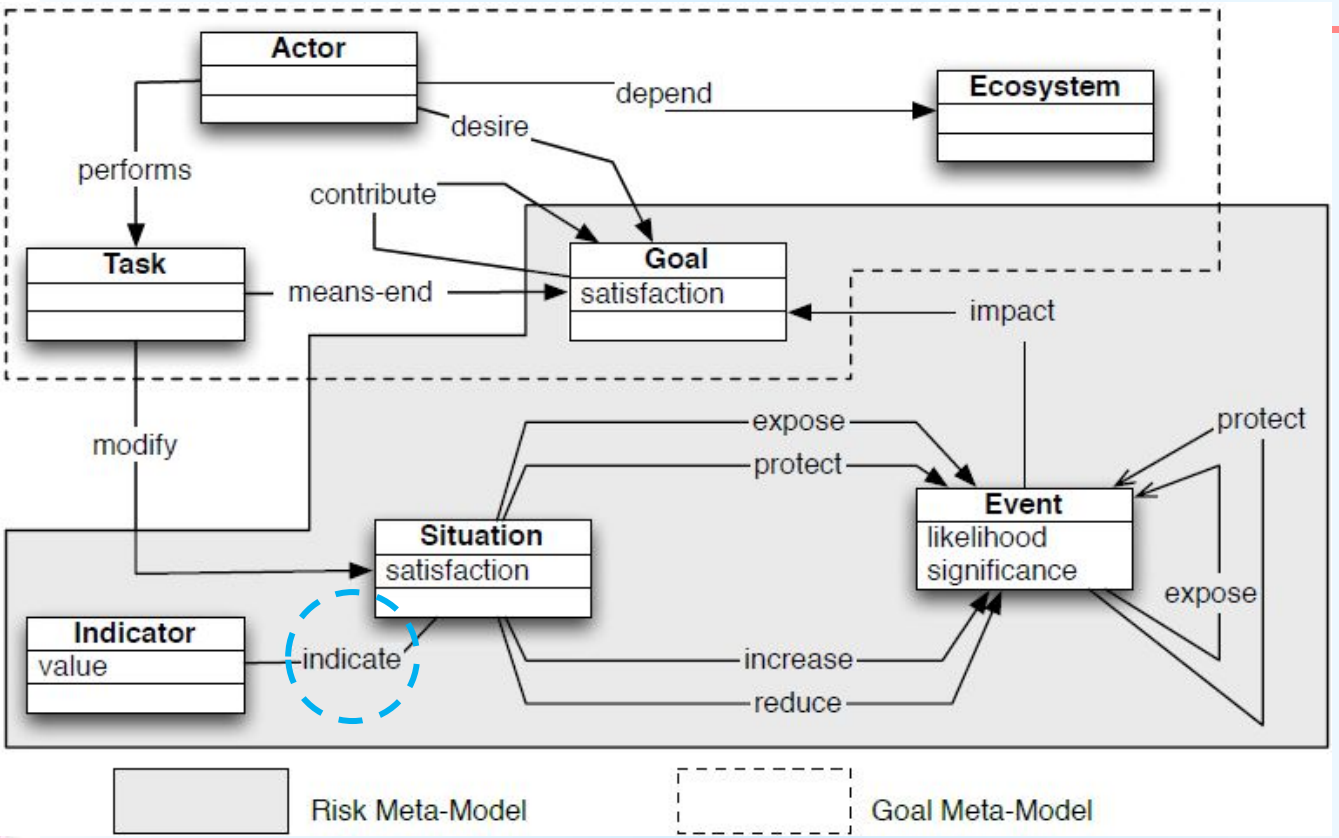
} *exposure*

Goal: a state of affairs desired by the stakeholder

Risk: expresses a lack of knowledge about some happening and its consequences, as a **tuple**

`«situations, event, impact to goals»`

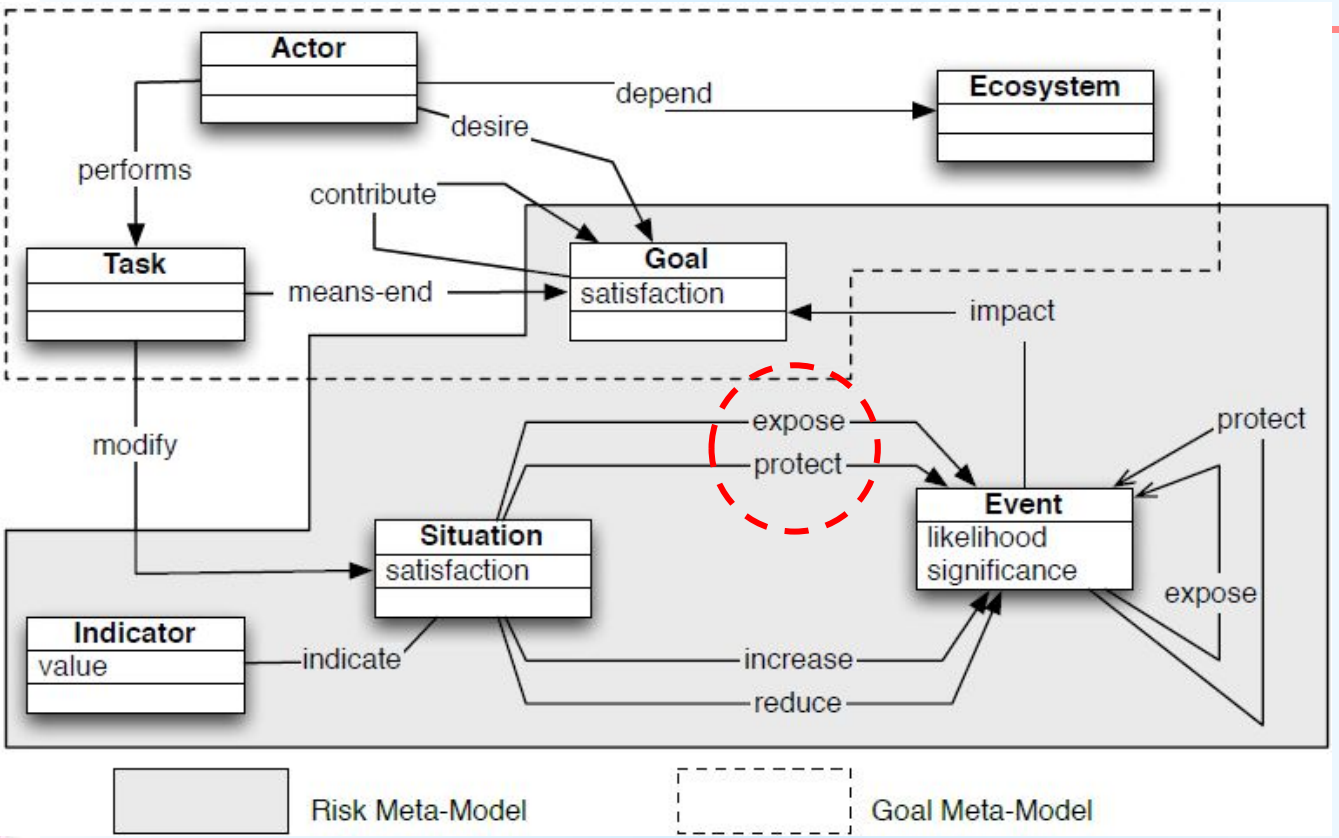
RiskML: relations (1/5)



Relations base on the propagation of **evidence**:

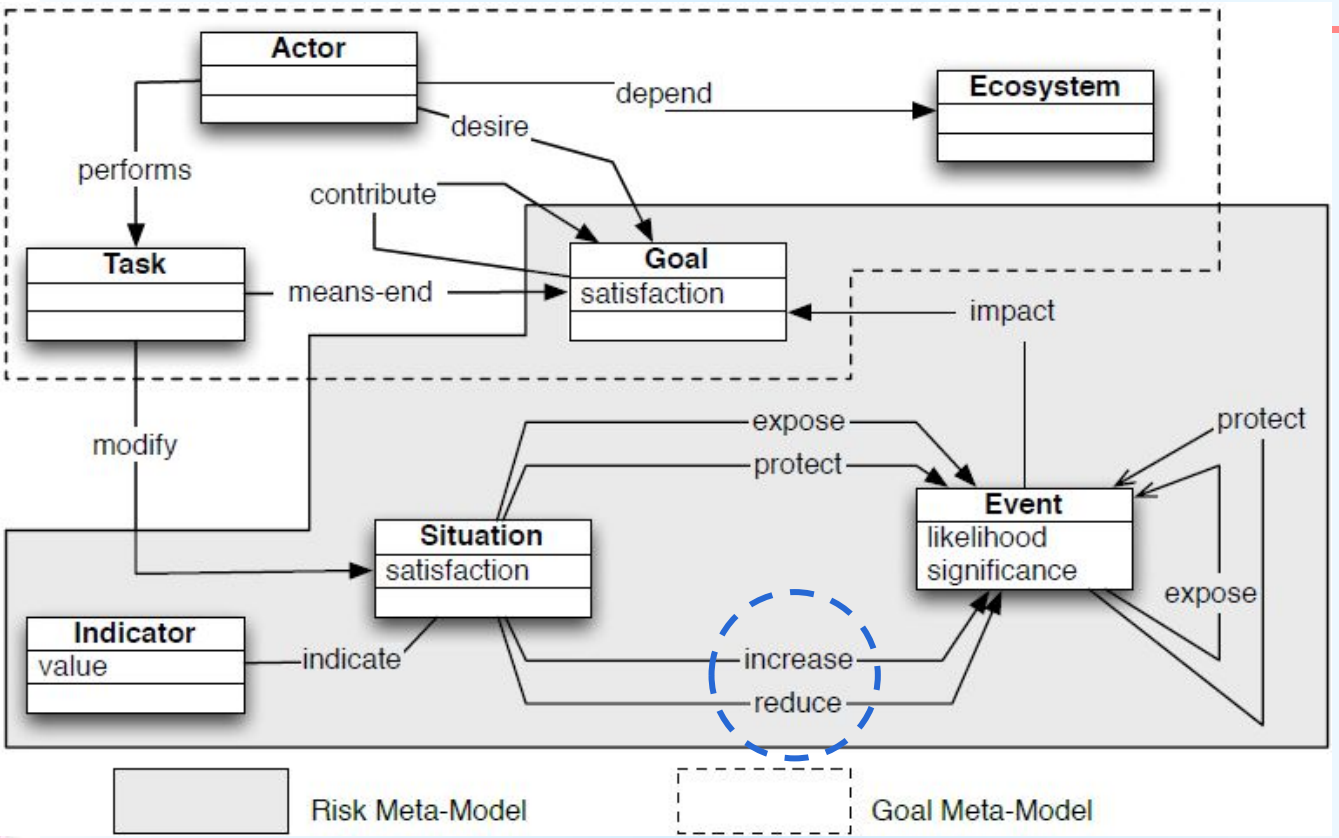
- Indicate: indicator value → evidence of situation satisfaction

RiskML: relations (2/5)



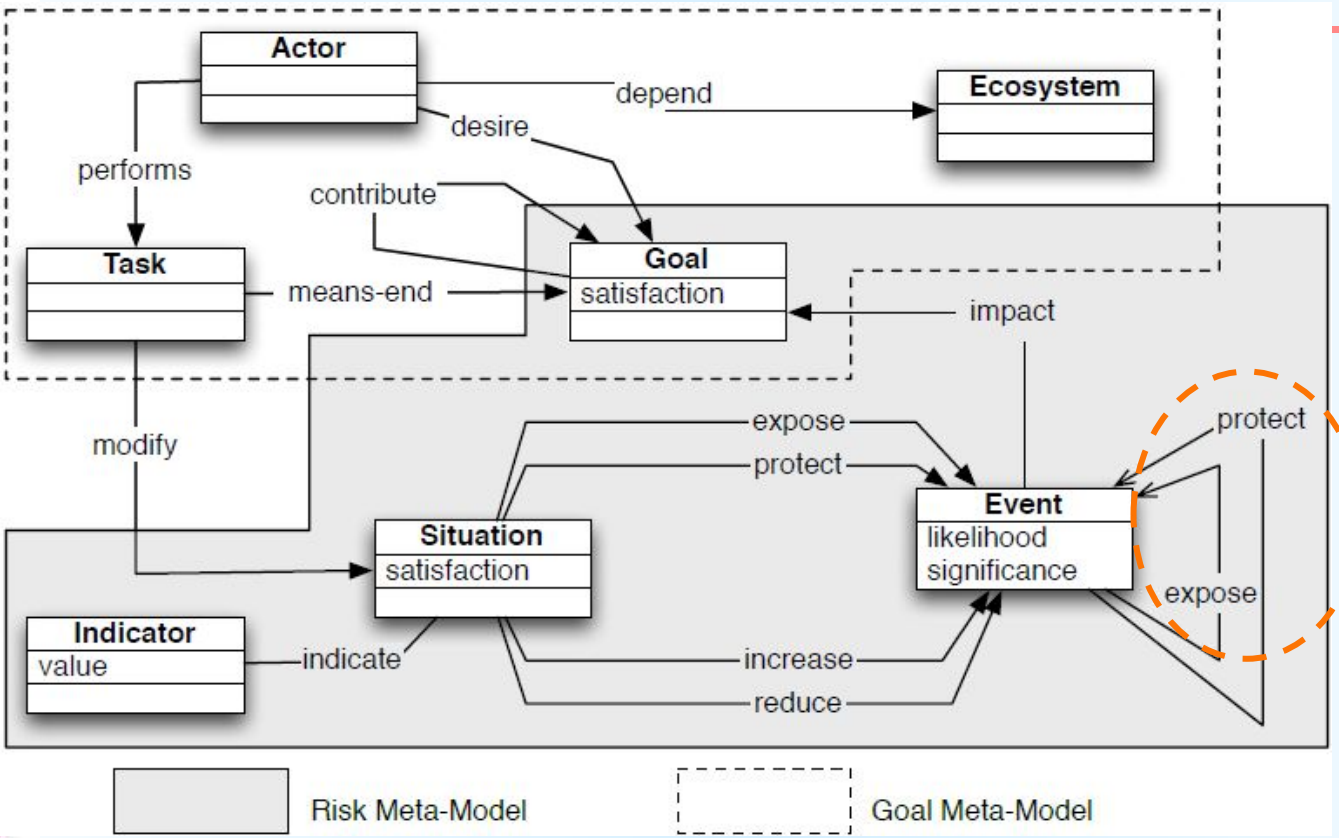
- Expose: higher satisfaction evidence → higher likelihood
- Protect: higher satisfaction evidence → lower likelihood

RiskML: relations (3/5)



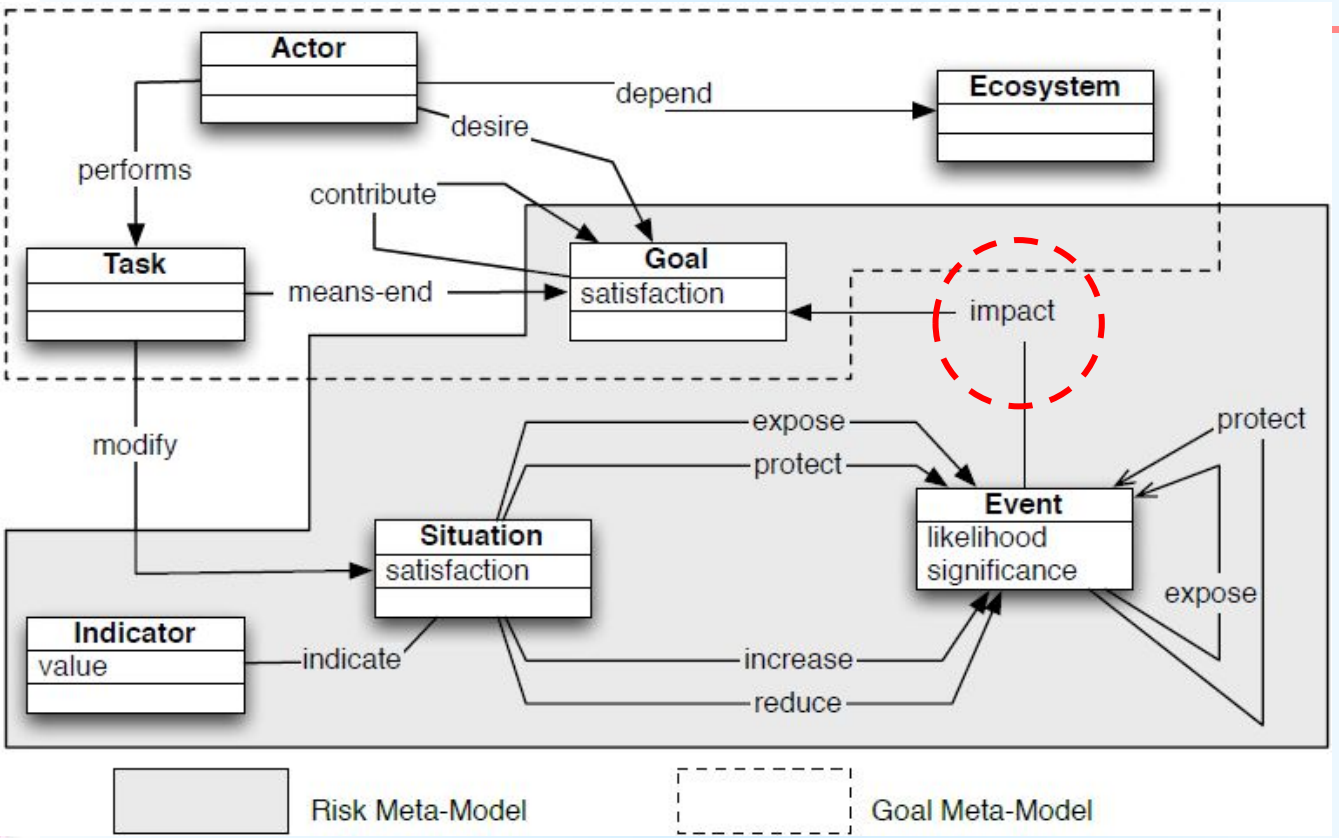
- Increase: higher satisfaction evidence → higher severity
- Reduce: higher satisfaction evidence → lower severity

RiskML: relations (4/5)



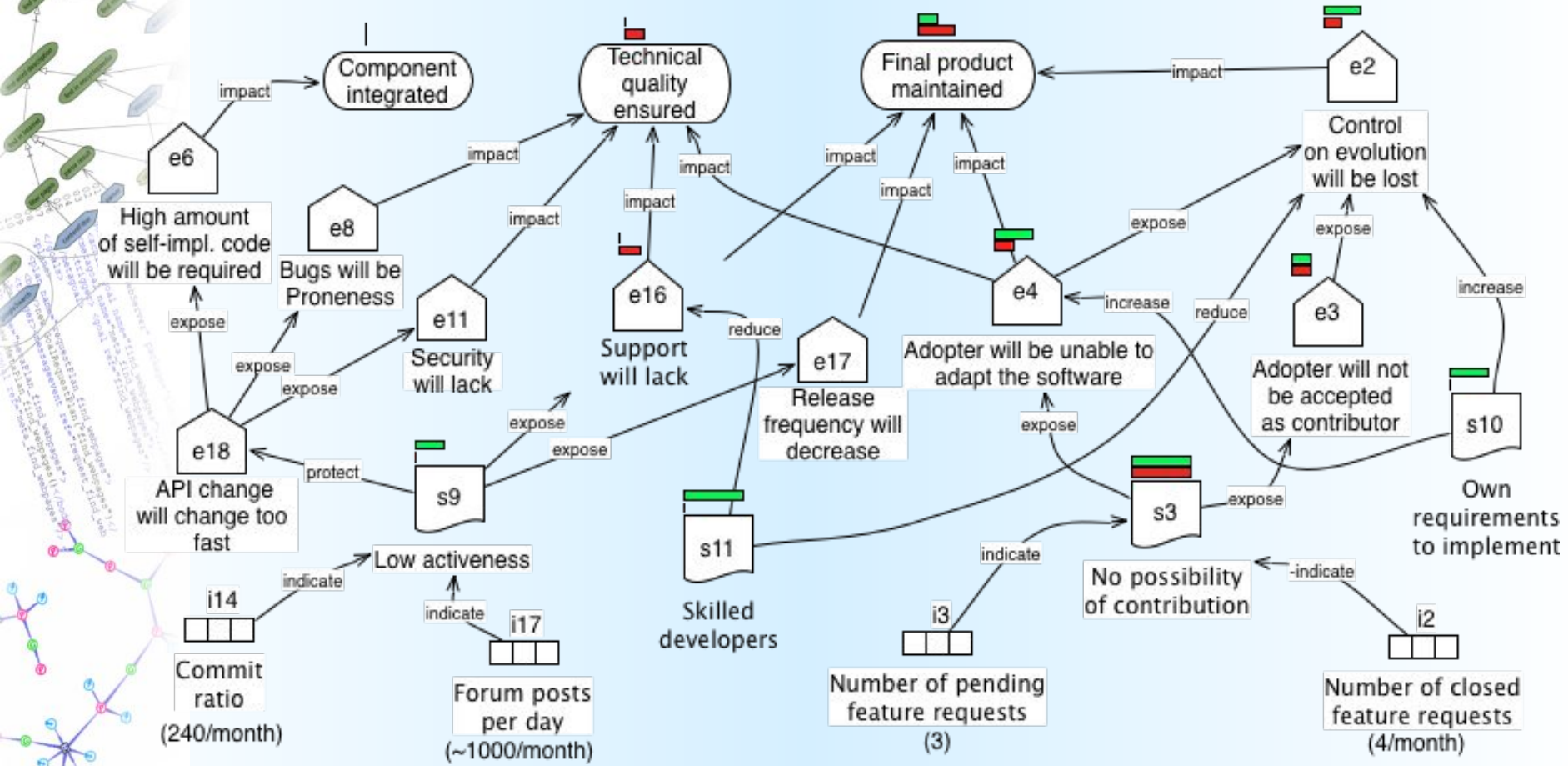
Relations base on the propagation of **effects between events**.

RiskML: relations (5/5)

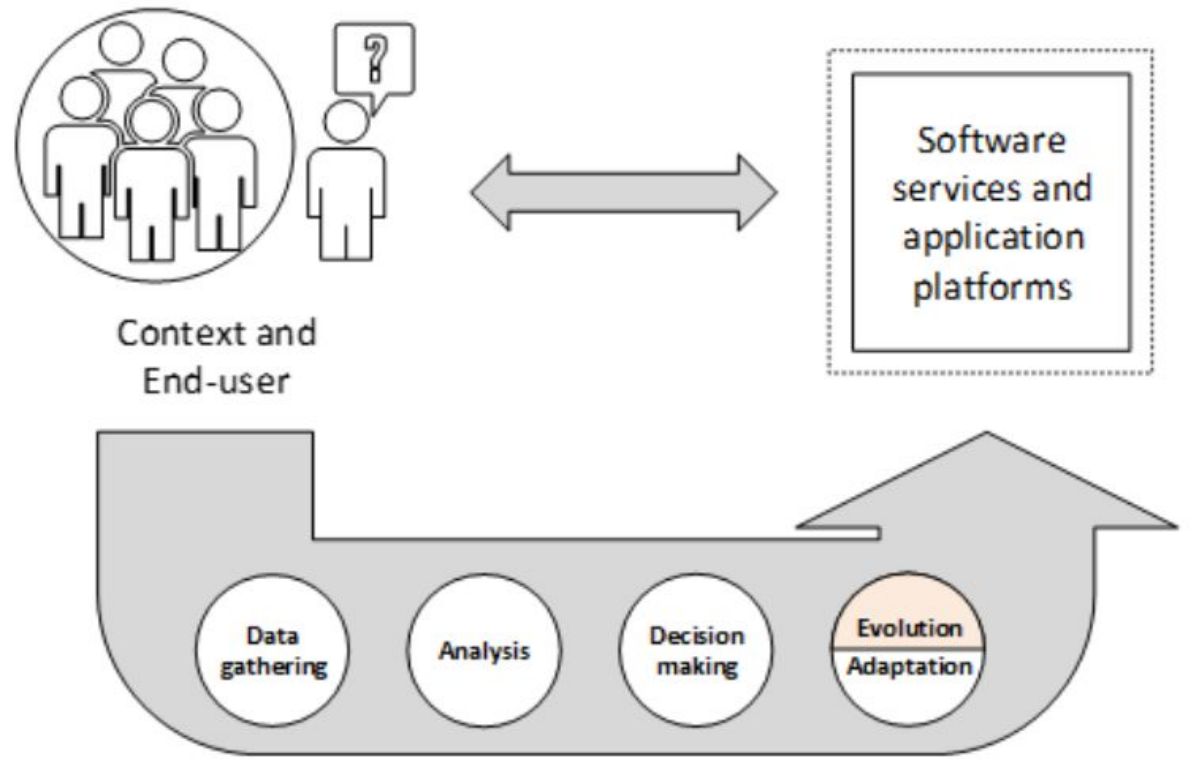


- Impact: event exposure → severity of impact to goal satisfaction

Risk evaluation



Goal-aware license risk analysis



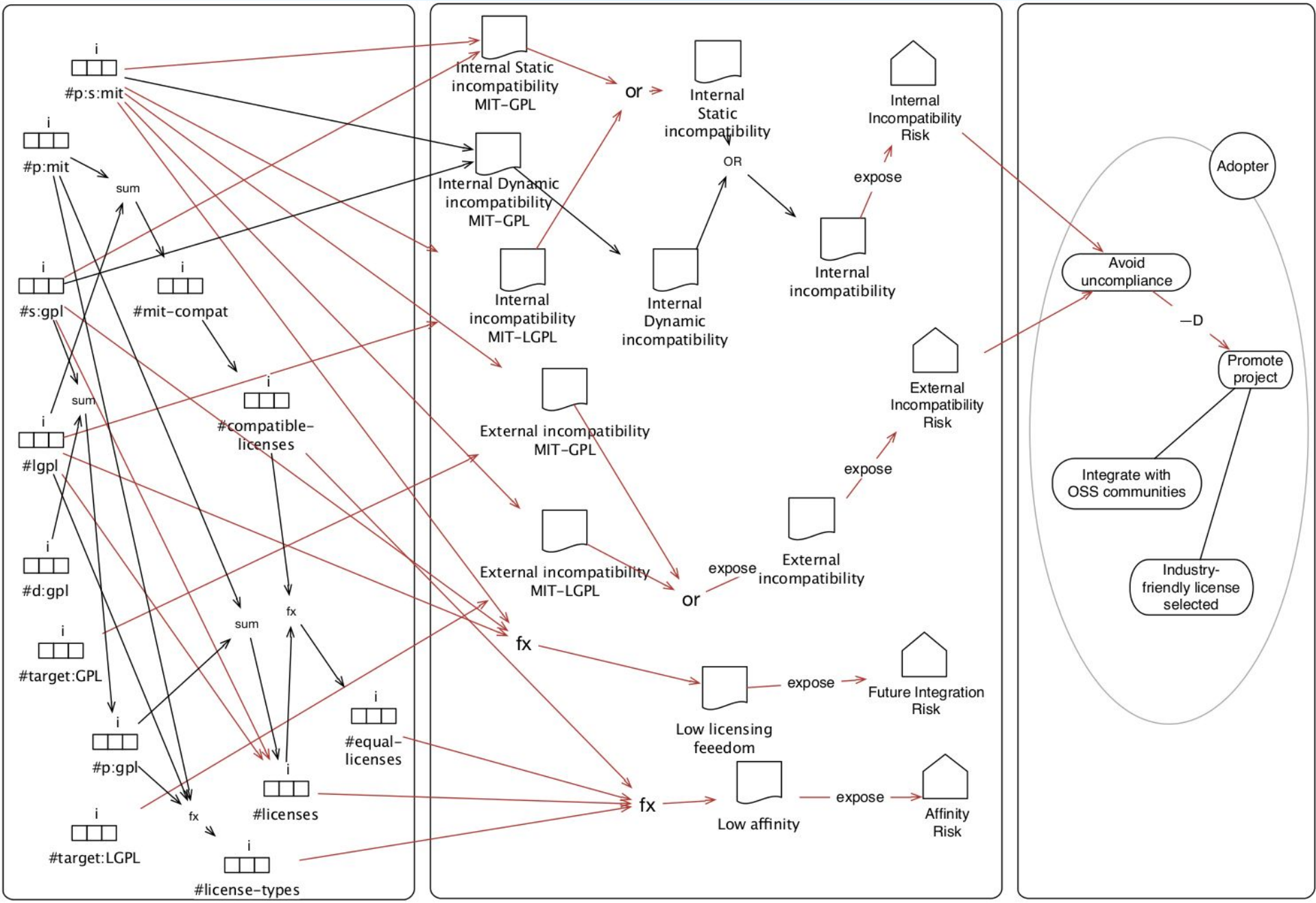
SUPERSEDE

Goal-aware license risk analysis

- SUPERSEDE goals to select appropriate licenses:
  - increase the project visibility and the acceptance in the industry
  - foster the integration with OSS community
  - avoid to generate legal issues
- RiskML was used to achieve these goals. Two main steps were performed:
  - (1) Modelling *licensing risks* to identify *indicators*, *situations*, *events* and *goals* =>
    SotA + OSS licensing experts opinions
  - (2) Analysing the licensing risk exposure

Goal-aware license risk analysis

- (1) Modelling *licensing risks* :
  - 3 goals, e.g. industry-friendly license selected
  - 17 licensing indicators, e.g. number of GPL licenses
  - 12 types of risks:
    - internal incompatibility,
    - external incompatibility,
    - lack of affinity,
    - future uncertainty,
    - reduced target license set,
    - declining components/target licenses,
    - infrequent components/target licenses,
    - lack of knowledge,
    - obsolete components/target licenses.


i
#p:s:mit
i
#p:mit
sum
i
#s:gpl
i
#mit-compat
sum
i
#lgpl
i
#compatible-licenses
i
#d:gpl
sum
fx
i
#target:GPL
i
#p:gpl
i
#equal-licenses
i
#licenses
fx
i
#target:LGPL
i
#license-types
Internal Static incompatibility MIT-GPL
Internal Dynamic incompatibility MIT-GPL
Internal incompatibility MIT-LGPL
External incompatibility MIT-GPL
External incompatibility MIT-LGPL
or
Internal Static incompatibility
OR
Internal Dynamic incompatibility
Internal incompatibility
expose
Internal Incompatibility Risk
External Incompatibility Risk
expose
External incompatibility
expose
or
fx
fx
Low licensing feeedom
expose
Future Integration Risk
Low affinity
expose
Affinity Risk
Adopter
Avoid uncompliance
—D
Promote project
Integrate with OSS communities
Industry-friendly license selected
Measure layer
Risk layer
Goal layer

Goal-aware license risk analysis

- (1) Modelling *licensing risks* - gathered information:
  - 25 components
  - 194 OSS libraries:
    - 176 with 10 different known licenses:
      ASL2, CPL-EPL, MIT, …
    - 18 with licenses whose nature was either *unknown* or *not captured by the model* developed in RISCOSS (only 17 licenses were identified), for 1 license was not-founded.

Number of components	25
Number of OSS libraries	194
ASL2	67
BSD3	3
BSD4	1
CC3.0	1
CDDL	9
CPL-EPL	31
GPL2	4
LGPL2.1	3
LGPL3+	5
MIT	31
Other/unknown	18

Goal-aware license risk analysis

- (2) Analysing the licensing risk exposure:
  - Objective: identify potential violations as cause of strategic failures.
  - Results: 5 license violations
    - The presence of components with GPL2 license, which are not compatible with non-GPL2 licenses.
    - Example: releasing a system (*DMGame* in Decision Making Package of SUPERSEDE) using Apache Software Foundation 2.0 (ASL2) but one of the components of the system has a GPL2 license.

Conclusion

- We introduced a licensing risk model to capture an important part of the expert knowledge.
- It allows to create risk awareness for non-expert analysts about the impact of risks on the organisational goals.
- In the SUPERSEDE context, RiskML allowed to obtain a preliminary result about licenses violations.

Thank you!

Questions,
Feedback?