Model-based Reasoning Tool for Software Ecosystems

Authors: Solomon Berhe, Rishabh Mankatala

Tool: releasetrain.io

Over the past decades, software domains have witnessed a trend towards faster software release cycles, an increase of software components, and their connectivity. Examples include products in Industry 4.0 domains such as health care, retail, or mobility, in which, technically, software components are connected to a software ecosystem. Faster release cycles and high connectivity make many software ecosystem updates an expensive, arduous, and risky maintenance task. This poster leverages software release metadata to reason about the maintenance cost of software ecosystem updates. In particular, the data is applied to model a software ecosystem graph and subsequent calculate its maintenance cost. To illustrate, the model is prototyped and evaluated in three Industry 4.0 use cases.

Research Question: How to improve reasoning of software ecosystem updates?

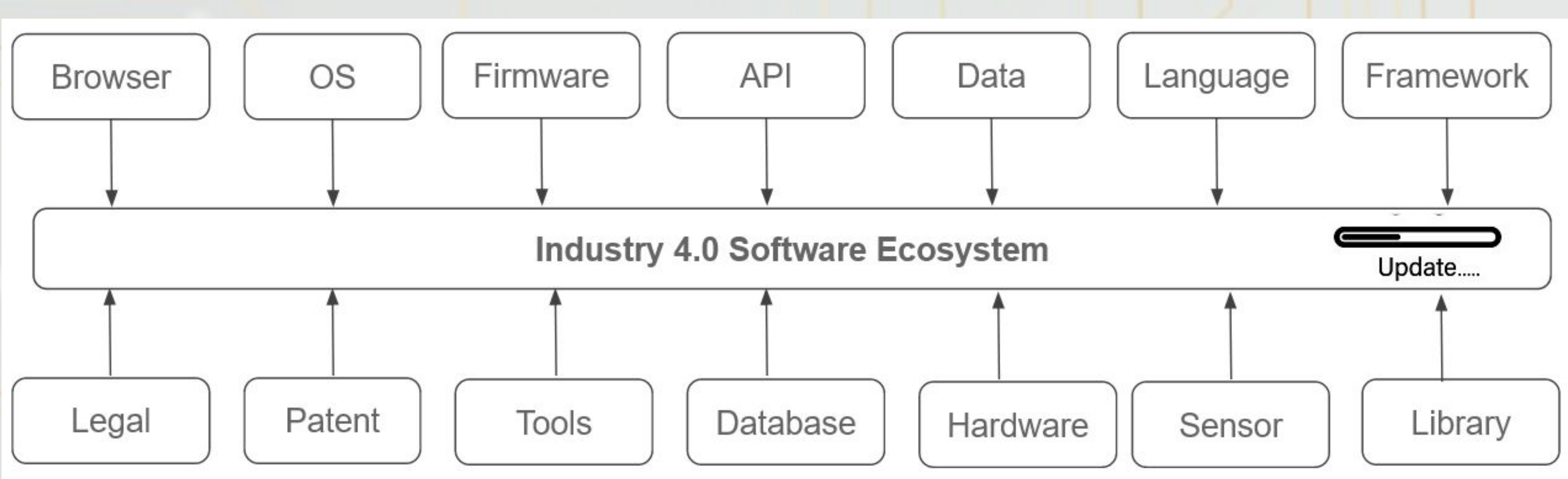
1. Ignore, automate, or impact evaluate an update?
2. Who is responsible, accountable, consulted, informed for an software update?
3. What ecosystem components are affected by an update?
4. What ecosystem teams are required to address an update?

Objective: Leverage software release data to quantify the maintenance cost of software ecosystem updates

1. Model a software ecosystem graph and calculate its maintenance cost using the data
2. Prototype and evaluate the model in three Industry 4.0 use cases

Methodology:

3. Collection of software release metadata
4. Modeling of software ecosystem graph
5. Calculation of maintenance cost using graph analysis algorithms



Use Case 1: Health Care and Java Ecosystem

A health care software ecosystem is implemented with Java as the main programming language. It includes both, internal and external components. For the ecosystem team, careful and **early impact evaluation as well as fast patching of CVEs are a main priority** to ensure there is no risk with regard to patient data. This month several CVE updates were released. Due to an upcoming holiday season most teams are on vacation next week. As a result, the ecosystem team shall request updating the higher risk CVE by addressing the question:

Q1: *What ecosystem components are potentially most affected by a CVE update?*

Use Case 2: Retail and Python Ecosystem

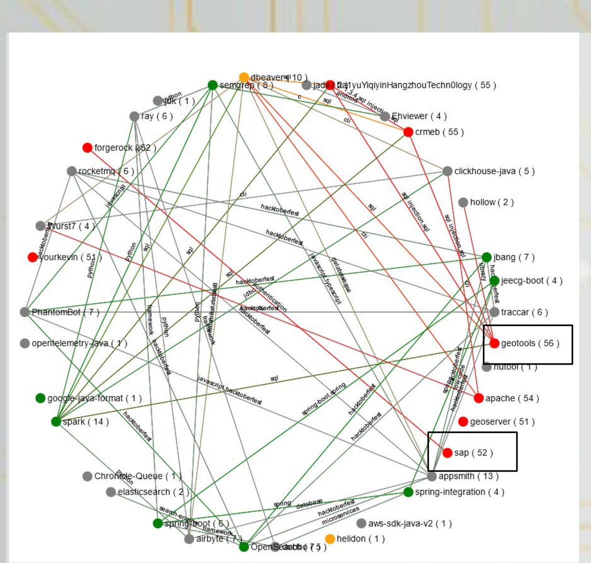
An retail startup leverages the Python ecosystem to develop an Industry 4.0 product. Due to limited testing capacity, they would like to **optimize where to invest the main testing team effort**. They assume patch and minor updates require more component testing, whereas major and CVE updates require more integration and system testing. Towards this objective, they need to address the question:

Q2: *What ecosystem testing teams are required to address an update?*

Use Case 3: Automotive and Software Stack Ecosystem

Many Industry 4.0 domains are based on traditionally companies manufacturing hardware such as vehicles, bicycles, or scooters with much slower hardware release cycles. As a result, software updates are often impeded by hardware related constrained, leading to infrequent software updates. In this use case, an automotive domain ecosystem team is constrained to perform yearly software updates. To address this impediment, when planning the software stack their overall objective is to **pick an ecosystem with the least maintenance cost** and address the question:

Q3: *What is the maintenance cost of a software stack ecosystem?*



Instructions Ecosystem:
- Customize table ecosystem weights
- View / Zoom ecosystem graph
- View ecosystem component (cost)
- View ecosystem maintenance cost

Ecosystem:	java	
Attribute	**Weight**	**Count**
Component Count	1	36 (x 1)
Link Count	1	112 (x 1)
Version Count	1	50
Version Major	0	0
Version Minor	0	0
Version Patch	0	0
Version CVE	50	400
Version Invalid		0
Maintenance Cost		148

Instructions Component:
- Click graph component node
- View component maintenance cost
- View latest update maintenance cost

Component:	geotools
Attribute	**Count**
Link Count	5 (x 1 weight)
Version Count	17 (x 1 weight)
Version Major	0 (x 0 weight)
Version Minor	0 (x 0 weight)
Version Patch	0 (x 0 weight)
Version CVE	1 (x 50 weight)
Cost Component	56
Cost Latest Version	51

Q1:*What ecosystem components are potentially most affected by a CVE update?*
A1: *Geotools or SAP ("Software vendor SAP has released security updates for 19 vulnerabilities, five rated as critical, meaning that administrators should* ***apply them as soon as possible*** *to mitigate the associated risks.")*

Ecosystem:	python	
Attribute	**Weight**	**Count**
Component Count	1	37 (x 1)
Link Count	1	142 (x 1)
Version Count	1	50
Version Major	1	2
Version Minor	1	9
Version Patch	1	35
Version CVE	1	4
Version Invalid		0
Maintenance Cost		179

Integration: 6
Component: 44

Q2: *What ecosystem testing teams are required to address an update?*
A2: Component Testing

LAMP Stack

Ecosystem:	linux,apache,mysql,php	
Attribute	**Weight**	**Count**
Component Count	1	32 (x 1)
Link Count	1	246 (x 1)
Version Count	1	50
Version Major	0	0
Version Minor	0	0
Version Patch	0	0
Version CVE	0	0
Version Invalid		0
Maintenance Cost		278

WAMP Stack

Ecosystem:	windows,apache,mysql,php	
Attribute	**Weight**	**Count**
Component Count	1	32 (x 1)
Link Count	1	276 (x 1)
Version Count	1	50
Version Major	0	0
Version Minor	0	0
Version Patch	0	0
Version CVE	0	0
Version Invalid		0
Maintenance Cost		308

MEAN Stack

Ecosystem:	mongodb,express,angular,node	
Attribute	**Weight**	**Count**
Component Count	1	14 (x 1)
Link Count	1	134 (x 1)
Version Count	1	50
Version Major	0	0
Version Minor	0	0
Version Patch	0	0
Version CVE	0	0
Version Invalid		0
Maintenance Cost		148

MEVN Stack

Ecosystem:	mongodb,express,vue,node	
Attribute	**Weight**	**Count**
Component Count	1	17 (x 1)
Link Count	1	132 (x 1)
Version Count	1	50
Version Major	0	0
Version Minor	0	0
Version Patch	0	0
Version CVE	0	0
Version Invalid		0
Maintenance Cost		149

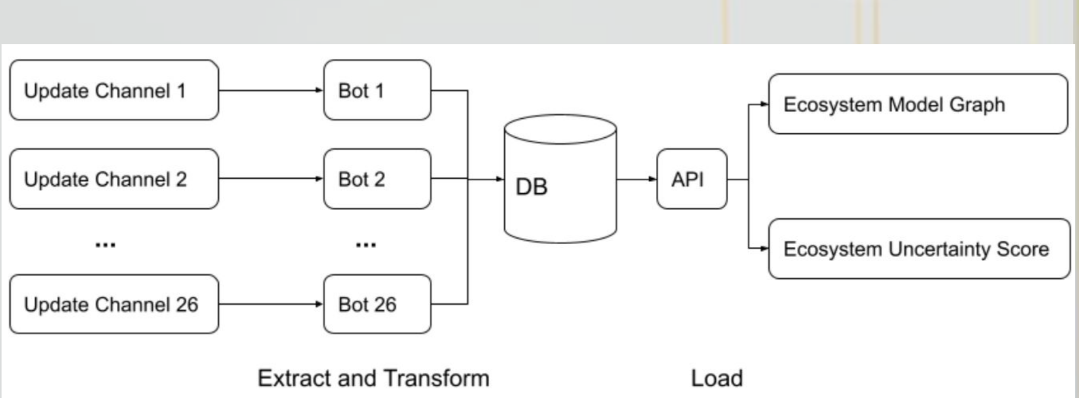
Q3: *What is the maintenance cost of a software stack ecosystem?*
A3: MEAN Stack

3.1. Definition Software Ecosystem

1. $Eco^a = (N^a, E^a)$ is a software ecosystem graph.
2. $N^a = \bigcup c_i^a$ is the set of software component nodes in Eco^a.
3. $E^a = \bigcup (c_i^a, c_j^a)$, where $TAG(c_i^a) \cap TAG(c_j^a) \neq \emptyset$ is the set of edges in Eco^a.
4. $TAG(c_i^a) = \bigcup tag_{ij}^a$ is the set of semantic description tags in c_i^a.

3.2. Definition Maintenance Cost

4. $Cost(c_i^a) = w(component) + w(version) + w(links) * |c_i^a| + \sum_{versiontype} w(versiontype) * |versiontype(c_i^a)|$
5. $Cost(Eco^a) = \sum_i Cost(c_i^a)$



Discussion:

- Maintaining a software ecosystem can be costly with frequent updates
- Component and graph based overview simplifies documenting of updates
- Weighted maintenance cost function adapts to different Industry 4.0 use cases
- Use cases show how software release notes metadata can be used to answer questions
- Picking the MEAN stack means a smaller and less connected ecosystem, implying less update efforts
- Open source components may allow fast reuse, but frequent updates may add significant maintenance cost
- Reviewing maintenance cost of open source software may reduce overall ecosystem cost
- Limitations include partial leverage of release notes data, semantic versioning required, relative cost values, and generic edges instead of only dependency edges

Conclusion:

Impact evaluating the maintenance cost of software updates is an important and challenging task for many Industry 4.0 domains that are often embedded in software ecosystems. The hypothesis of this research is that software release notes metadata can be leveraged to support with reasoning this task. Towards this objective, the research proposed:

- Model a software ecosystem graph and calculate its maintenance cost using the data
- Prototype and evaluate the model in three Industry 4.0 use cases

Future work includes validating the model, prototype, and use cases in a real Industry 4.0 setting.

References:

[1] Berhe, S., Maynard, M., Khomh, F. (2020). *Software Release Patterns - When is a good time to update a software component?* In Proceedings of the 3rd International Conference on Emerging Data and Industry 4.0
[2] Berhe, S., Maynard, M., Khomh, F. (2023). *Maintenance Cost of Software Ecosystem Updates.* In Proceedings of the 6th International Conference on Emerging Data and Industry 4.0

HCSS
THE TWENTY-THIRD ANNUAL
HIGH CONFIDENCE SOFTWARE AND SYSTEMS CONFERENCE
MAY 8 -10, 2023 | http://cps-hcss.org