Connecting SBOMs with OSS Project Health to Better Understand Dependencies

FOSDEM 2025, Brussels, Belgium - Feb. 2, 2025

Georg Link, PhD

Thesis: Tracking OSS Project Health is Proactive Risk Management

OSS Project Health

The potential for an OSS project to continue releasing quality software.

Risk Management

Evaluation of risks and procedures to avoid or minimize their impact.

Community Activity Indicates Health

Healthy ?

Abandon ?

The community activity today is a leading indicator for the software project's future.

Meet Georg Link

Open Source Strategist

- Business focus
- 20+ years in open source
- Co-Founder of CHAOSS
- Community Builder

“My mission is to improve the health and sustainability of open source.”

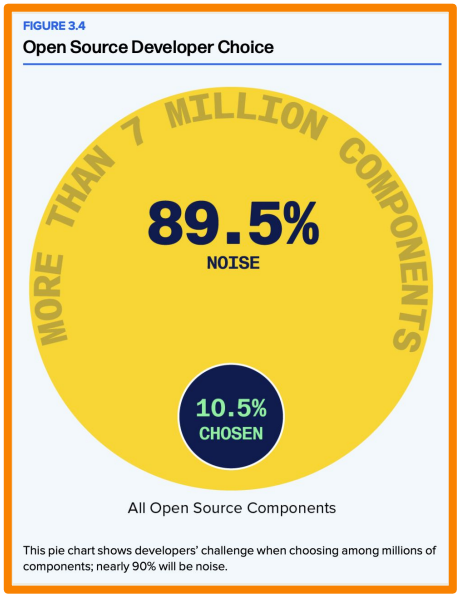
Agenda

- Being perplexed by OSS in Software Supply Chain
- Articulating with SBOMs
- Unlocking new analyses with SBOMs
- Anticipating risk with project health
- Scaling understanding with SBOMs + Project Health

Being perplexed

by OSS in Software Supply Chain

Dilemma: Choice and Maintenance

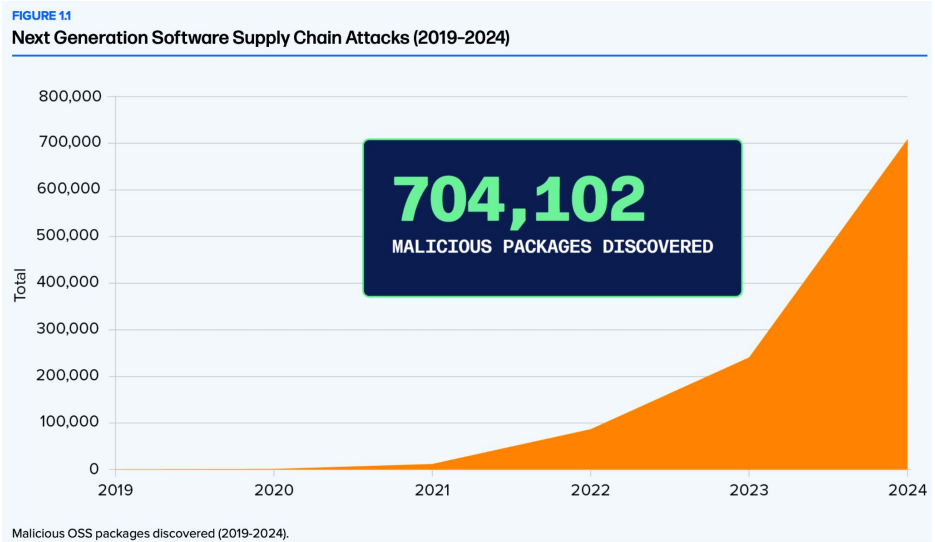


https://www.sonatype.com/state-of-the-software-supply-chain

Sonatype 9th State of Software Supply Chain report:

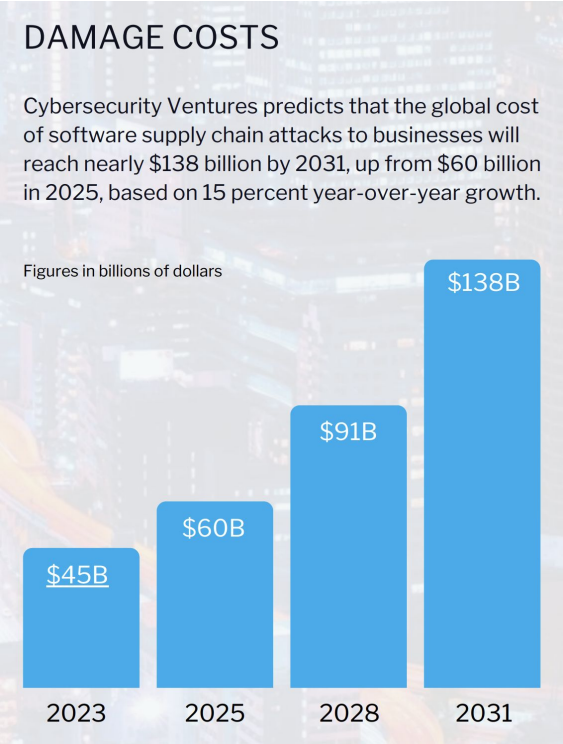
"Consider this: last year, we revealed that a staggering **85% of projects in Maven Central** — the largest public repository for Java open source components — **are inactive.** In other words, developers are faced with a perplexing array of choices, with only a fraction of them leading to active, well-maintained projects."

Software Supply Chain Attacks are on the Rise

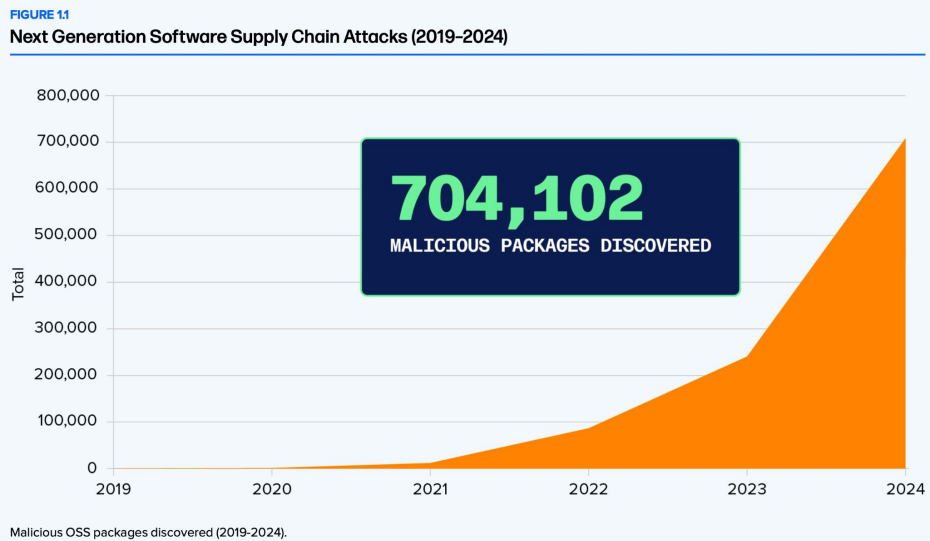


https://www.sonatype.com/state-of-the-software-supply-chain

Software Supply Chain Attacks are on the Rise

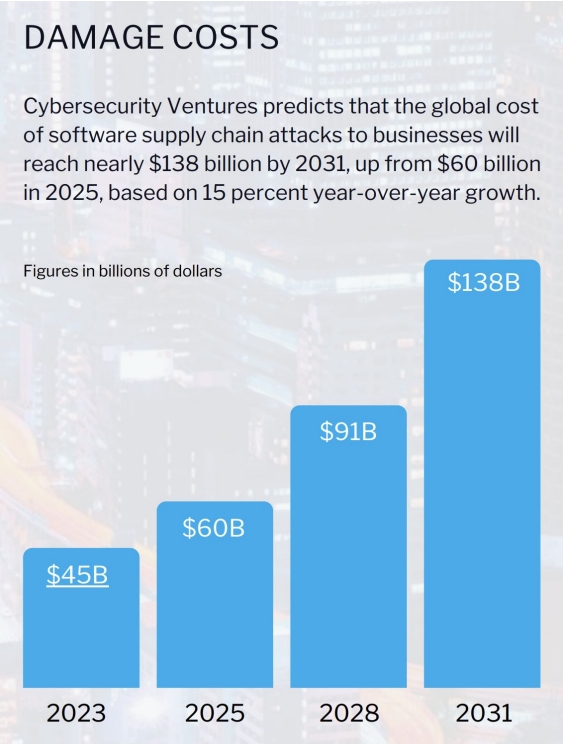


https://go.snyk.io/2023-supply-chain-attacks-report.html



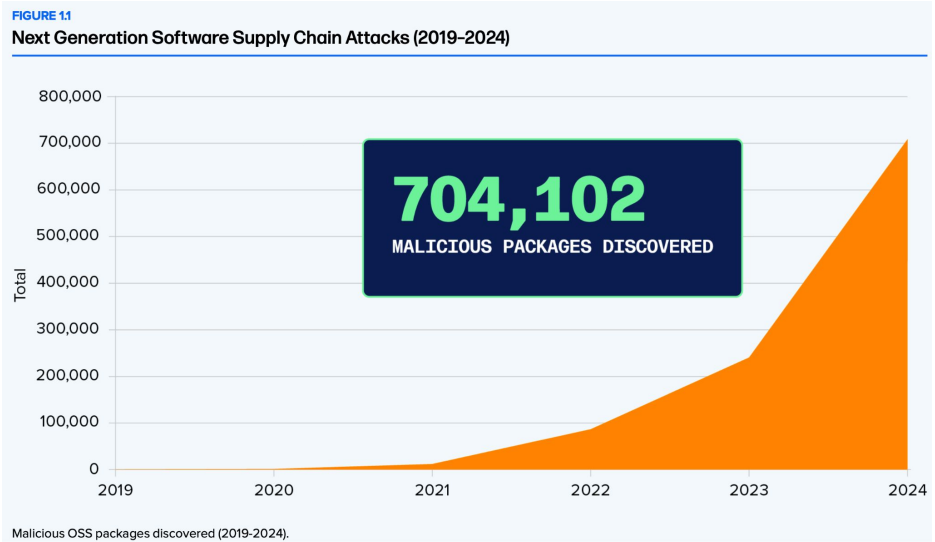
https://www.sonatype.com/state-of-the-software-supply-chain

Software Supply Chain Attacks are on the Rise



https://go.snyk.io/2023-supply-chain-attacks-report.html

Gartner predicts that by 2025, **45 percent** of organizations worldwide will have experienced attacks on their software supply chains, a three-fold increase from 2021.
(Gartner, Mar 7, 2022)



https://www.sonatype.com/state-of-the-software-supply-chain

Research Found the SolarWinds Cyber Attack Cost Affected Companies in Key Sectors 11% of Total Annual Revenue on Average

Results indicate cyber-related information sharing is increasing, signaling a positive response to national-and industry-level calls to action

By Business Wire

Jun 28, 2021

Articulating

with SBOMs

Software ages like Milk, not Wine

Trust: Expiration Label and Source Information

Latest driver: Cyber Resilience Act (CRA)

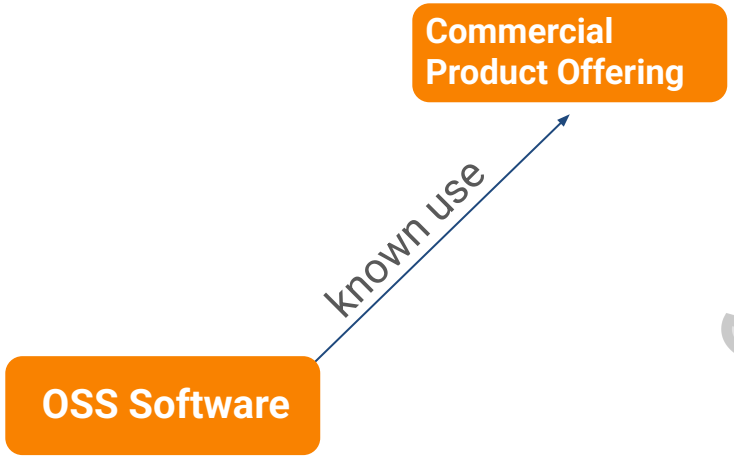
(34) When integrating components sourced from third parties in products with digital elements during the design and development phase, manufacturers should, in order to ensure that the products are designed, developed and produced in accordance with the

When integrating components sourced from third parties … manufacturers should, … exercise due diligence with regard to those components, including free and open-source software components …

the market and for the support period, apply to products with digital elements in their entirety, including to all integrated components. Where, in the exercise of due diligence, the manufacturer of the product with digital elements identifies a vulnerability in a component, including in a free and open-source component, it should inform the person or entity manufacturing or maintaining the component, address and remediate the vulnerability, and, where applicable, provide the person or entity with the applied security fix.

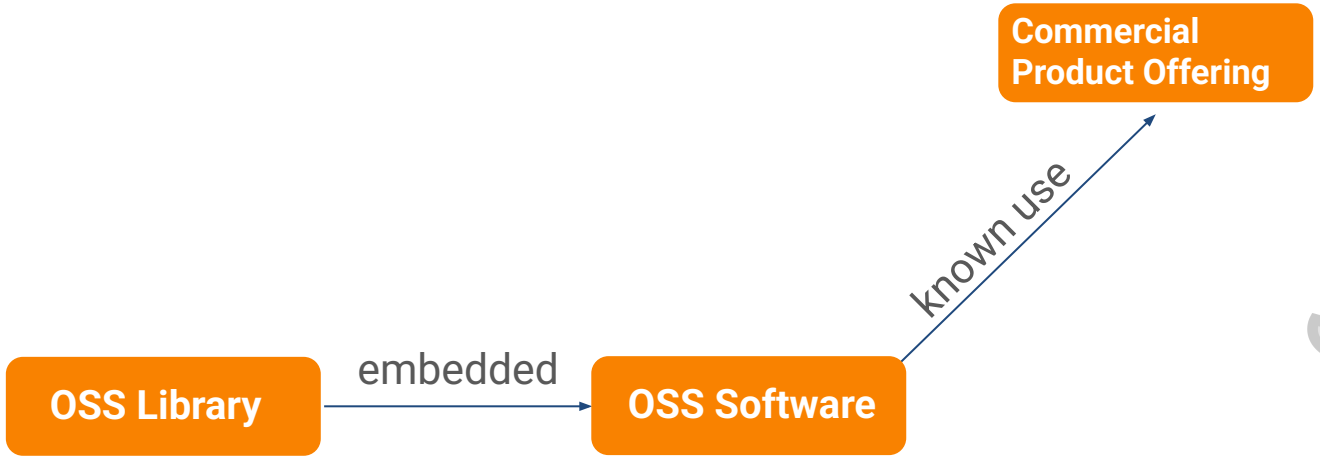
Context: Unmanaged OSS Use → Unknown Risk

- Developers use open source software
- 70% - 95% of software includes open source



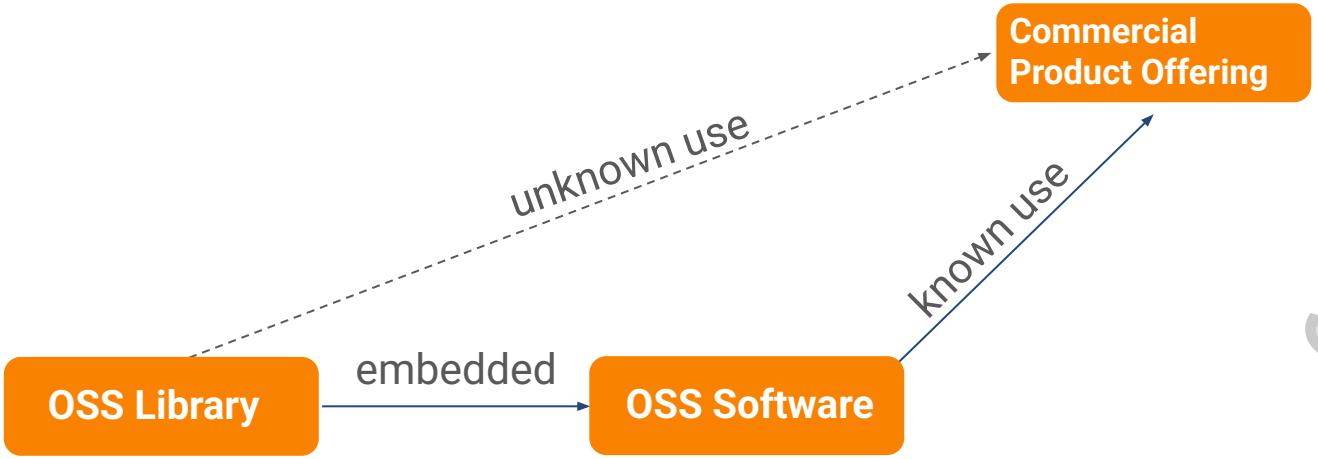
Context: Unmanaged OSS Use → Unknown Risk

- Developers use open source software
- 70% - 95% of software includes open source
- Unmanaged OSS use → unknown dependencies



Context: Unmanaged OSS Use → Unknown Risk

- Developers use open source software
- 70% - 95% of software includes open source
- Unmanaged OSS use → unknown dependencies → **unknown risk**



Articulated: Software Bill of Material (SBOM)

An SBOM is a nested inventory, a list of ingredients that make up software components.

Unlocking

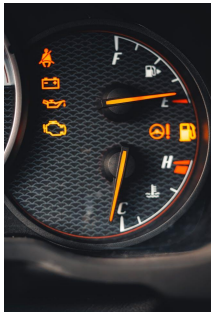
new analyses with SBOMs

Imagine a Car

State Today - relying on instruments:

- Flat Tires
- No Gas
- Warning Symbols
- Error Codes

→ You know how to fix today's situation



Future Support - leveraging origin information:

- Availability of Replacement Parts
- Skilled Workers to Repair
- Network of Repair shops
- Life Expectancy of Car

→ Unsupported Oldtimer vs. Supported Modern Car



Imagine a Car - a metaphor for software

State Today - relying on instruments:

- Flat Tires
- No
- Wa
- Err

→ You k uation

Future Support - leveraging origin information:

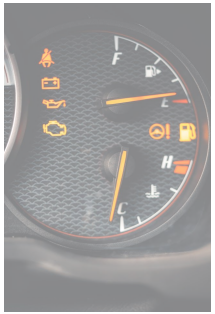
- Availability of Replacement Parts
- Skill
- Netv
- Life

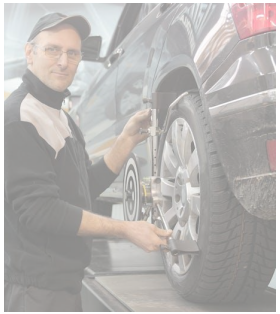
→ Unsupp ted Modern Car

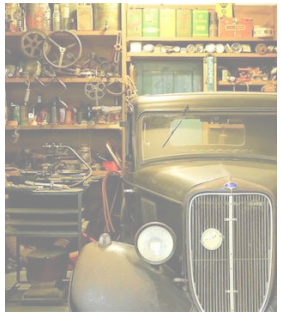
Licenses

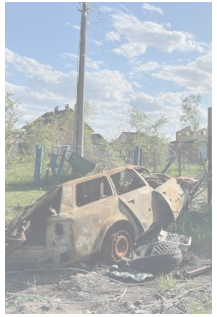
Security

Under-maintained projects









Problem: Trust in OSS Libraries to Manage Risk

- Licenses
- Security
- Under-maintained projects

Problem: Trust in OSS Libraries to Manage Risk

Licenses

- License scanners

Security

- Software Composition Analysis (SCA)
- Vulnerability Databases

Under-maintained projects

https://www.sonatype.com/state-of-the-software-supply-chain

Problem: Trust in OSS Libraries to Manage Risk

Licenses

- License scanners

Security

- Software Composition Analysis (SCA)
- Vulnerability Databases

Under-maintained projects

- Community Health Metrics (CHAOSS)

Missing: Forward looking risk

Anticipating

risk with project health

Thesis: Tracking OSS Project Health is Proactive Risk Management

OSS Project Health

The potential for an OSS project to continue releasing quality software.

Risk Management

Evaluation of risks and procedures to avoid or minimize their impact.

Community Activity Indicates Health

Healthy ?

Abandon?

The community activity today is a leading indicator for the software project's future.

Operationalized Risk: “Under-maintained Projects”

CHAOSS metrics that scale, include 7 metrics:

Community cannot handle workload
- Backlog Management Index
- Review Efficiency Index

Community does not address work quickly
- Median Lead Time for Issues
- Median Lead Time for Pull Requests

Community lacks sufficient talent
- Retention Rate
- Growth of Active Contributors
- Contributor Absence Factor (aka Bus or Pony Factor)

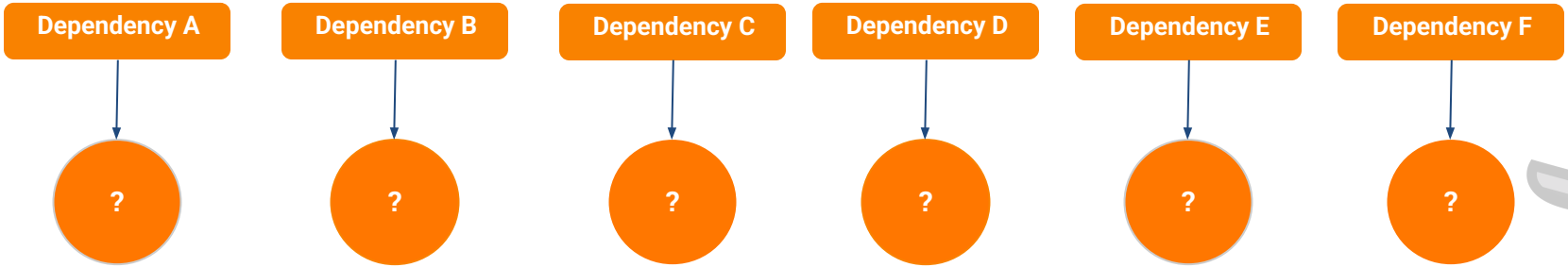
Scaling

understanding with SBOMs + Project Health

BITERGIA | 2024

What's the project health of our dependencies?

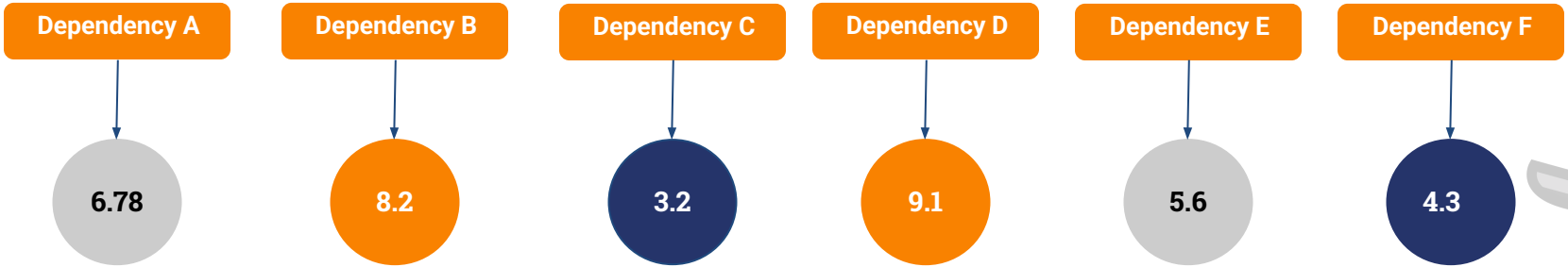
The SBOM gives us the list of dependencies, that we can now investigate.



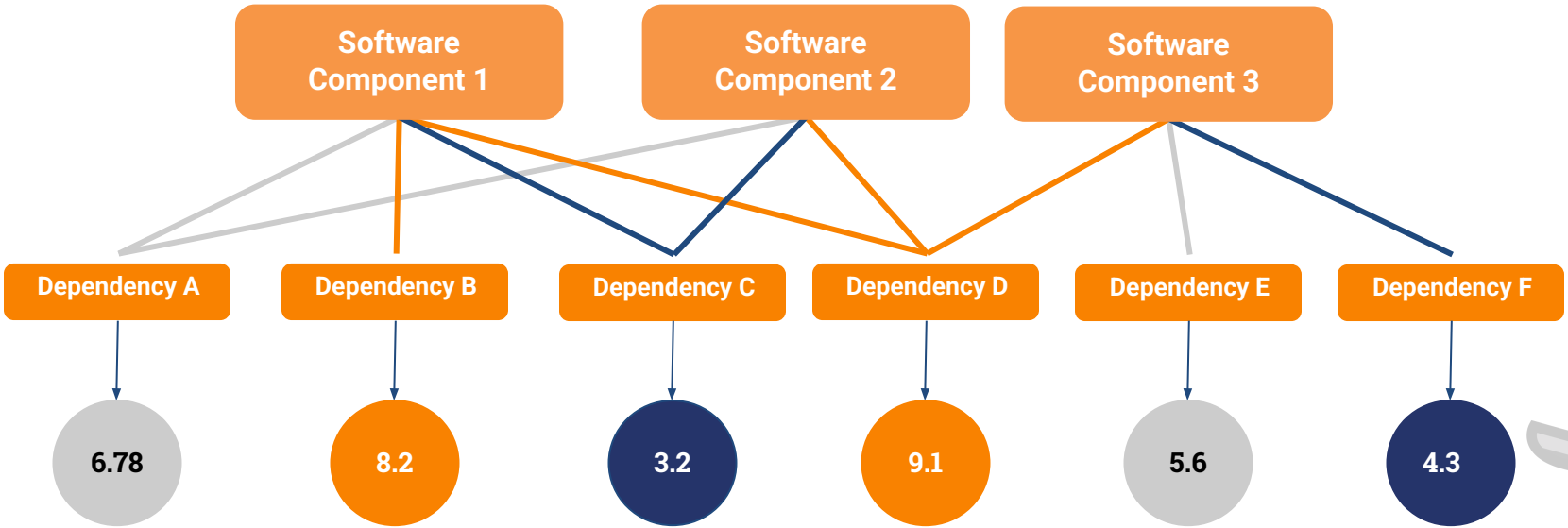
A single Risk Score per OSS library

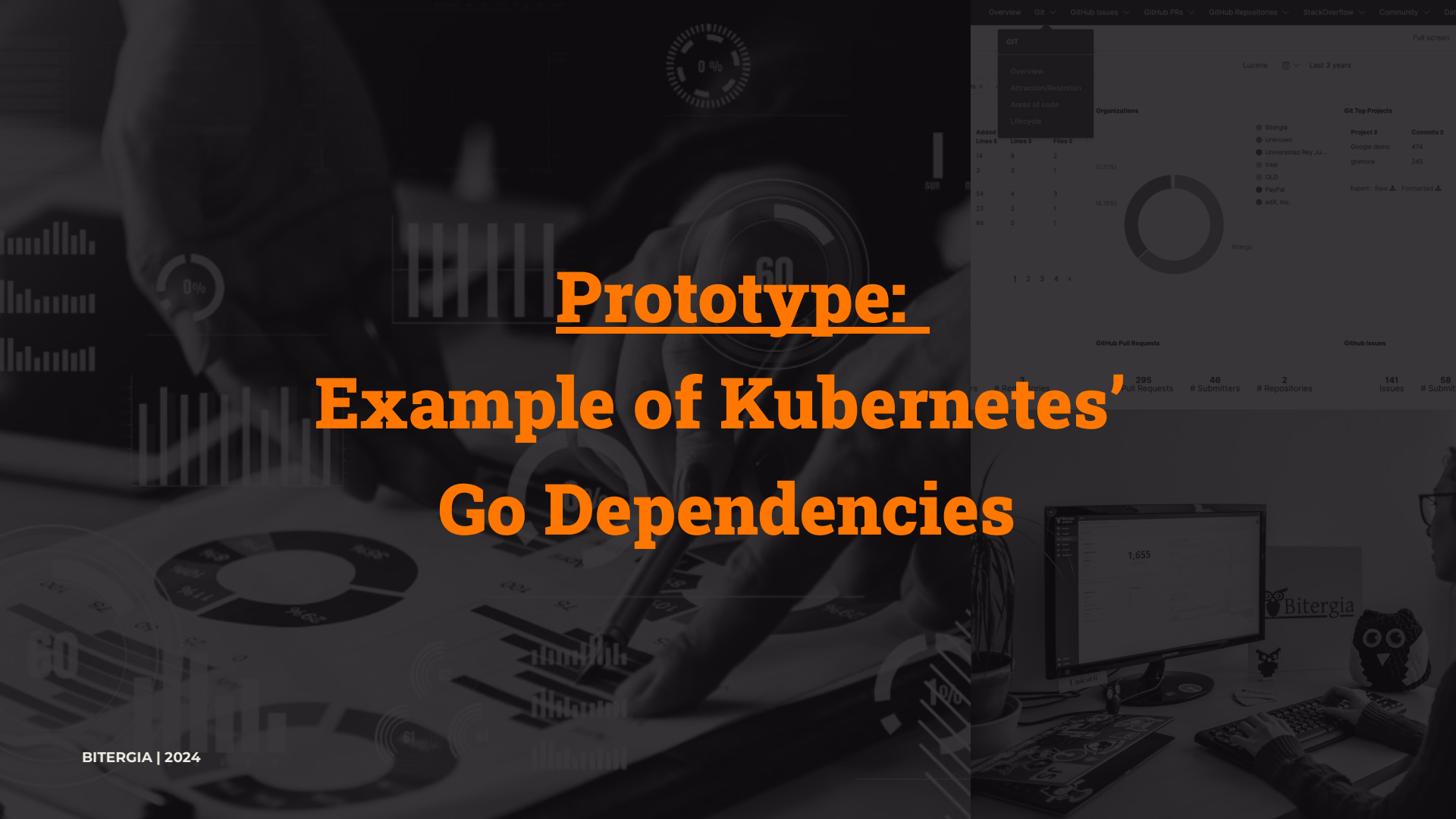
7 metrics, normalized, and

combined into one score for each dependency



Risk Model - Aggregate By Component





Prototype:
Example of Kubernetes' Go Dependencies

project: Kubernetes - Golang Deps × | Add filter

Risk Model Overview Dashboard

Check the Risk Model Help Dashboard for more information about this analysis.

For more details, pin a filter by origin and visit the Risk Model Dashboard for Individual Projects.

Filters

Team

Select...

Project Category

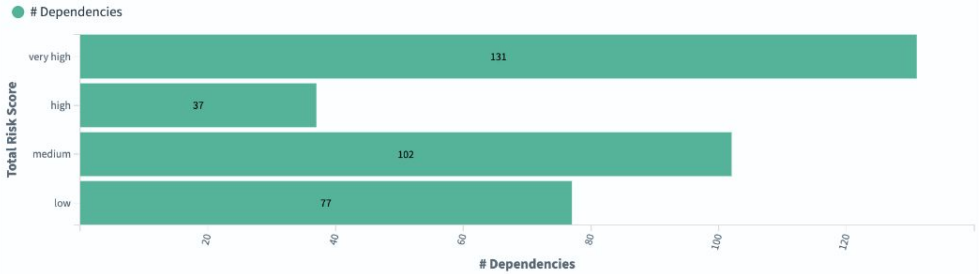
Kubernetes - Golang Deps ×

Overview

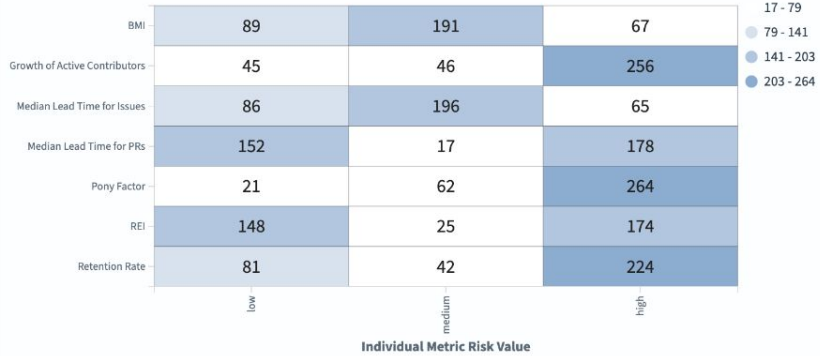
347
Dependencies analyzed

Bitergia
Team

Dependencies by Risk value



Risk Value per Metric, by number of Dependencies



Overall results by dependency repository

Filter...

Repository	Category	Risk Value	Risk Score (over 10)	# "Low risk" metrics	# "Medium risk" metrics	# "High risk" metrics	Last analyzed on
https://github.com/json-iterator/go	Kubernetes - Golang Deps	very high	10	0	0	7	Jun 27, 2024 @ 12:35
https://github.com/spf13/afero	Kubernetes - Golang Deps	very high	10	0	0	7	Jun 27, 2024 @ 12:35
https://github.com/Azure/go-ansiterm	Kubernetes - Golang Deps	very high	9.29	0	1	6	Jun 27, 2024 @ 12:35
https://github.com/ThalesIgnite/crypto11	Kubernetes - Golang Deps	very high	9.29	0	1	6	Jun 27, 2024 @ 12:35
https://github.com/coreos/go-semver	Kubernetes - Golang Deps	very high	9.29	0	1	6	Jun 27, 2024 @ 12:35
https://github.com/curioswitch/go-reassign	Kubernetes - Golang Deps	very high	9.29	0	1	6	Jun 27, 2024 @ 12:35
https://github.com/davecgh/go-spew	Kubernetes - Golang Deps	very high	9.29	0	1	6	Jun 27, 2024 @ 12:35

Export: Raw Formatted

1 2 3 4 5 ... 35

Risk Model Overview Dashboard

Check the Risk Model Help Dashboard for more information about this analysis.

For more details, pin a filter by origin and visit the Pi

Filters

Team

Select...

Project Category

Kubernetes - Golang Deps ×

Bitergia
Team

Dependencies by Risk value

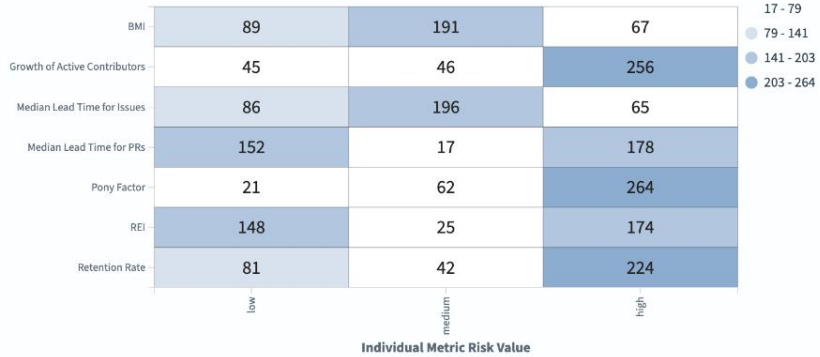
Dependencies

very high 131

Project Category

Kubernetes - Golang Deps ×

Risk Value per Metric, by number of Dependencies



Filter...

Repository	Category	Risk Value	Risk Score (over 10)	# "Low risk" metrics	# "Medium risk" metrics	# "High risk" metrics	Last analyzed on
https://github.com/json-iterator/go	Kubernetes - Golang Deps	very high	10	0	0	7	Jun 27, 2024 @ 12:35
https://github.com/spf13/afero	Kubernetes - Golang Deps	very high	10	0	0	7	Jun 27, 2024 @ 12:35
https://github.com/Azure/go-ansiterm	Kubernetes - Golang Deps	very high	9.29	0	1	6	Jun 27, 2024 @ 12:35
https://github.com/ThalesIgnite/crypto11	Kubernetes - Golang Deps	very high	9.29	0	1	6	Jun 27, 2024 @ 12:35
https://github.com/coreos/go-semver	Kubernetes - Golang Deps	very high	9.29	0	1	6	Jun 27, 2024 @ 12:35
https://github.com/curioswitch/go-reassign	Kubernetes - Golang Deps	very high	9.29	0	1	6	Jun 27, 2024 @ 12:35
https://github.com/davecgh/go-spew	Kubernetes - Golang Deps	very high	9.29	0	1	6	Jun 27, 2024 @ 12:35

Export: Raw Formatted

1 2 3 4 5 ... 36

Bitergia Analytics

Dashboards | Demo - Risk Model Overview Dashboard

Full screen | Share | Clone | Edit

project: Kubernetes - Golang Deps × Add filter

Risk Model Overview Dashboard

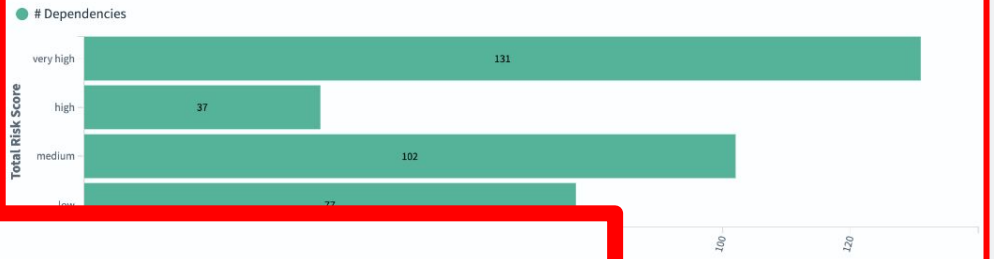
Check the Risk Model Help Dashboard for more information about this analysis.

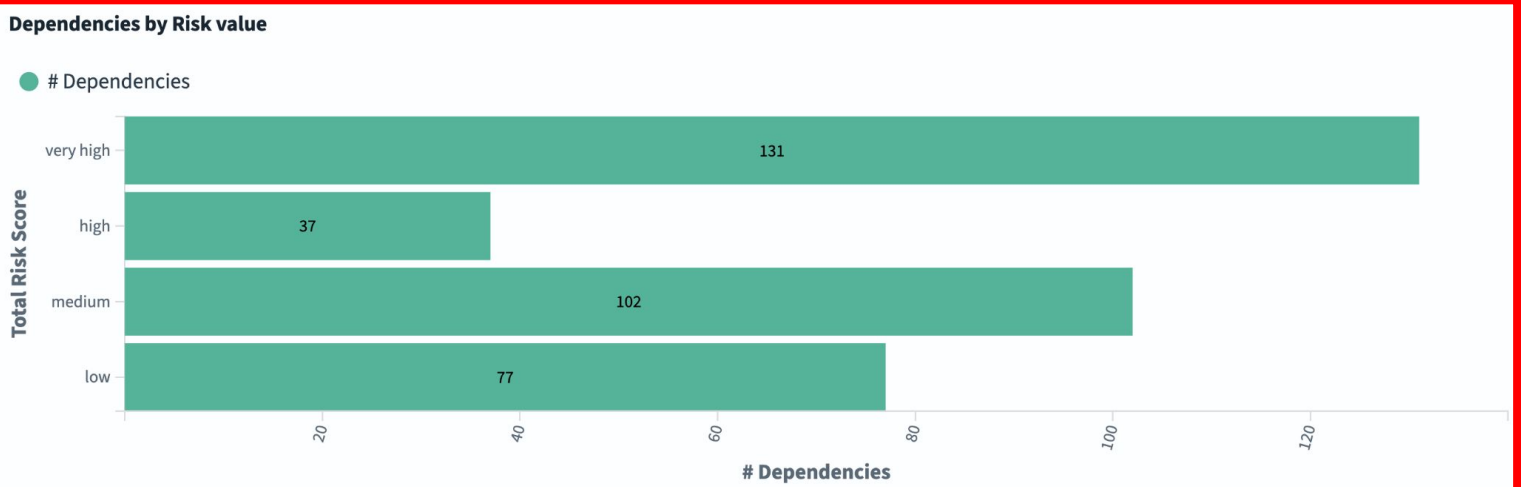
For more details, pin a filter by origin and visit the Risk Model Dashboard for Individual Projects.

Filters

Overview

Dependencies by Risk value





	low	medium	high
REI	148	25	174
Retention Rate	81	42	224

Individual Metric Risk Value

# "Medium risk" metrics	# "High risk" metrics	Last analyzed on
0	7	Jun 27, 2024 @ 12:35
0	7	Jun 27, 2024 @ 12:35
1	6	Jun 27, 2024 @ 12:35
1	6	Jun 27, 2024 @ 12:35
1	6	Jun 27, 2024 @ 12:35
	6	Jun 27, 2024 @ 12:35
1	6	Jun 27, 2024 @ 12:35

https://github.com/davecgh/go-spew | Kubernetes - Golang Deps | very high | 9.29 | 0

Export: Raw Formatted

1 2 3 4 5 ... 37

Bitergia Analytics

Dashboards | Demo - Risk Model Overview Dashboard

Full screen | Share | Clone | Edit

project: Kubernetes - Golang Deps × Add filter

Risk Model Overview Dashboard

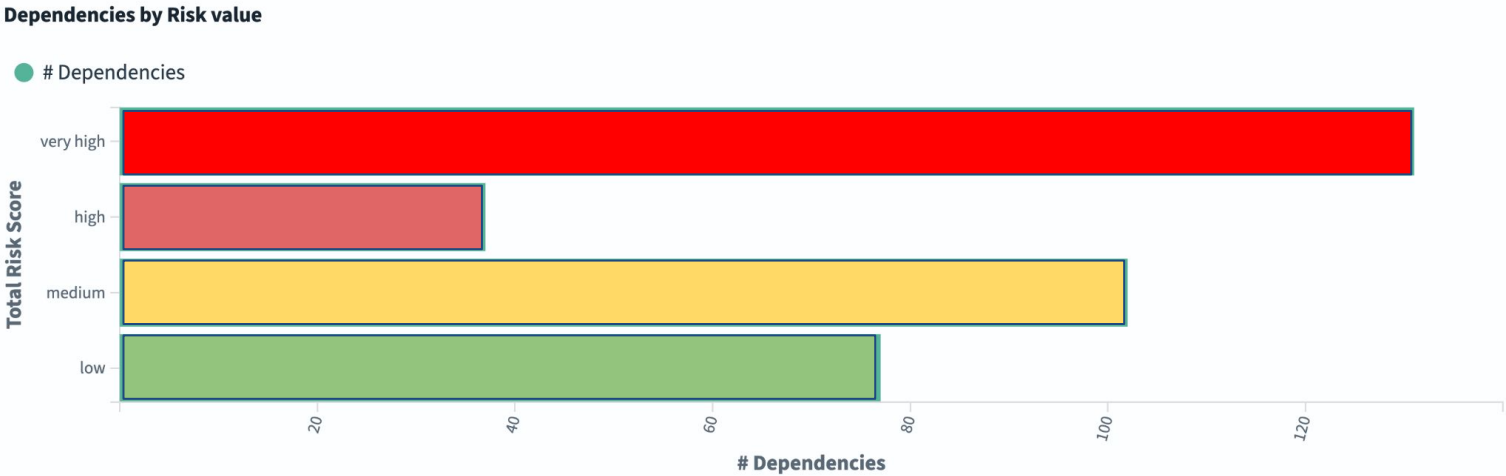
Check the Risk Model Help Dashboard for more information about this analysis.

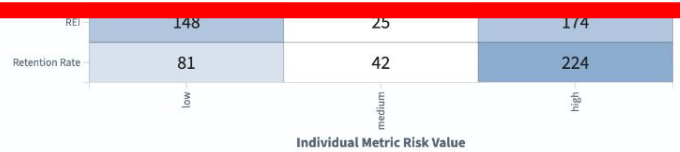
For more details, pin a filter by origin and visit the Risk Model Dashboard for Individual Projects.

Filters

Overview

Dependencies by Risk value





# "Medium risk" metrics	# "High risk" metrics	Last analyzed on
0	7	Jun 27, 2024 @ 12:35
0	7	Jun 27, 2024 @ 12:35
1	6	Jun 27, 2024 @ 12:35
1	6	Jun 27, 2024 @ 12:35
1	6	Jun 27, 2024 @ 12:35
	6	Jun 27, 2024 @ 12:35
1	6	Jun 27, 2024 @ 12:35

https://github.com/davecgh/go-spew | Kubernetes - Golang Deps | very high | 9.29 | 0

Export: Raw Formatted

1 2 3 4 5 ... 38

Bitergia Analytics

Dashboards

project: Kubernetes - Gola

Risk Model Overvi

Check the Risk Model He

For more deta s, pin a f

Filters

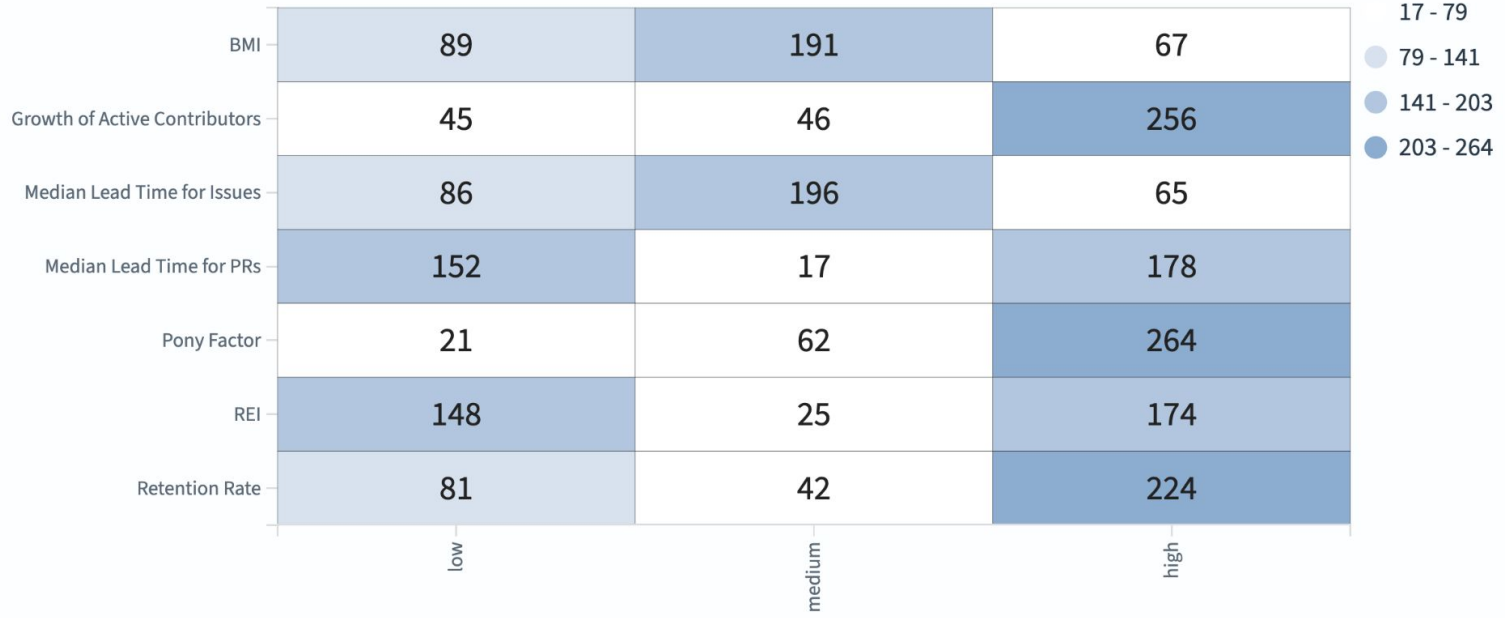
Team

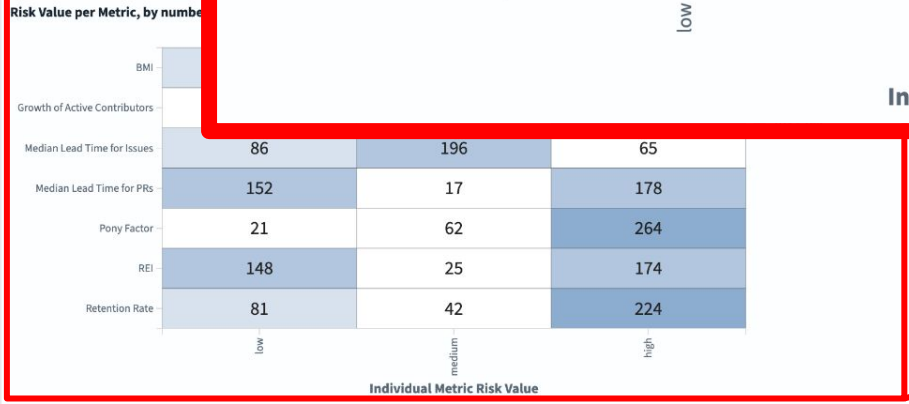
Select.

Project Category

Ku ernetes - Golang Deps ×

Risk Value per Metric, by number of Dependencies





	Deps						12:35
https://github.com/coreos/go-semver	Kubernetes - Golang Deps	very high	9.29	0	1	6	Jun 27, 2024 @ 12:35
https://github.com/curioswitch/go-reassign	Kubernetes - Golang Deps	very high	9.29	0	1	6	Jun 27, 2024 @ 12:35
https://github.com/davecgh/go-spew	Kubernetes - Golang Deps	very high	9.29	0	1	6	Jun 27, 2024 @ 12:35

Export: Raw Formatted

1 2 3 4 5 ... 39

Risk Model Overview Dashboard

Check the Risk Model Help Dashboard for more information about this analysis.

For more details, pin a filter by origin and visit the Risk Model Dashboard for Individual Projects.

Filters

Team

Select...

Project Category

Kubernetes - Golang Deps ×

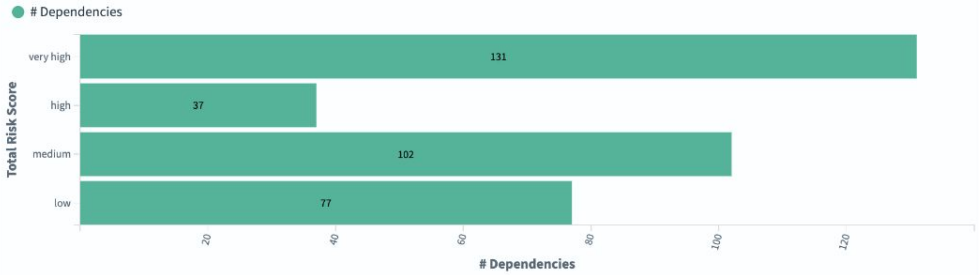
Overview

347

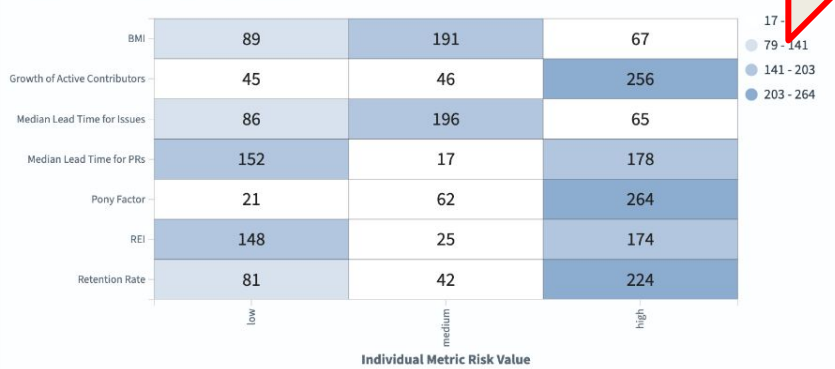
Dependencies analyze

Drill Down

Dependencies by Risk value



Risk Value per Metric, by number of Dependencies



Overall results by dependency repository

Filter...

Repository	Category	Risk Value	Risk Score (over 10)	# "Low risk" metrics	# "Medium risk" metrics	# "High risk" metrics	Last analyzed on
https://github.com/json-iterator/go	Kubernetes - Golang Deps	very high	10	0	0	7	Jun 27, 2024 @ 12:35
https://github.com/spf13/afero	Kubernetes - Golang Deps	very high	10	0	0	7	Jun 27, 2024 @ 12:35
https://github.com/Azure/go-ansiterm	Kubernetes - Golang Deps	very high	9.29	0	1	6	Jun 27, 2024 @ 12:35
https://github.com/ThalesIgnite/crypto11	Kubernetes - Golang Deps	very high	9.29	0	1	6	Jun 27, 2024 @ 12:35
https://github.com/coreos/go-semver	Kubernetes - Golang Deps	very high	9.29	0	1	6	Jun 27, 2024 @ 12:35
https://github.com/curioswitch/go-reassign	Kubernetes - Golang Deps	very high	9.29	0	1	6	Jun 27, 2024 @ 12:35
https://github.com/davecgh/go-spew	Kubernetes - Golang Deps	very high	9.29	0	1	6	Jun 27, 2024 @ 12:35

Export: Raw Formatted

1 2 3 4 5 ... 40

Risk Model Dashboard for Individual Projects

Make sure to filter by a given repository before using this dashboard.

Check the Risk Model Help Dashboard for more information about this analysis.

Go back to the Risk Model Overview Dashboard.

Total Risk Score

github.com/stretchr/testify	medium	4.29	https://github.com/stretchr/t
Package Name	Total Risk	Total Risk Score (over 10)	Package Repository

Select an dependency first to check its risk

Team: Select...

Dependency Pkg Name: Select...

Dependency Repository: Select...

Project Category: Kubernetes - Golang Deps ×

Risk Model: Detailed view - dependency selector table

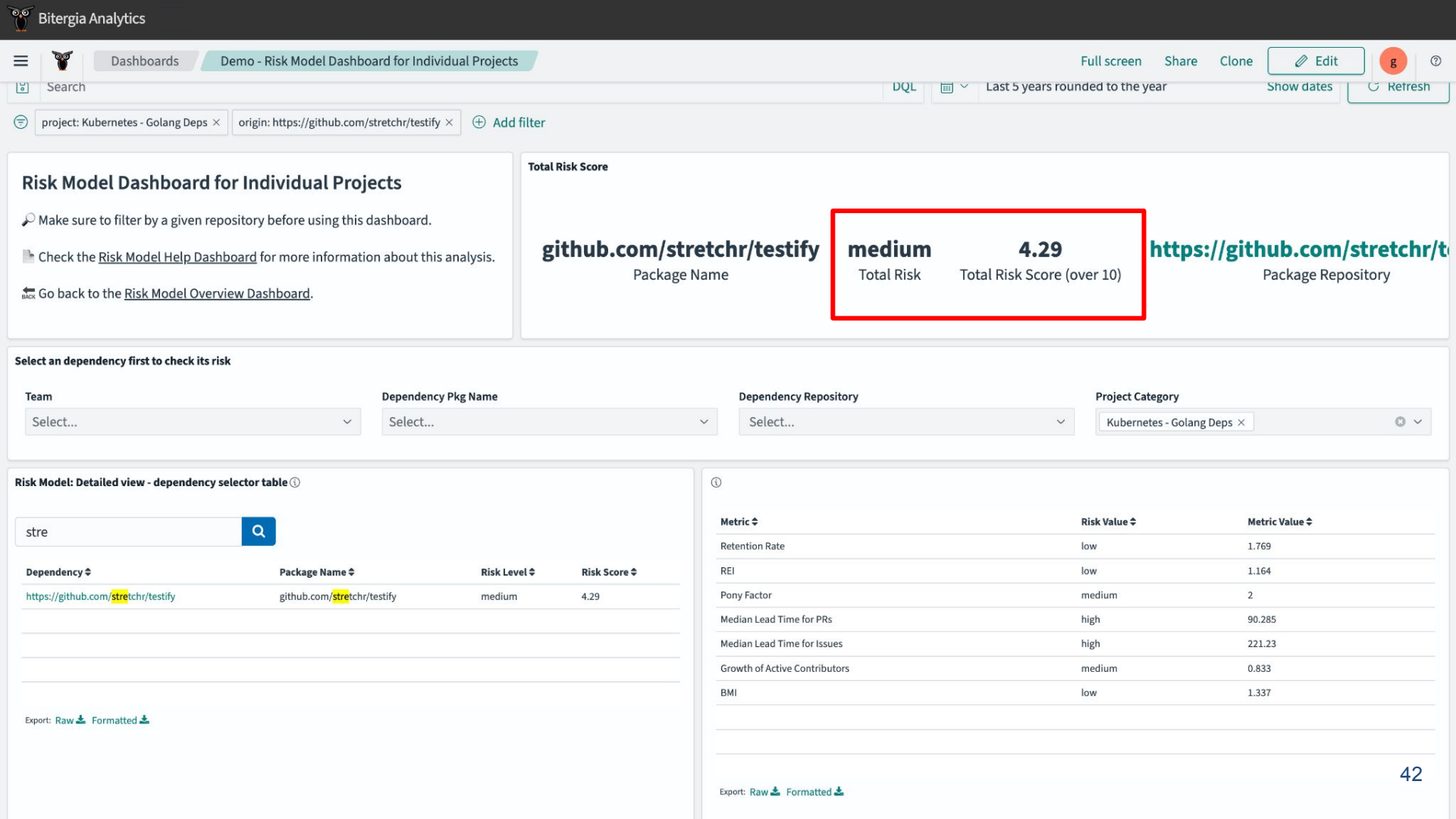
stre

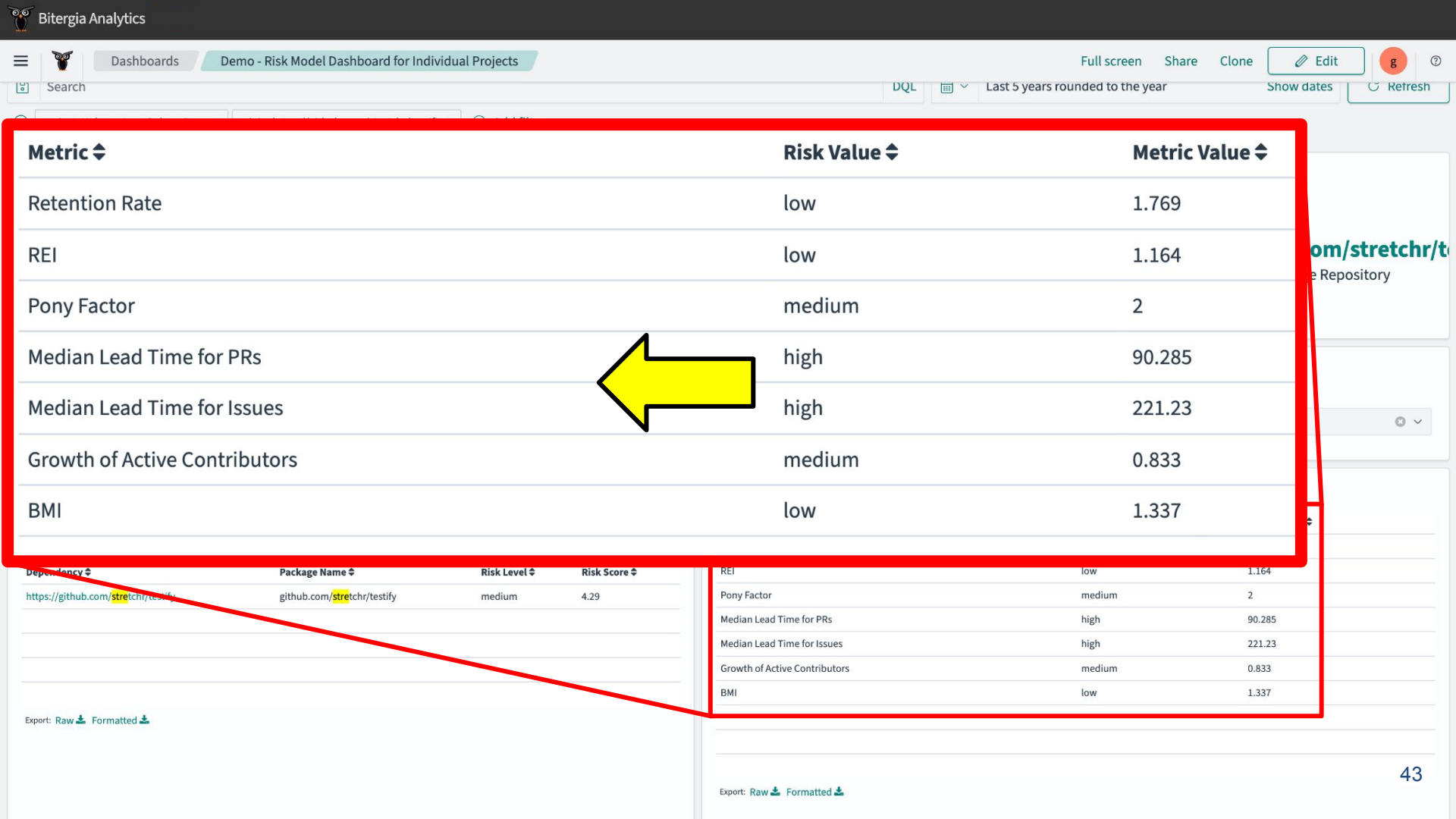
Dependency	Package Name	Risk Level	Risk Score
https://github.com/stretchr/testify	github.com/stretchr/testify	medium	4.29

Export: Raw Formatted

Metric	Risk Value	Metric Value
Retention Rate	low	1.769
REI	low	1.164
Pony Factor	medium	2
Median Lead Time for PRs	high	90.285
Median Lead Time for Issues	high	221.23
Growth of Active Contributors	medium	0.833
BMI	low	1.337

Export: Raw Formatted


Bitergia Analytics
Dashboards
Demo - Risk Model Dashboard for Individual Projects
Full screen
Share
Clone
Edit
Search
DQL
Last 5 years rounded to the year
Show dates
Refresh
project: Kubernetes - Golang Deps
origin: https://github.com/stretchr/testify
Add filter
Risk Model Dashboard for Individual Projects
Make sure to filter by a given repository before using this dashboard.
Check the Risk Model Help Dashboard for more information about this analysis.
Go back to the Risk Model Overview Dashboard.
Total Risk Score
github.com/stretchr/testify
Package Name
medium
Total Risk
4.29
Total Risk Score (over 10)
https://github.com/stretchr/t
Package Repository
Select an dependency first to check its risk
Team
Select...
Dependency Pkg Name
Select...
Dependency Repository
Select...
Project Category
Kubernetes - Golang Deps
Risk Model: Detailed view - dependency selector table
stre
Dependency
Package Name
Risk Level
Risk Score
https://github.com/stretchr/testify
github.com/stretchr/testify
medium
4.29
Export: Raw Formatted
Metric
Risk Value
Metric Value
Retention Rate
low
1.769
REI
low
1.164
Pony Factor
medium
2
Median Lead Time for PRs
high
90.285
Median Lead Time for Issues
high
221.23
Growth of Active Contributors
medium
0.833
BMI
low
1.337
Export: Raw Formatted
42


Bitergia Analytics
Dashboards
Demo - Risk Model Dashboard for Individual Projects
Full screen
Share
Clone
Edit
Search
DQL
Last 5 years rounded to the year
Show dates
Refresh
Metric
Risk Value
Metric Value
Retention Rate
low
1.769
REI
low
1.164
Pony Factor
medium
2
Median Lead Time for PRs
high
90.285
Median Lead Time for Issues
high
221.23
Growth of Active Contributors
medium
0.833
BMI
low
1.337
Package Name
Risk Level
Risk Score
github.com/stretchr/testify
medium
4.29
Export: Raw Formatted
Repository
43

CHAOSS Expert Guide is in the works

Food for discussion,
additional metrics we could consider:

- licenses
- known vulnerabilities
- origin of developers
- organizational diversity

GrimoireLab: The Open Source Tool

BITERGIA | 2024

Story of GrimoireLab

- 2004 LibreSoft @ University Rey Juan Carlos in Spain
- 2012 Bitergia offers commercial services with Metrics Grimoire
- 2016 GrimoireLab starts, using ElasticSearch for Dashboarding
- 2017 Founding of CHAOSS
- 2024 version 1.0 released

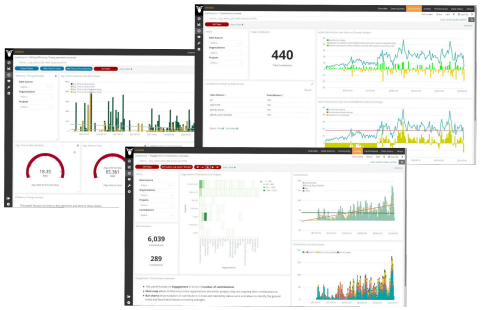
Example: Mozilla Foundation

“[...] holistic view of our contributor ecosystem’s network structure, health and impact [...]”

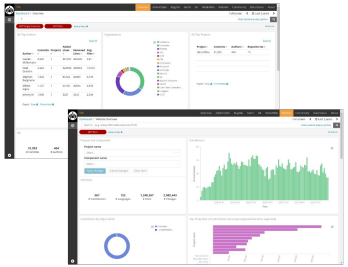
“[...] we’re able to visually describe these distinct contributor communities as well as how they are interconnected [...]”

https://report.mozilla.community/

Platforms built with GrimoireLab

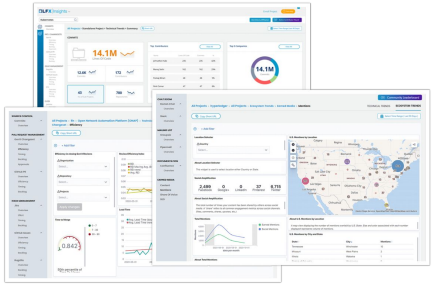


Bitergia Analytics

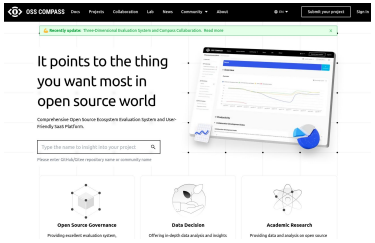


The Document Foundation

Mystic



Linux Foundation Insights



OSS Compass

Collecting traces from OSS communities

Data Collection

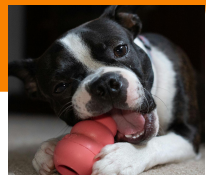
Traces: digital footprints from data source

(biased towards activities that are logged)

Enrichment

Translate data into information

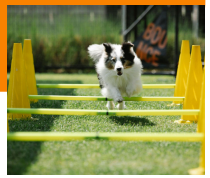
(connect and unify for consistency)



Visualization and Reporting

Gain insights and decide actions

(tell stories and convince)



SortingHat to disambiguate contributors

08/2008 - 07/2011 Bankhaus C. L. Seeliger
10/2011 - 05/2015 TU Braunschweig
08/2015 - 05/2019 University of Nebraska at Omaha
05/2019 - now Bitergia

git

Georg J.P. Link <linkgeorg@gmail.com>
Georg Link <linkgeorg@gmail.com>
Link, Georg <glink@unomaha.edu>
Georg Link <georglink@bitergia.com>

GeorgLink

linkgeorg@gmail.com
glink@unomaha.edu
georglink@bitergia.com

PHABRICATOR

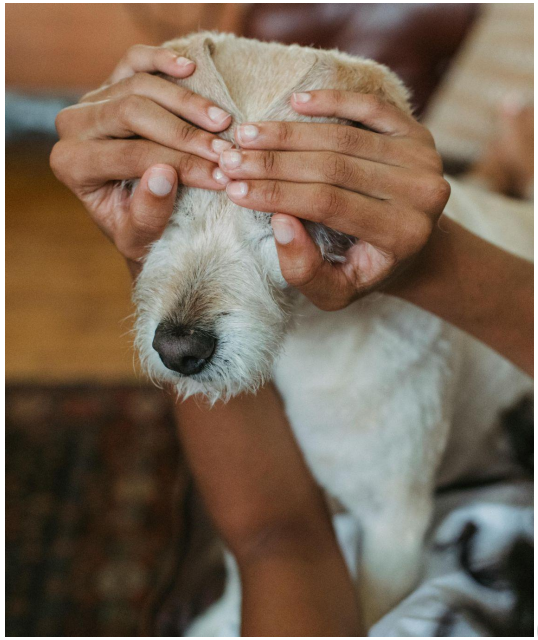
georglink@bitergia.com

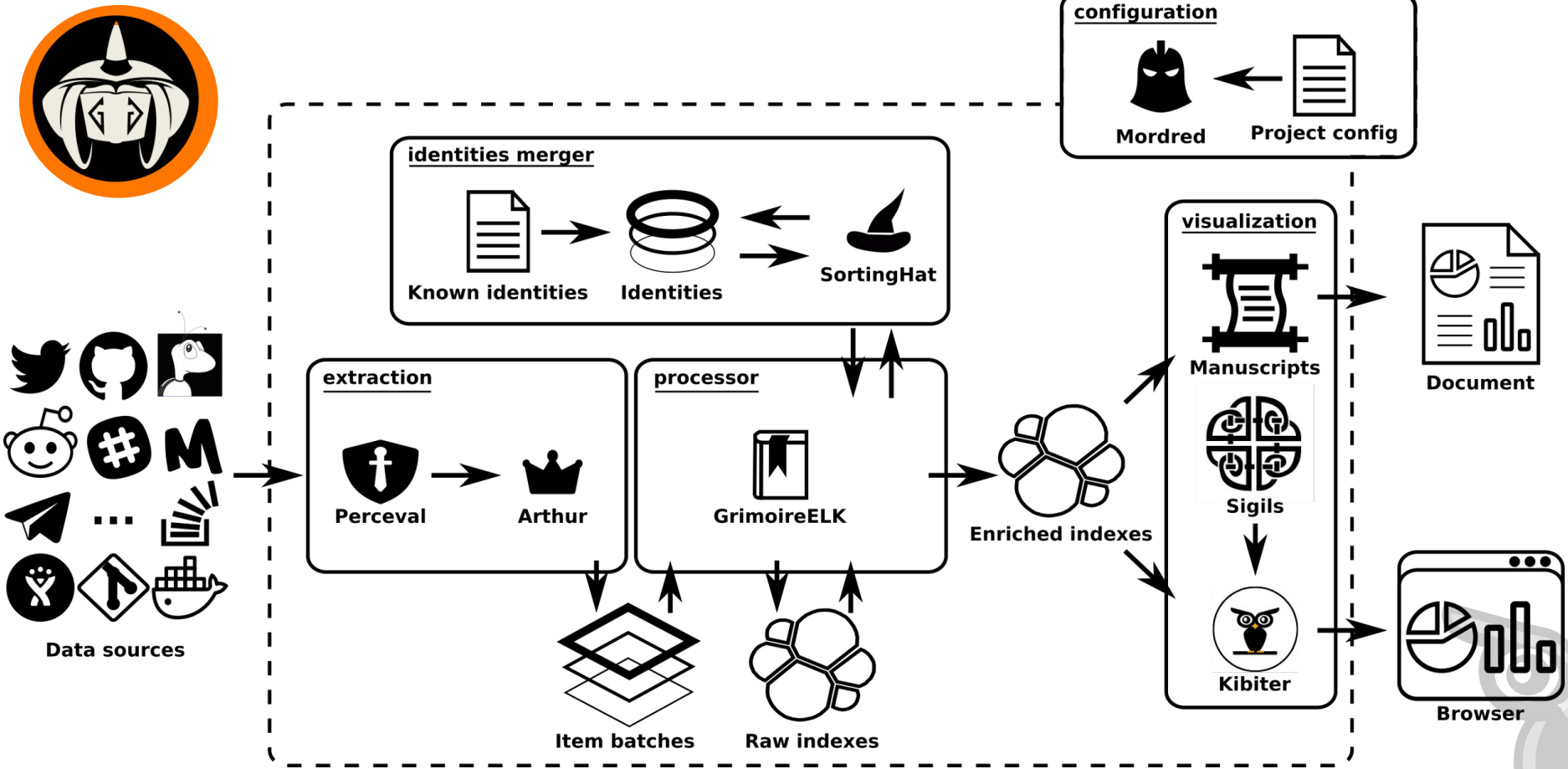
slack

GeorgLink

Excursion: Minding data privacy

- GDPR is gold standard
- Opt-in vs. Opt-out
- Enriching data from data sources
- Offering a “remove my data” feature




configuration
Mordred
Project config
identities merger
Known identities
Identities
SortingHat
visualization
Manuscripts
Document
extraction
Perceval
Arthur
processor
GrimoireELK
Enriched indexes
Sigils
Kibiter
Browser
Data sources
Item batches
Raw indexes

GrimoireLab 2.0 roadmap

- **Maintenance effort:** ⬇️
  Graphical user interface and an API for configuring data collection
- **Scalability and performance:** ⬆️
  Currently, 3,500 high-active repositories require three days of data analysis before the data is ready for the user
- **Integration with other tools:** 🔀
  Support more tools for visualizing and analyzing the data

How to Get Started?

Open Source: GrimoireLab tutorial

- https://chaoss.github.io/grimoirelab-tutorial/

Commercial: Bitergia Risk Radar

- https://bitergia.com/risk-radar/

Call to Action: **Please include the source PURL in the SBOM.**

SBOM + Project Health = Better Understanding

Thank you and please reach out!

Georg Link, PhD
georglink@bitergia.com

FOSDEM 2025
Brussels, Belgium
Feb. 2, 2025