InstaDeep™

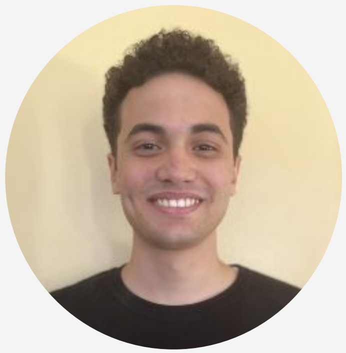
Growing the MARL software ecosystem in JAX

MARL Research Team @ InstaDeep

Presented by:
Ruan de Kock, Sasha Abramowitz, Callum Rhys Tilbury

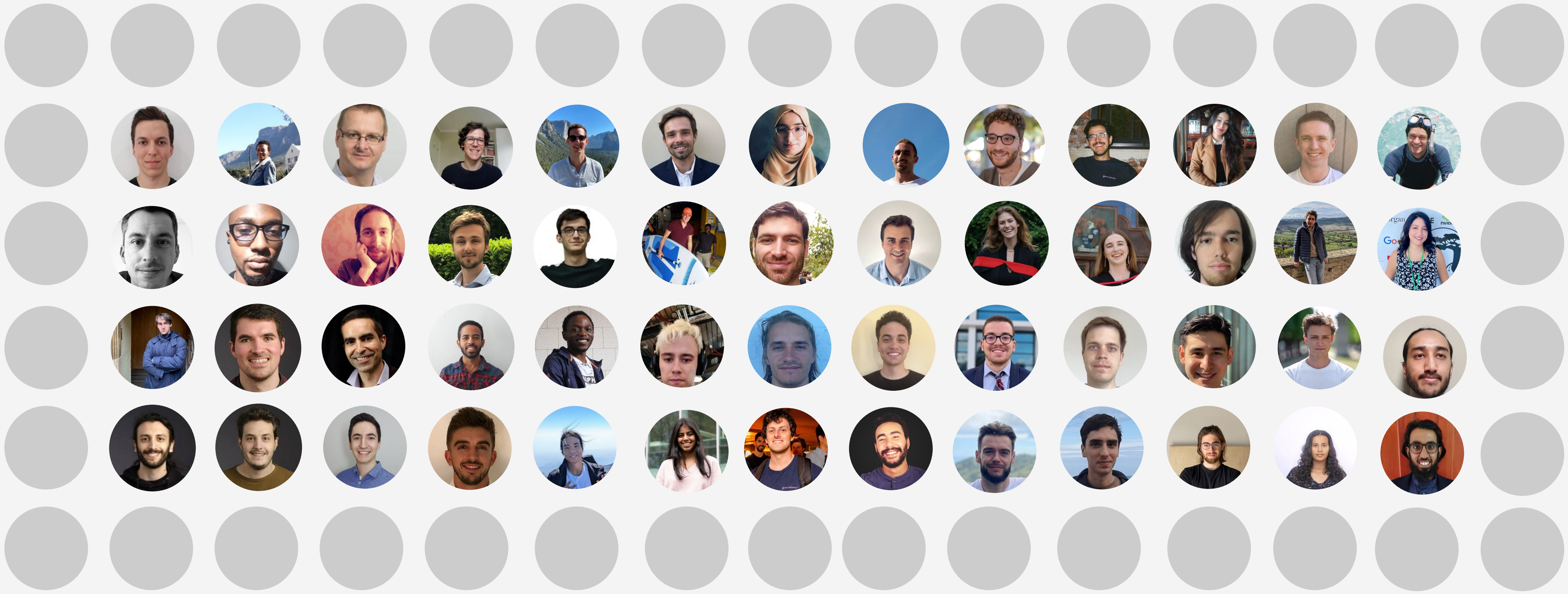
June 2024

Our team...



InstaDeep™

...with many past and public contributors!



Why?

We want to solve hard multi-agent problems

→ Need to push the MARL research frontier

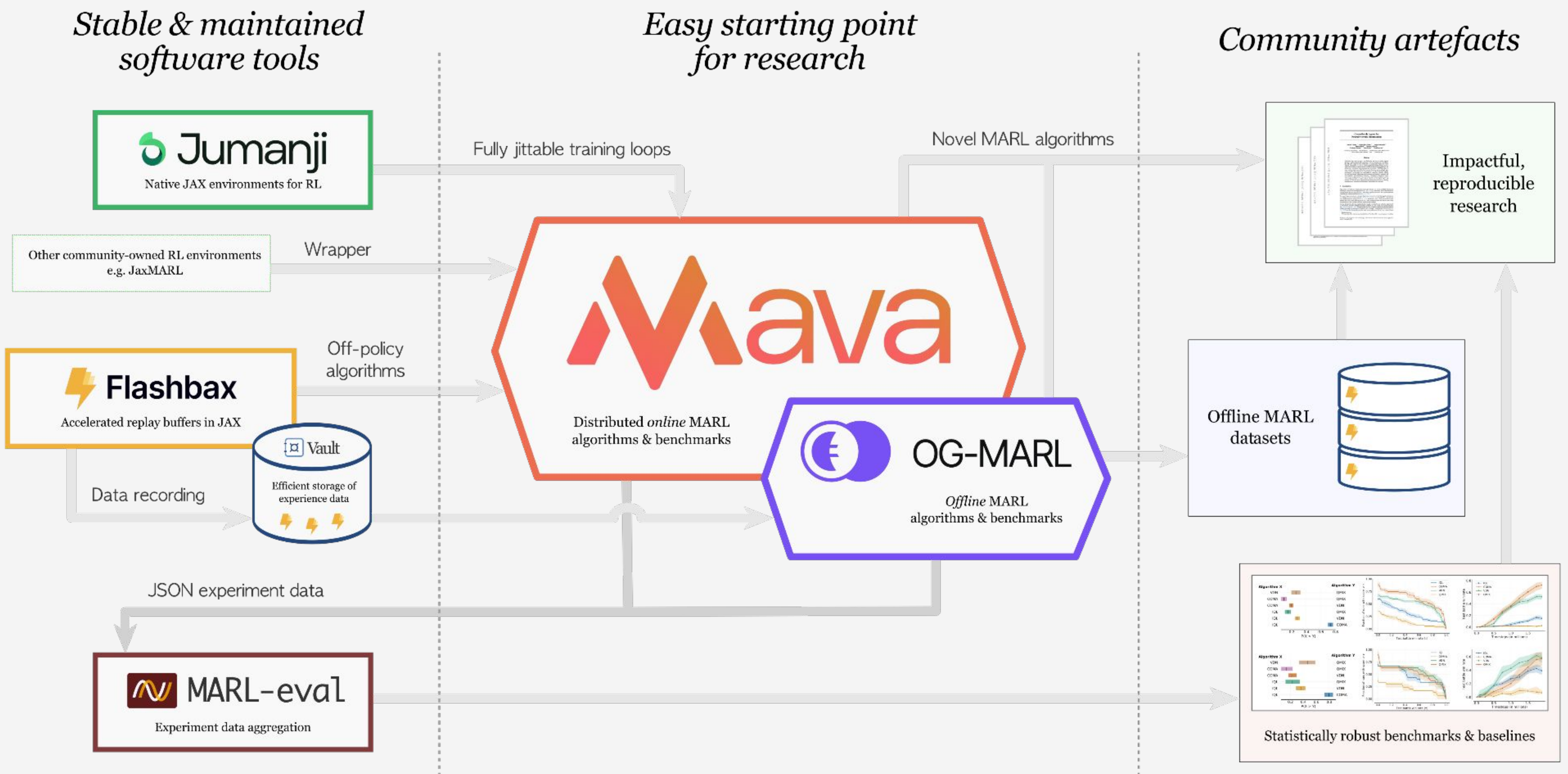
→ Need software that is…

reliable, flexible, scalable,

with stable, maintained tooling,

& robust evaluation methods.

Our MARL software ecosystem

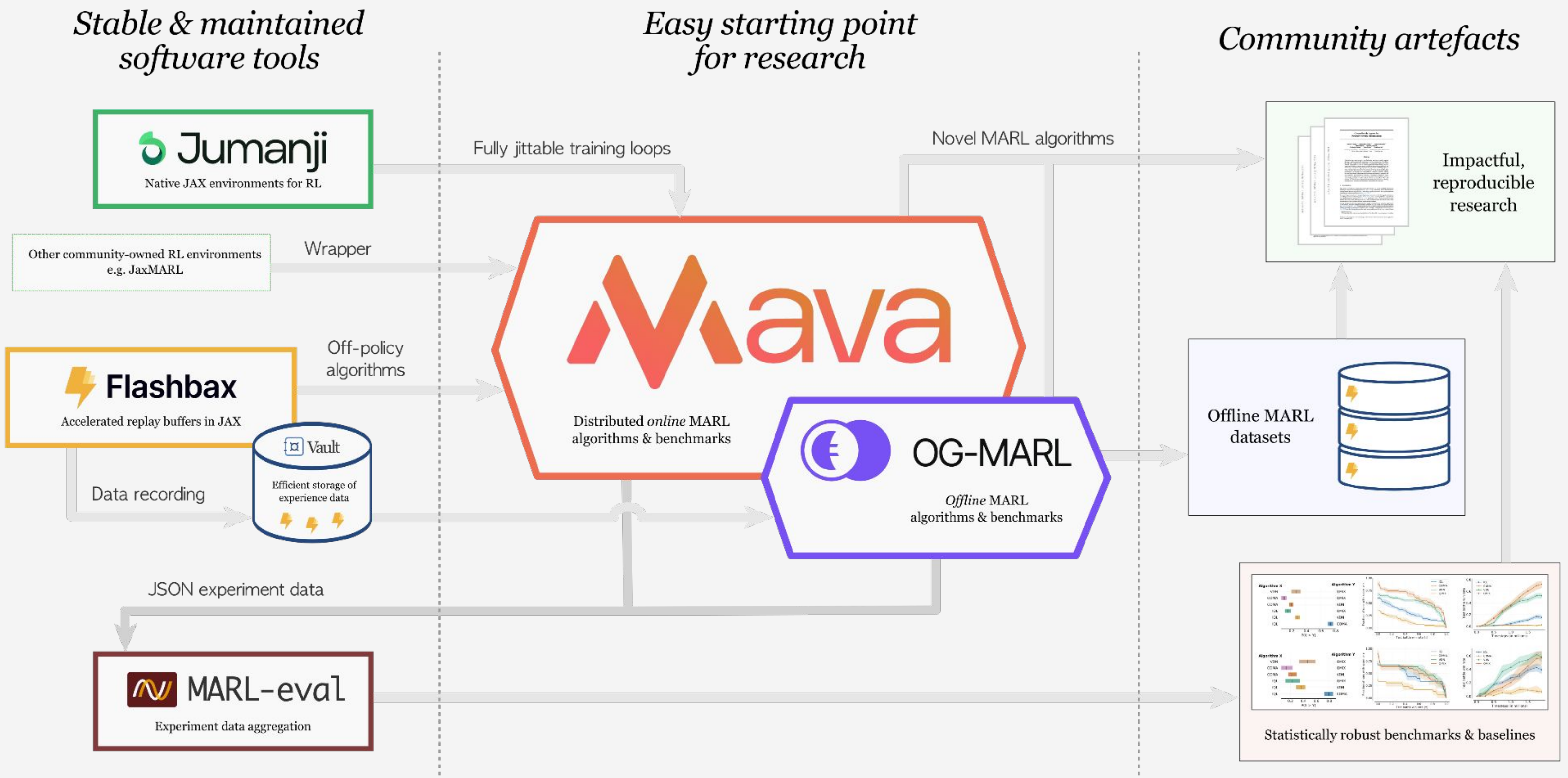

Stable & maintained software tools
Easy starting point for research
Community artefacts
Jumanji
Native JAX environments for RL
Fully jittable training loops
Other community-owned RL environments e.g. JaxMARL
Wrapper
Flashbax
Accelerated replay buffers in JAX
Off-policy algorithms
Vault
Efficient storage of experience data
Data recording
Mava
Distributed online MARL algorithms & benchmarks
OG-MARL
Offline MARL algorithms & benchmarks
Novel MARL algorithms
Impactful, reproducible research
Offline MARL datasets
JSON experiment data
MARL-eval
Experiment data aggregation
Statistically robust benchmarks & baselines

InstaDeep™

A JAX primer

```
jax.jit (f)(x)
jax.vmap(f)(x)
jax.pmap(f)(x)
```

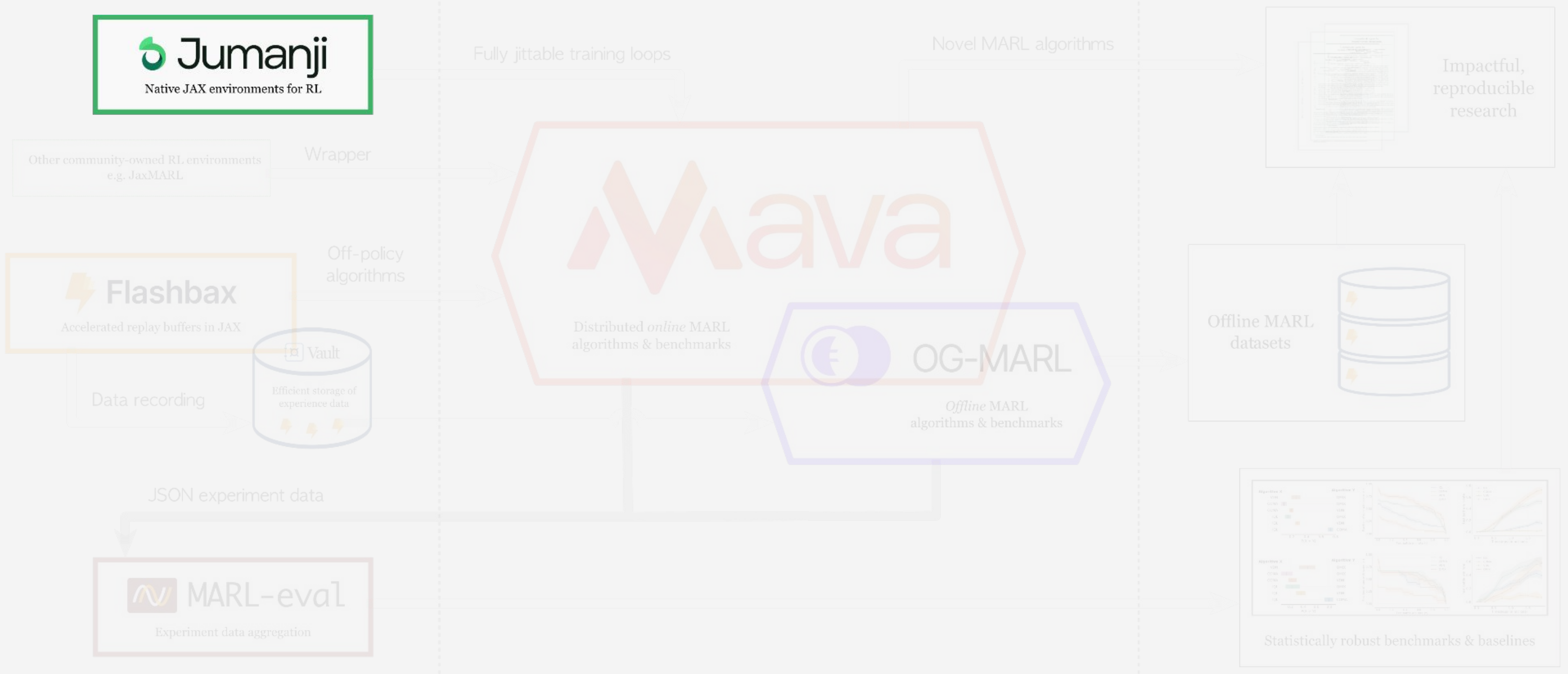
Bradbury et al., *JAX: composable transformations of Python+NumPy programs*, 2018


Stable & maintained software tools
Easy starting point for research
Community artefacts
Jumanji
Native JAX environments for RL
Fully jittable training loops
Other community-owned RL environments e.g. JaxMARL
Wrapper
Flashbax
Accelerated replay buffers in JAX
Off-policy algorithms
Vault
Efficient storage of experience data
Data recording
Mava
Distributed online MARL algorithms & benchmarks
OG-MARL
Offline MARL algorithms & benchmarks
Novel MARL algorithms
Impactful, reproducible research
Offline MARL datasets
JSON experiment data
MARL-eval
Experiment data aggregation
Statistically robust benchmarks & baselines

Stable & maintained software tools

Easy starting point for research

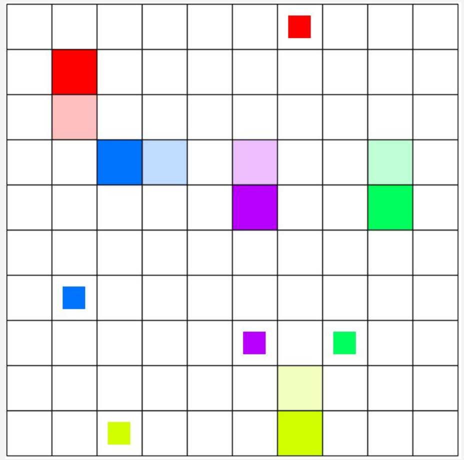
Community artefacts



Jumanji
Native JAX environments for RL
Fully jittable training loops
Novel MARL algorithms
Impactful, reproducible research
Other community-owned RL environments e.g. JaxMARL
Wrapper
Mava
Distributed online MARL algorithms & benchmarks
Off-policy algorithms
Flashbax
Accelerated replay buffers in JAX
Vault
Efficient storage of experience data
Data recording
OG-MARL
Offline MARL algorithms & benchmarks
Offline MARL datasets
JSON experiment data
MARL-eval
Experiment data aggregation
Statistically robust benchmarks & baselines

Jumanji

Native JAX environments for RL

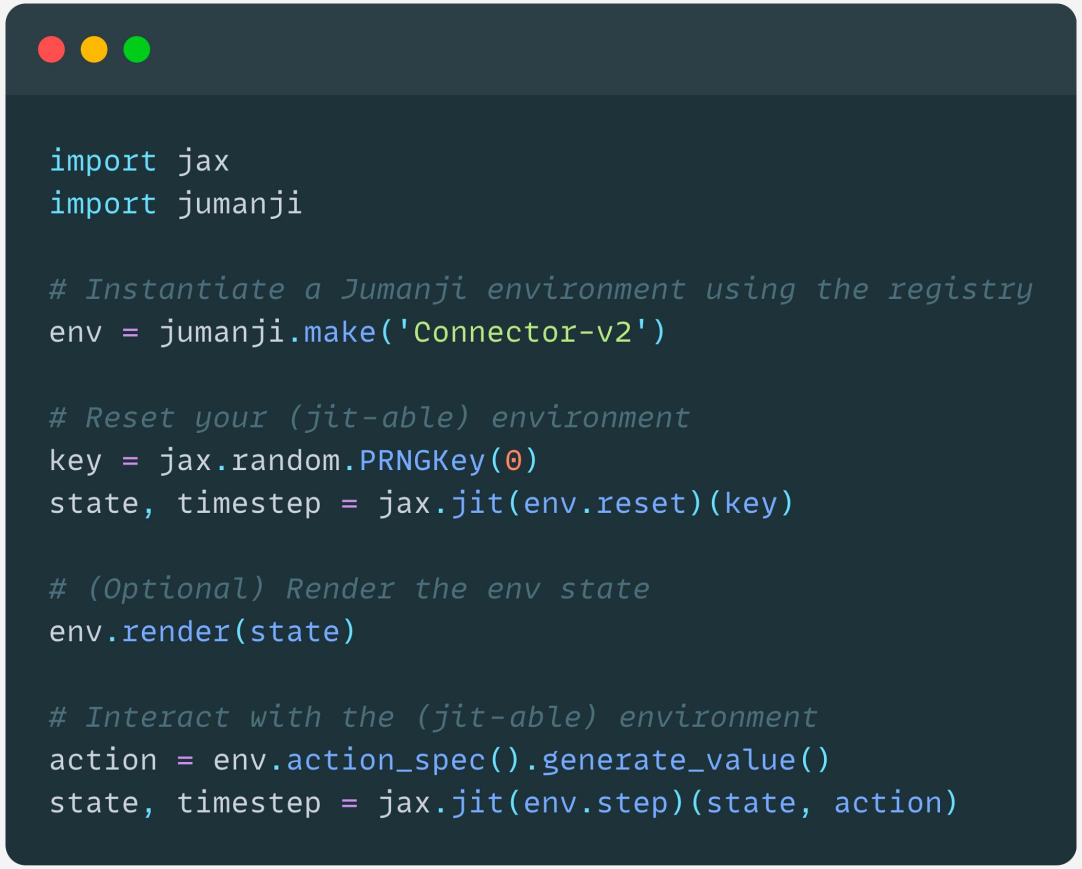


pip installable

familiar dm-env API

reproducible rollouts

jit/pmap environments



```python
import jax
import jumanji

# Instantiate a Jumanji environment using the registry
env = jumanji.make('Connector-v2')

# Reset your (jit-able) environment
key = jax.random.PRNGKey(0)
state, timestep = jax.jit(env.reset)(key)

# (Optional) Render the env state
env.render(state)

# Interact with the (jit-able) environment
action = env.action_spec().generate_value()
state, timestep = jax.jit(env.step)(state, action)
```

bit.ly/
id-jumanji

Bonnet et al., *Jumanji: a Diverse Suite of Scalable Reinforcement Learning Environments in JAX*,
12th International Conference on Learning Representations (ICLR 2024)

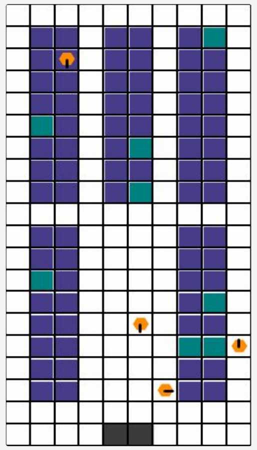
Jumanji

Native JAX environments for RL

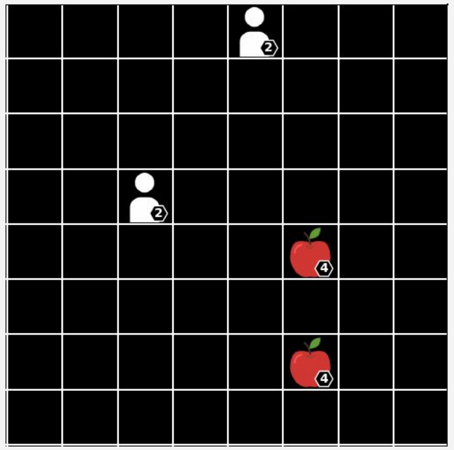
Existing multi-agent environments

New multi-agent environments

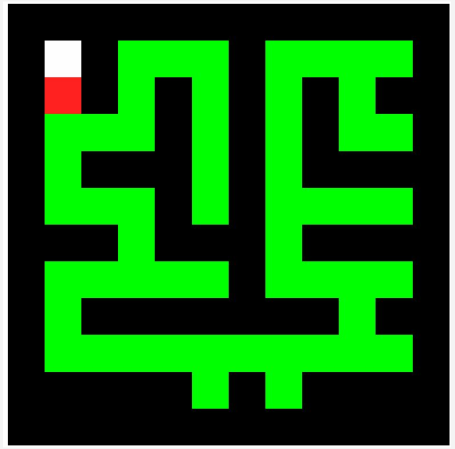
Robot Warehouse



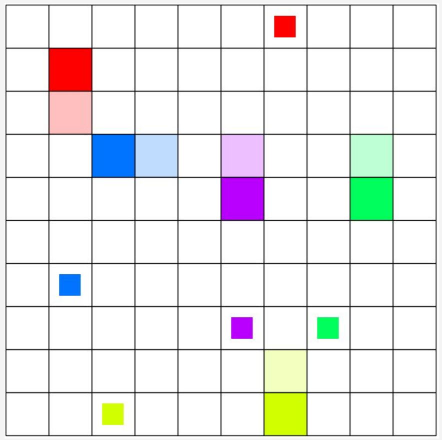
Level-Based Foraging



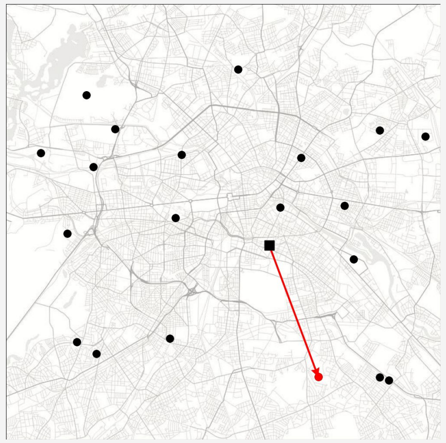
Cleaner



Connector



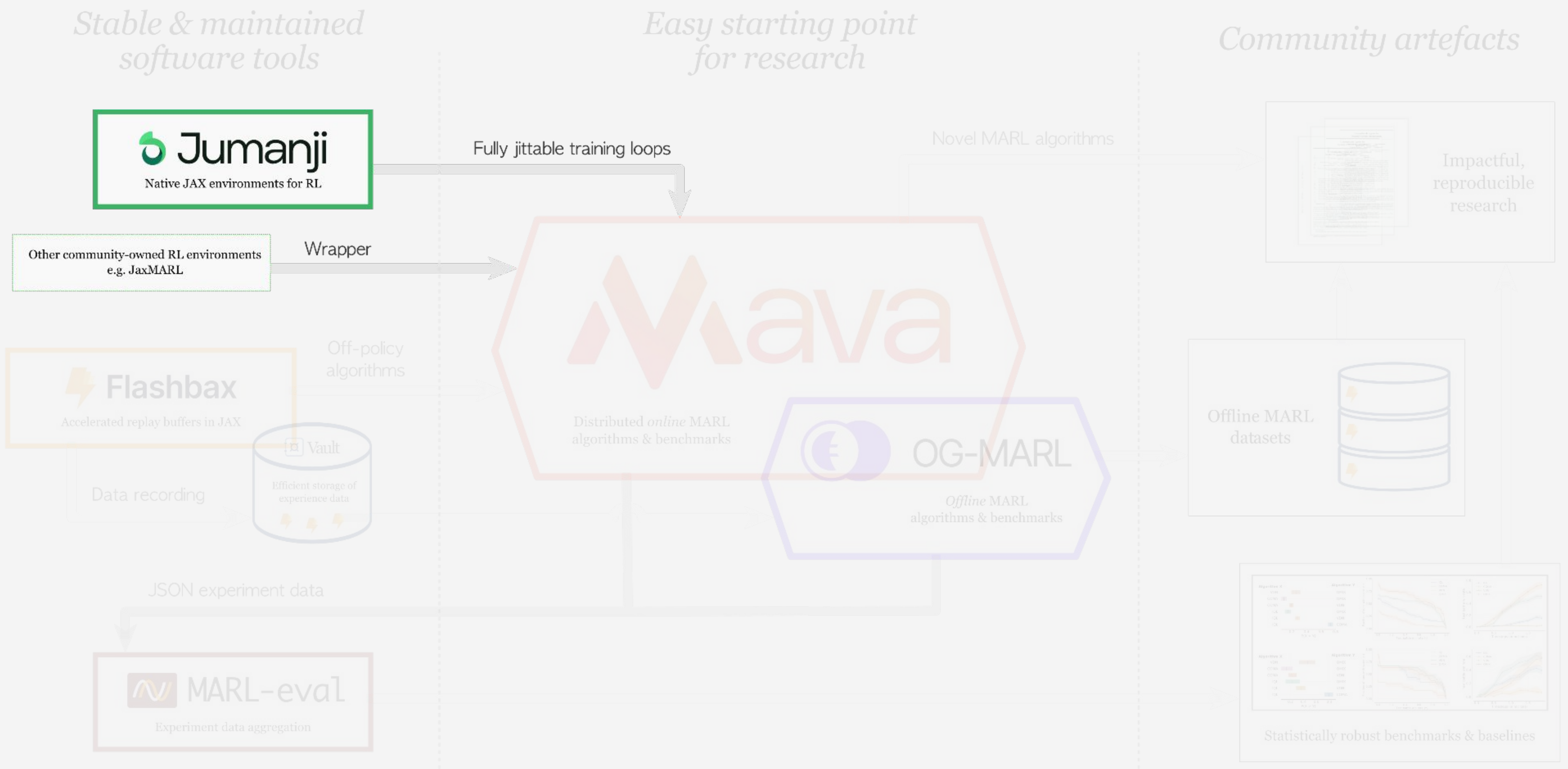
Multi-capacitive vehicle routing



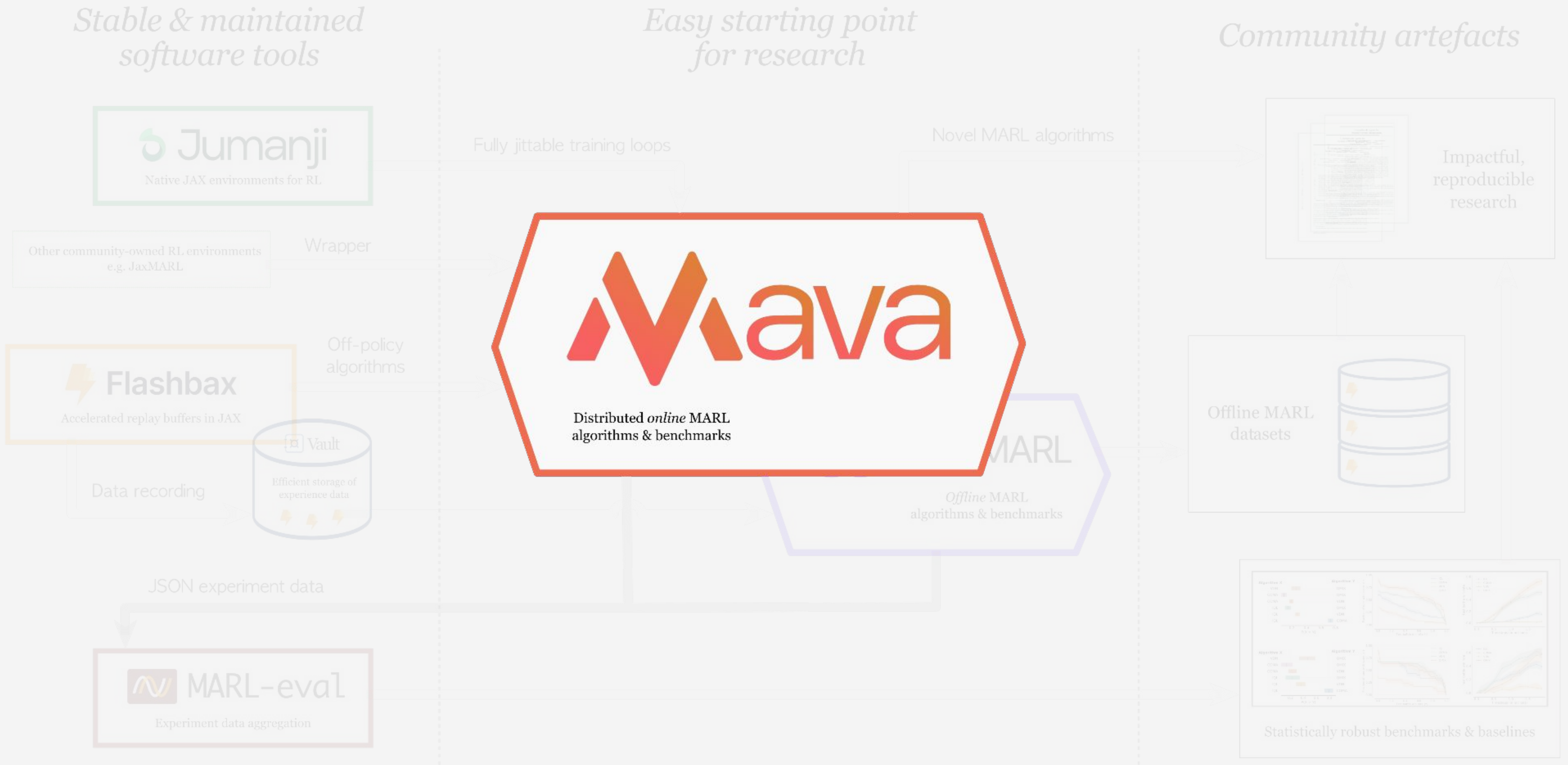
bit.ly/
id-jumanji

Papoudakis et al., *Benchmarking Multi-Agent Deep Reinforcement Learning Algorithms in Cooperative Tasks*, 35th Conference on Neural Information Processing Systems (NeurIPS 2021) Track on Datasets and Benchmarks

Christianos et al., *Shared Experience Actor-Critic for Multi-Agent Reinforcement Learning*, 34th Conference on Neural Information Processing Systems (NeurIPS 2020)

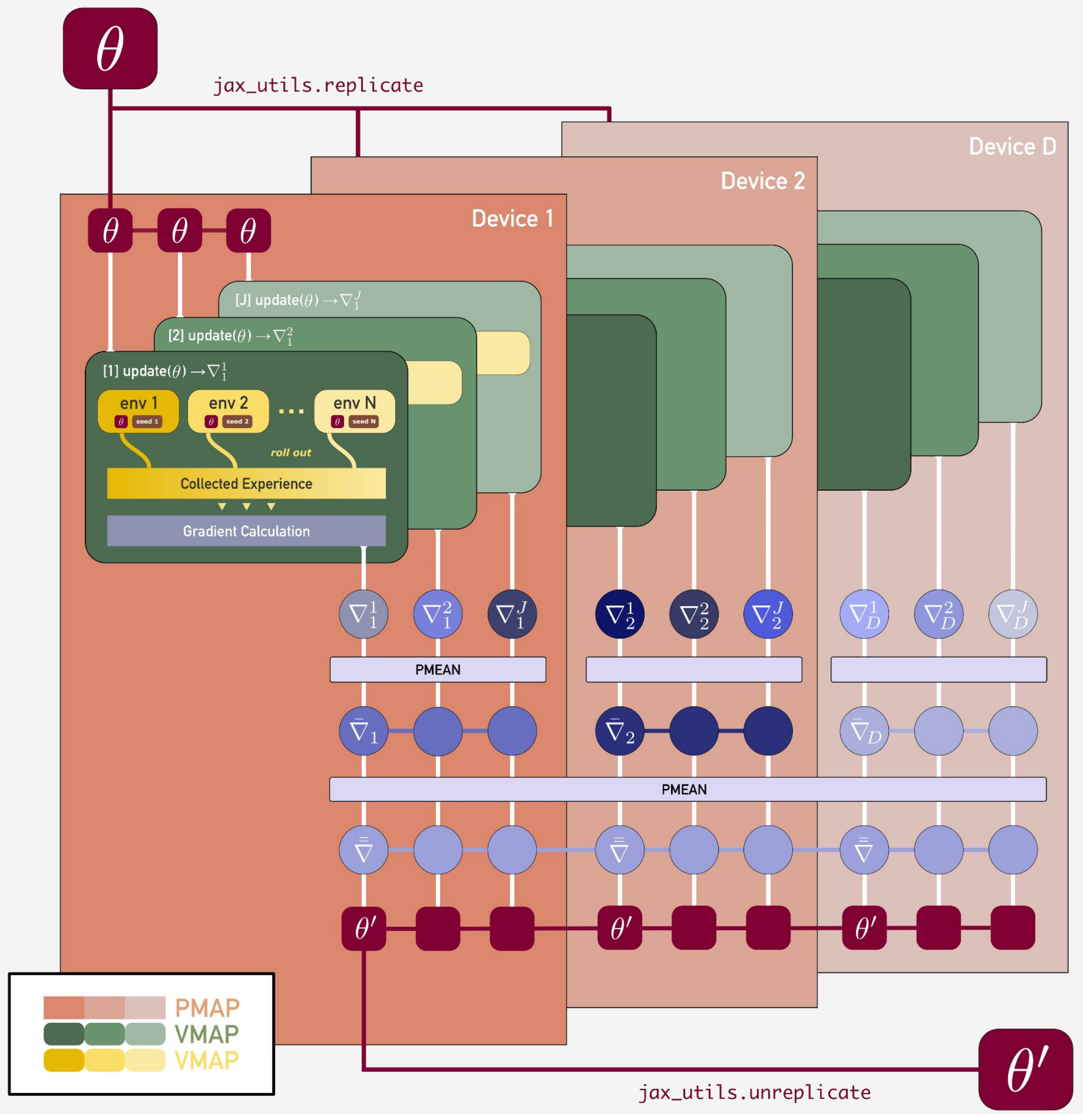


Stable & maintained software tools
Easy starting point for research
Community artefacts
Jumanji
Native JAX environments for RL
Fully jittable training loops
Other community-owned RL environments e.g. JaxMARL
Wrapper
Novel MARL algorithms
Impactful, reproducible research
Mava
Distributed online MARL algorithms & benchmarks
Flashbax
Accelerated replay buffers in JAX
Off-policy algorithms
Vault
Efficient storage of experience data
Data recording
OG-MARL
Offline MARL algorithms & benchmarks
Offline MARL datasets
JSON experiment data
MARL-eval
Experiment data aggregation
Statistically robust benchmarks & baselines


Stable & maintained software tools
Easy starting point for research
Community artefacts
Jumanji
Native JAX environments for RL
Fully jittable training loops
Novel MARL algorithms
Impactful, reproducible research
Other community-owned RL environments e.g. JaxMARL
Wrapper
Mava
Distributed *online* MARL algorithms & benchmarks
Off-policy algorithms
Flashbax
Accelerated replay buffers in JAX
Vault
Efficient storage of experience data
Data recording
MARL
Offline MARL algorithms & benchmarks
Offline MARL datasets
JSON experiment data
MARL-eval
Experiment data aggregation
Statistically robust benchmarks & baselines

InstaDeep™

Distributed online MARL algorithms

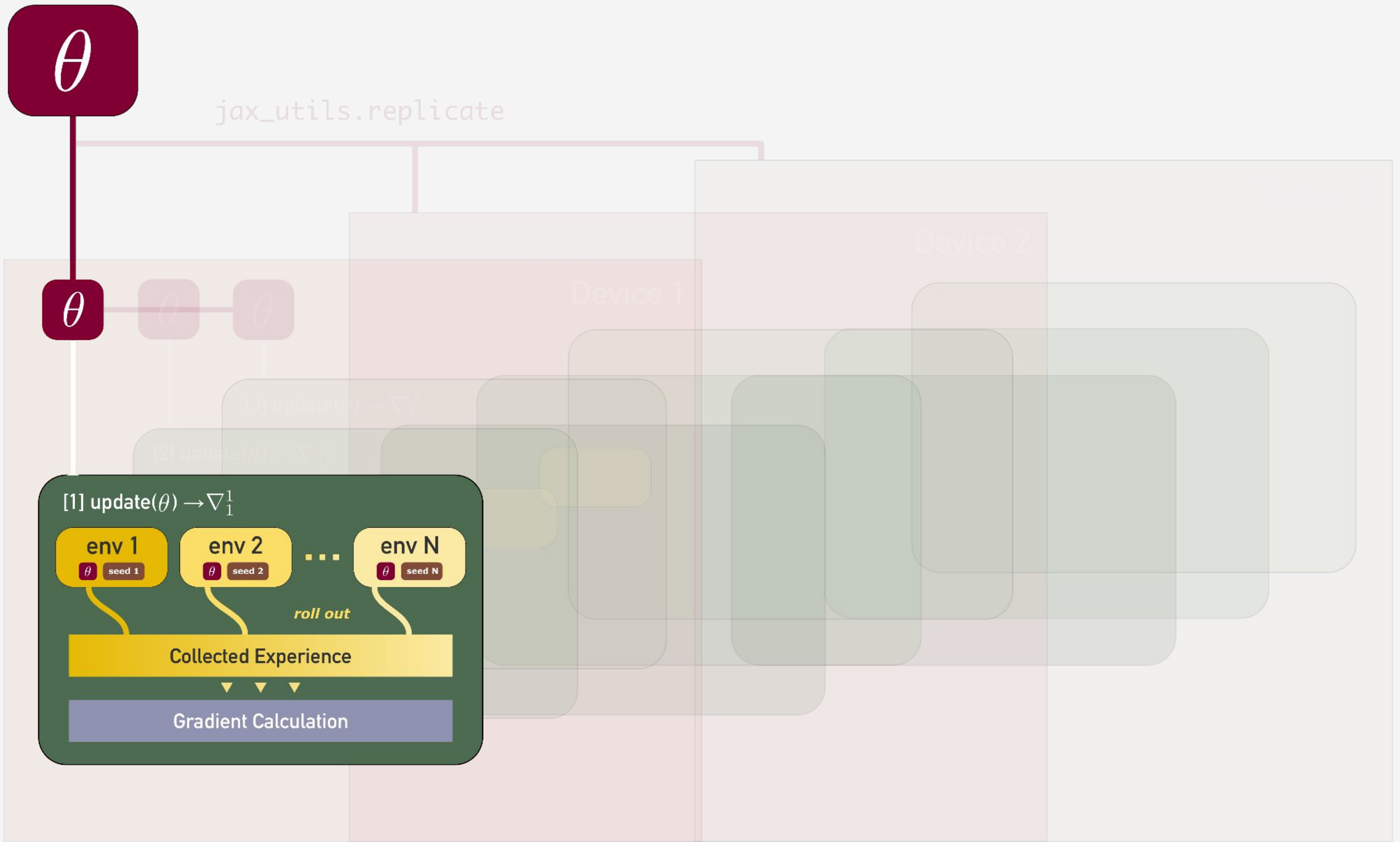


θ
jax_utils.replicate
Device 1
Device 2
Device D
θ θ θ
[J] update(θ) $\rightarrow \nabla_1^J$
[2] update(θ) $\rightarrow \nabla_1^2$
[1] update(θ) $\rightarrow \nabla_1^1$
env 1
env 2
env N
seed 1
seed 2
seed N
roll out
Collected Experience
Gradient Calculation
∇_1^1 ∇_1^2 ∇_1^J
∇_2^1 ∇_2^2 ∇_2^J
∇_D^1 ∇_D^2 ∇_D^J
PMEAN
$\bar{\nabla}_1$
$\bar{\nabla}_2$
$\bar{\nabla}_D$
PMEAN
$\bar{\bar{\nabla}}$
θ'
jax_utils.unreplicate
θ'
PMAP
VMAP
VMAP

Hessel et al., Podracer architectures for scalable reinforcement learning
arXiv preprint arXiv:2104.06272 (2021).

bit.ly/
id-mava

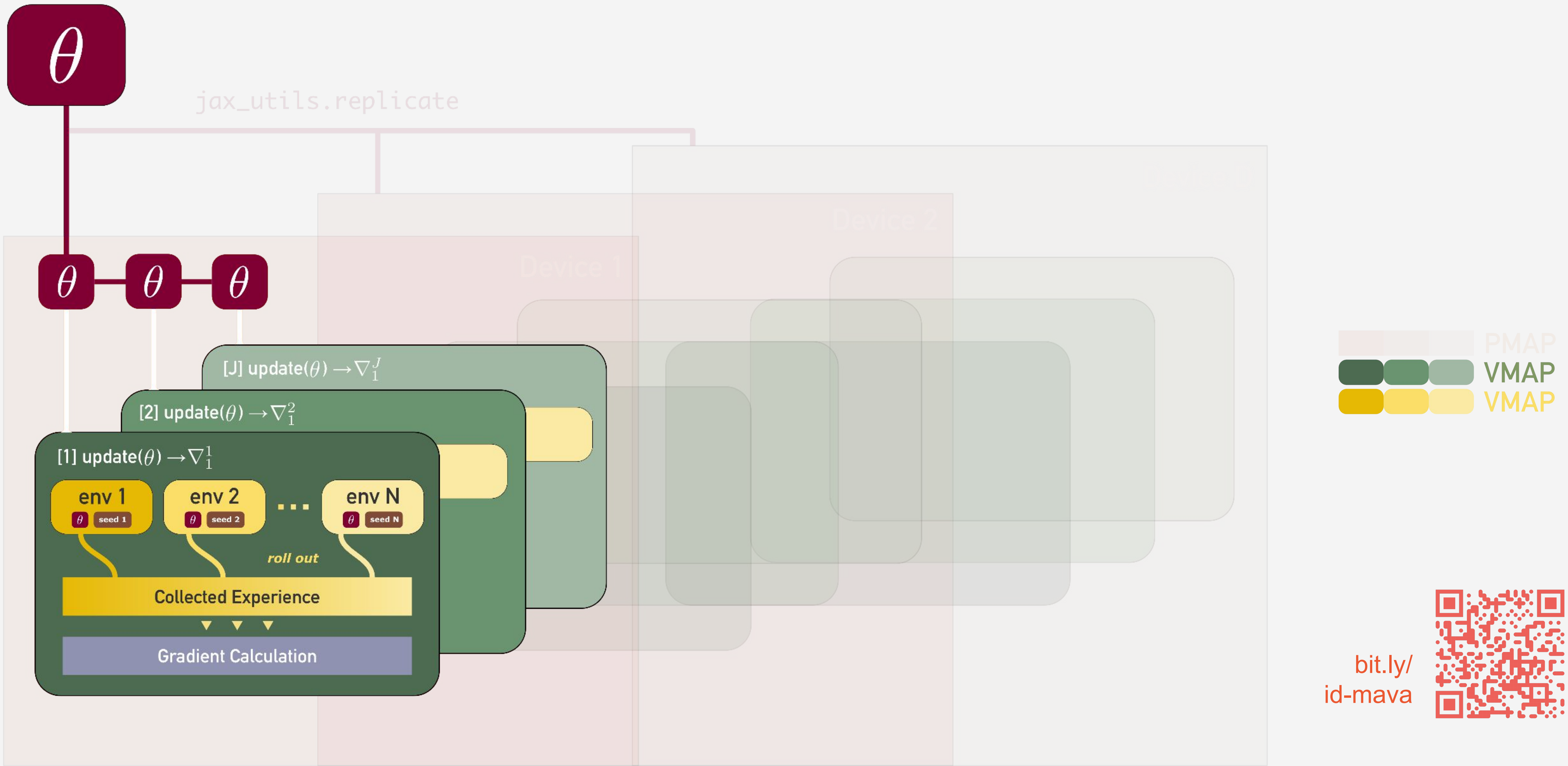
Distributed online MARL algorithms



bit.ly/
id-mava

Mava

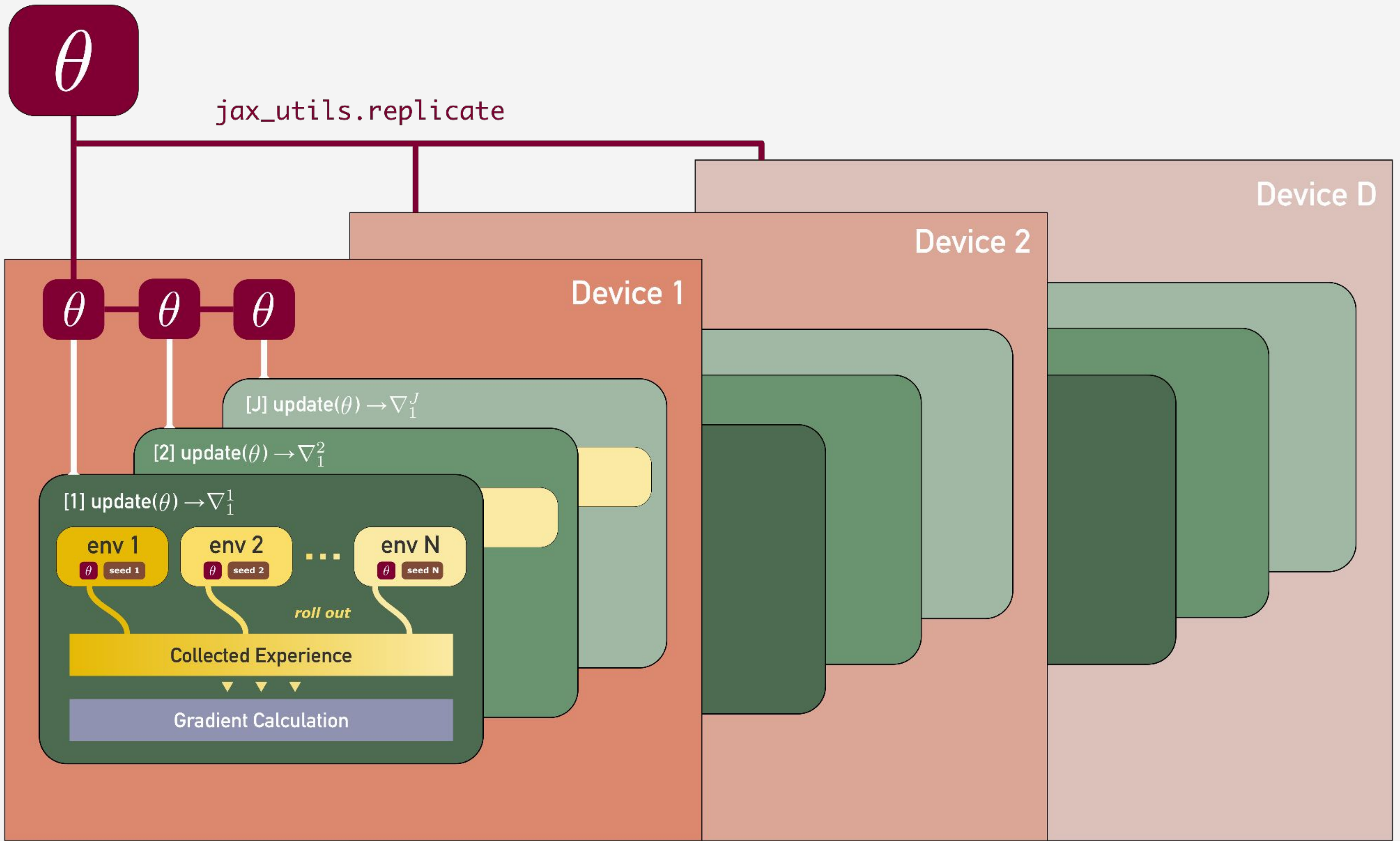
Distributed online MARL algorithms

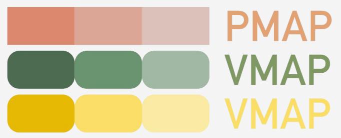


bit.ly/
id-mava

Mava

Distributed online MARL algorithms





bit.ly/
id-mava

Mava

Distributed online MARL algorithms

Variety of algorithms

- ✔ IPPO / MAPPO
- ✔ ISAC / MASAC
- ✔ IQL
- ✔ VDN
- ✔ QMIX

Supports JAX-envs

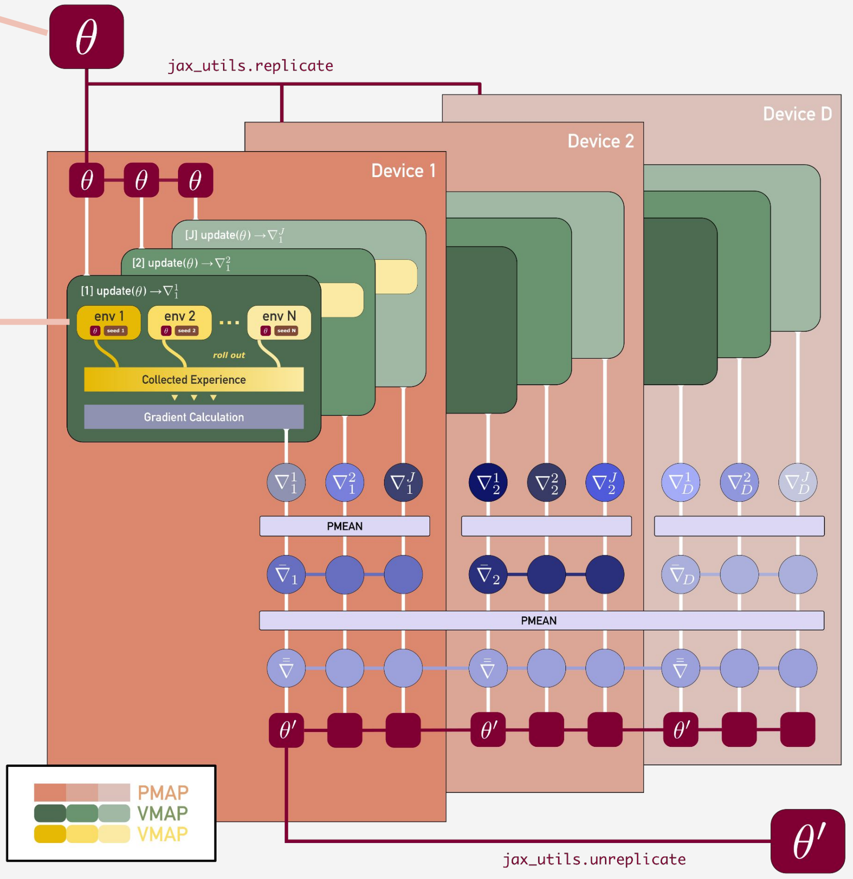
👾 More than 8 environments and many scenarios per environment

Jumanji

JaxMARL

GIGA STEP

Hardware acceleration via GPUs & TPUs


θ
jax_utils.replicate
Device 1
Device 2
Device D
θ θ θ
[J] update(θ) $\rightarrow \nabla_1^J$
[2] update(θ) $\rightarrow \nabla_1^2$
[1] update(θ) $\rightarrow \nabla_1^1$
env 1
env 2
env N
seed 1
seed 2
seed N
roll out
Collected Experience
Gradient Calculation
∇_1^1 ∇_1^2 ∇_1^J
∇_2^1 ∇_2^2 ∇_2^J
∇_D^1 ∇_D^2 ∇_D^J
PMEAN
∇_1 ∇_2 ∇_D
PMEAN
$\bar{\nabla}$
θ'
jax_utils.unreplicate
θ'
PMAP
VMAP
VMAP

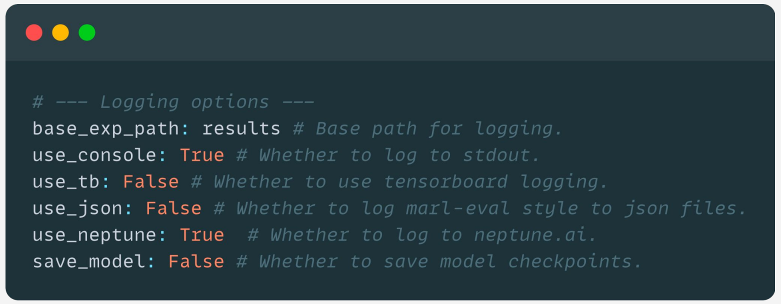
Research-friendly codebase

📃 Single-file implementation.

⚡ Core algorithm logic exposed in~400 lines.

Reliable

- Integrated robust evaluation.
- General and MARL utils such as networks, checkpointing and logging.



```
# --- Logging options ---
base_exp_path: results # Base path for logging.
use_console: True # Whether to log to stdout.
use_tb: False # Whether to use tensorboard logging.
use_json: False # Whether to log marl-eval style to json files.
use_neptune: True  # Whether to log to neptune.ai.
save_model: False # Whether to save model checkpoints.
```

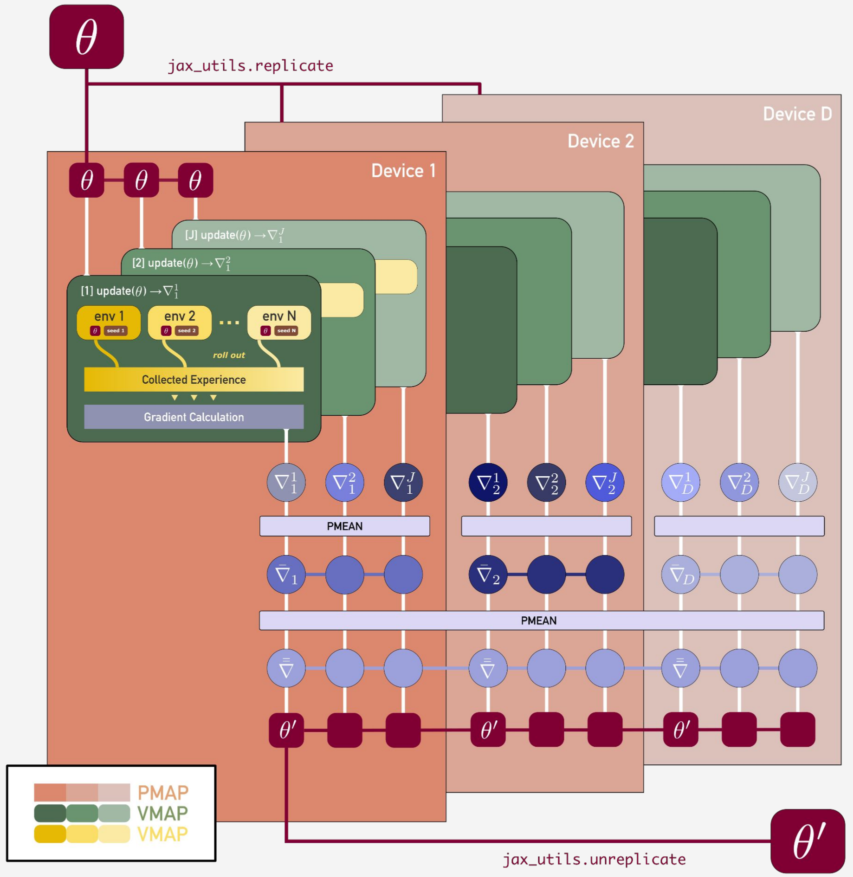
bit.ly/
id-mava

Agarwal et al., Deep Reinforcement Learning at the Edge of the Statistical Precipice, 35th Conference on Neural Information Processing Systems (NeurIPS 2021)
Huang et al., CleanRL: High-quality Single-file Implementations of Deep Reinforcement Learning Algorithms, Journal of Machine Learning Research 23 (2022)
Lu et al., Discovered policy optimisation, 36th Conference on Neural Information Processing Systems (NeurIPS 2022)
Rutherford et al., JaxMARL: Multi-Agent RL Environments and Algorithms in JAX, Agent Learning in Open-Endedness (ALOE) workshop at NeurIPS 2023
Lechner et al., Gigastep - One Billion Steps per Second Multi-agent Reinforcement Learning, 37th Conference on Neural Information Processing Systems (NeurIPS 2023) Track on Datasets and Benchmarks

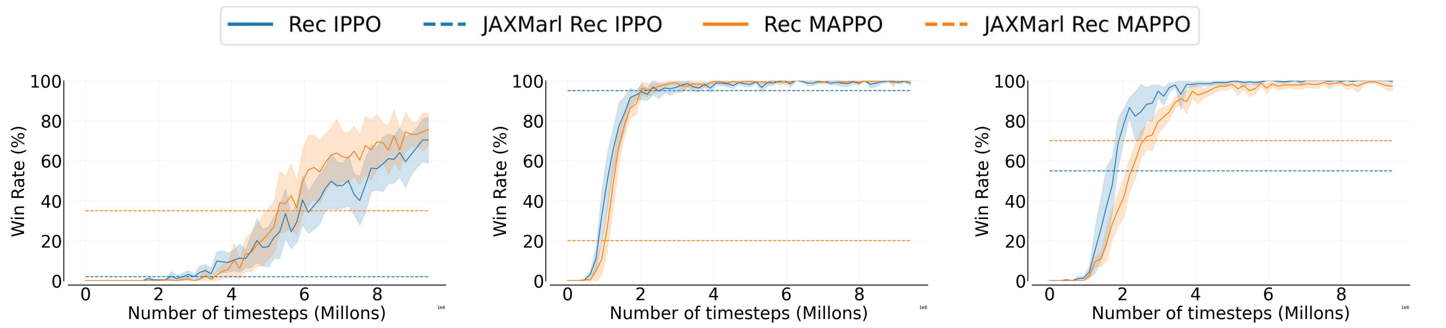
Mava

Distributed online MARL algorithms

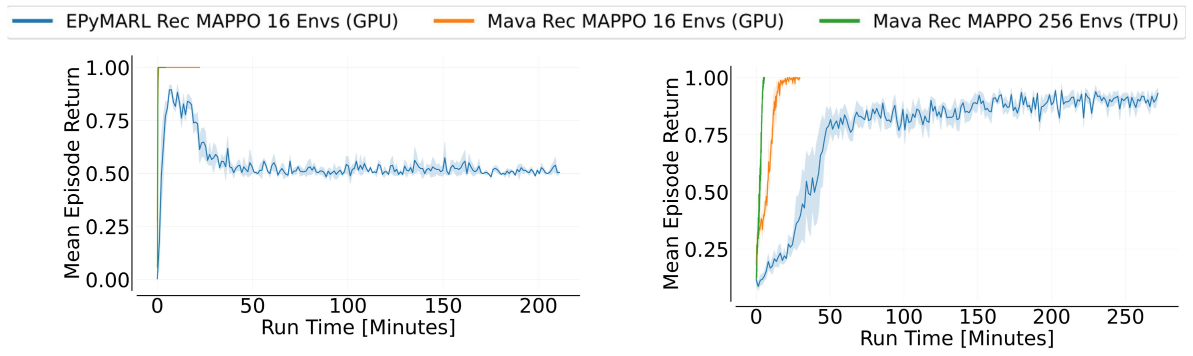
Hardware acceleration via GPUs & TPUs



Stable & performant

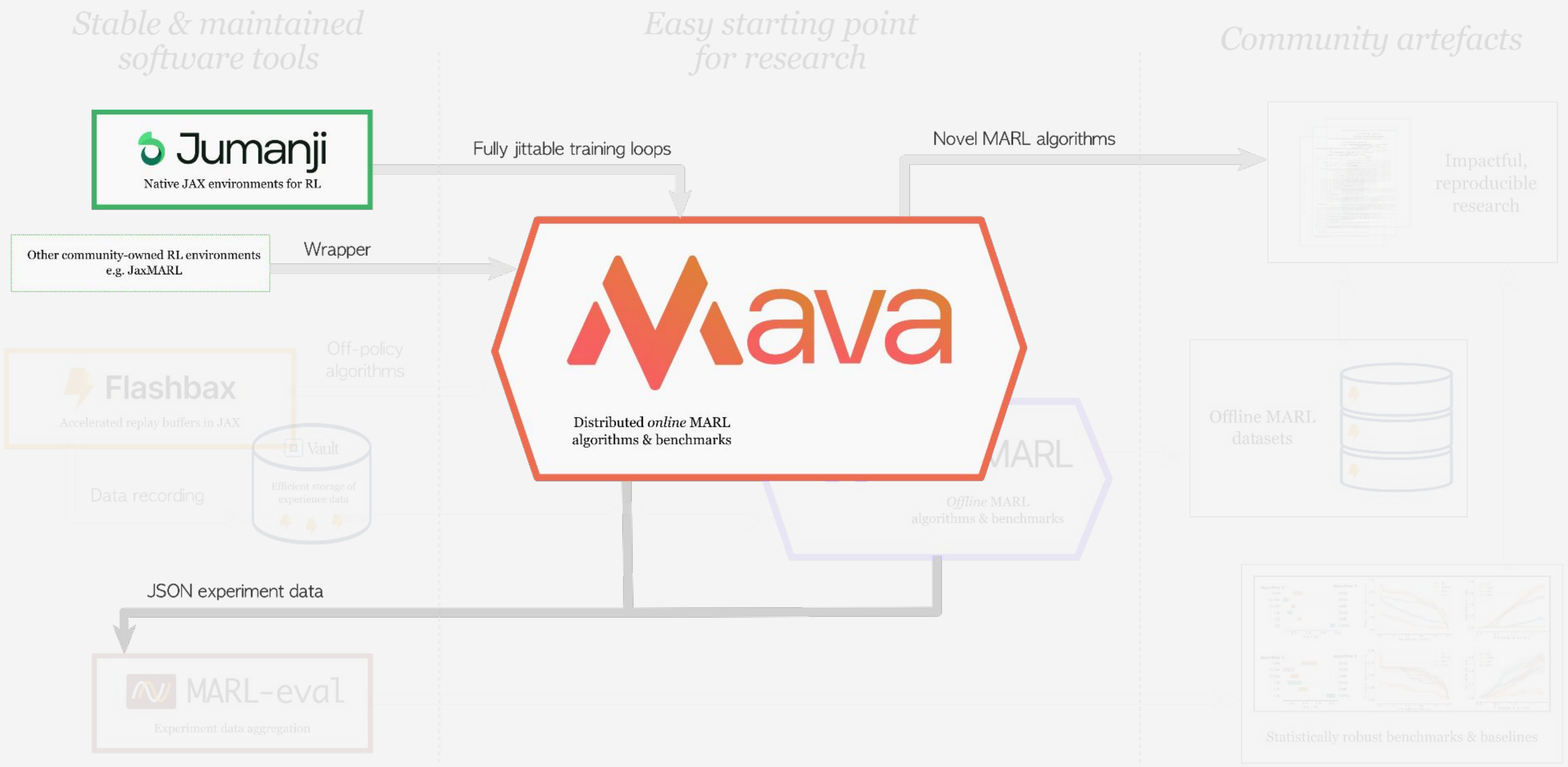


Mava Recurrent IPPO and MAPPO performance on the 3s5z, 6h_vs_8z and 3s5z_vs_3s6z SMAX tasks.

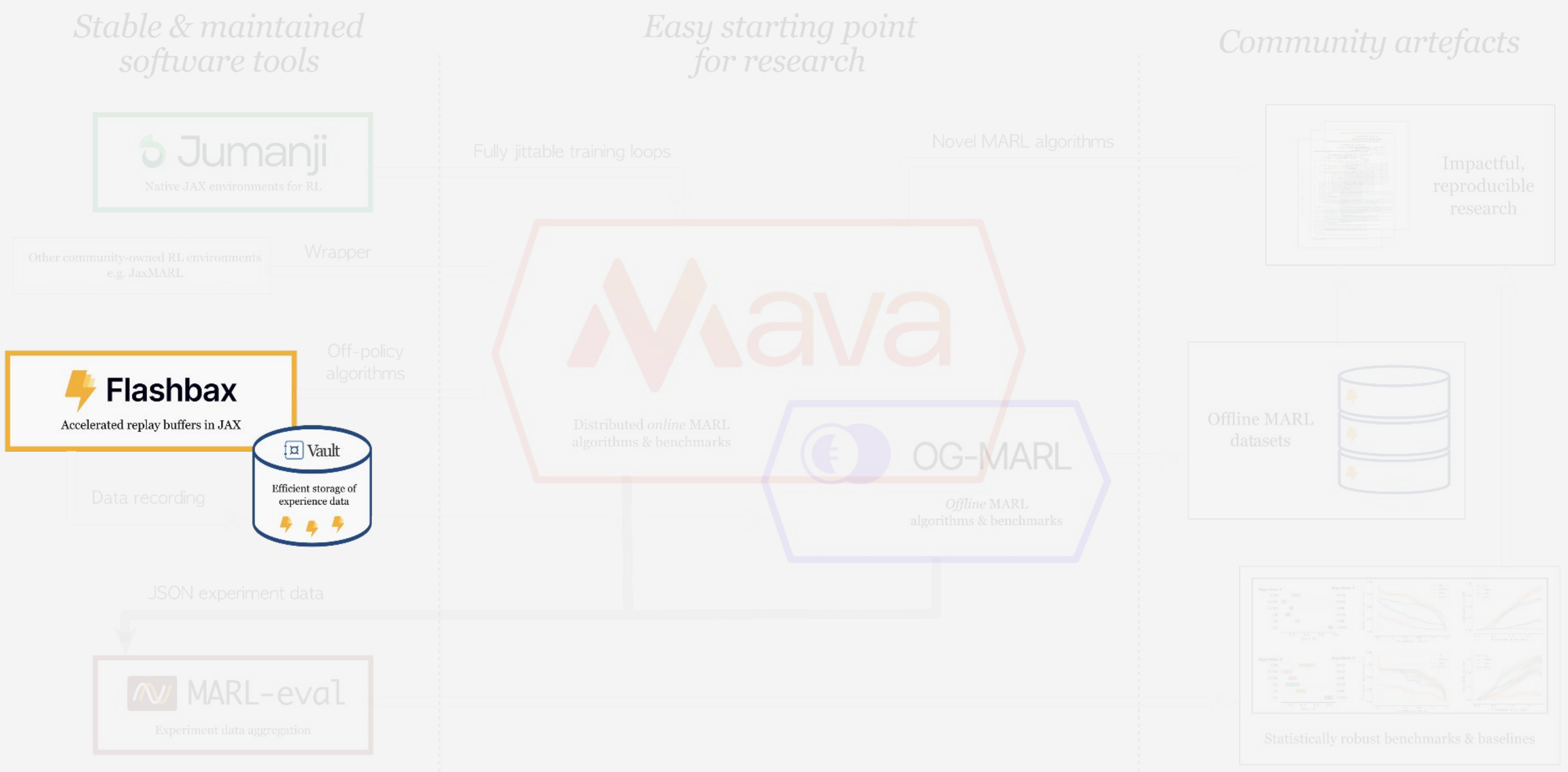


Mava Recurrent MAPPO performance on the 2s-8x8-2p-2f-coop, and 15x15-4p-3fz Level-Based Foraging tasks.

bit.ly/ id-mava


Stable & maintained software tools
Easy starting point for research
Community artefacts
Jumanji
Native JAX environments for RL
Fully jittable training loops
Novel MARL algorithms
Impactful, reproducible research
Other community-owned RL environments e.g. JaxMARL
Wrapper
Mava
Distributed online MARL algorithms & benchmarks
Flashbax
Accelerated replay buffers in JAX
Off-policy algorithms
Vault
Efficient storage of experience data
Data recording
MARL
Offline MARL algorithms & benchmarks
Offline MARL datasets
JSON experiment data
MARL-eval
Experiment data aggregation
Statistically robust benchmarks & baselines

InstaDeep™


Stable & maintained software tools
Easy starting point for research
Community artefacts
Jumanji
Native JAX environments for RL
Fully jittable training loops
Novel MARL algorithms
Impactful, reproducible research
Other community-owned RL environments e.g. JaxMARL
Wrapper
Mava
Off-policy algorithms
Flashbax
Accelerated replay buffers in JAX
Vault
Efficient storage of experience data
Data recording
Distributed online MARL algorithms & benchmarks
OG-MARL
Offline MARL algorithms & benchmarks
Offline MARL datasets
JSON experiment data
MARL-eval
Experiment data aggregation
Statistically robust benchmarks & baselines

Flashbax

Accelerated replay buffers in JAX

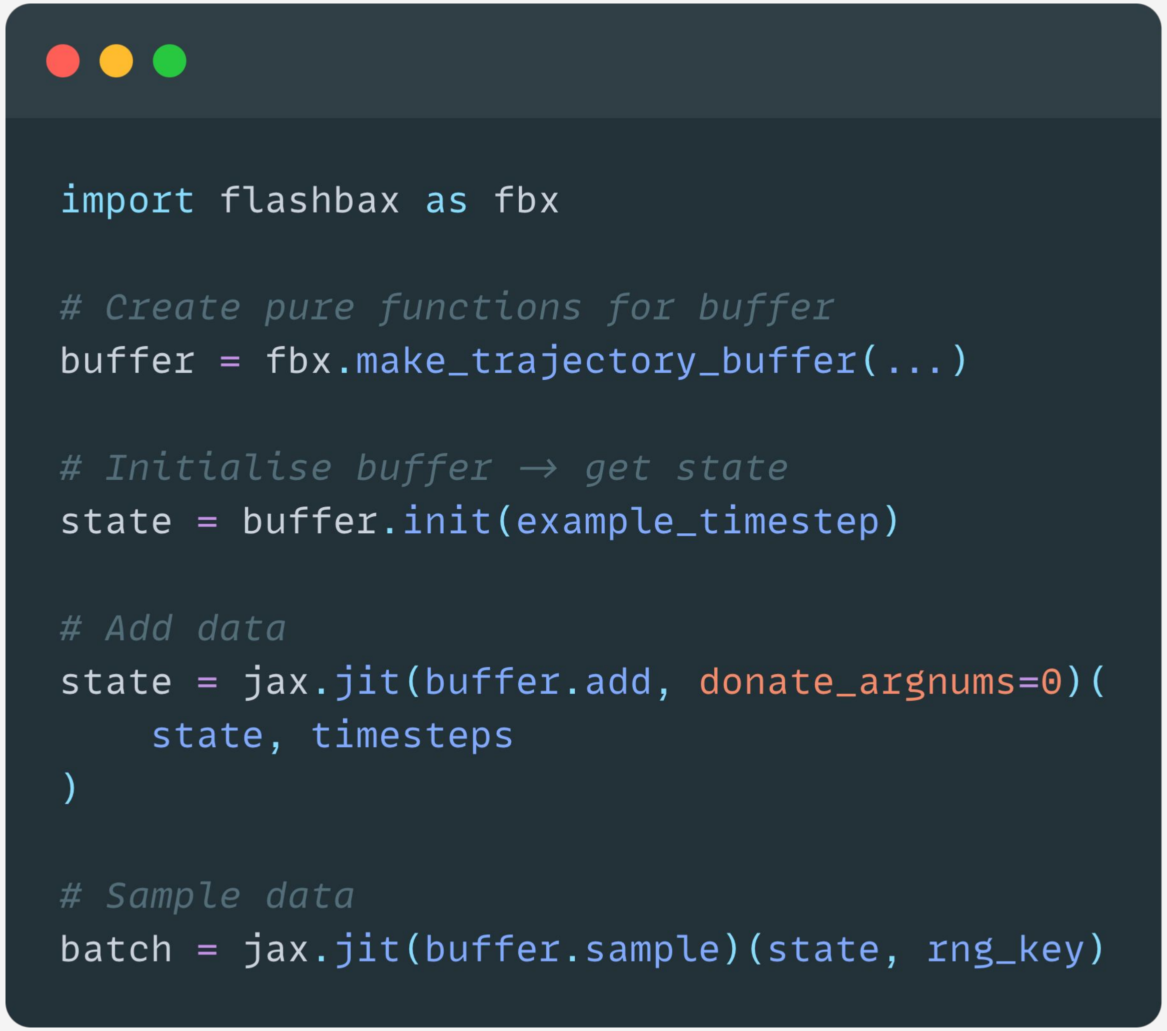
Vault

pip installable

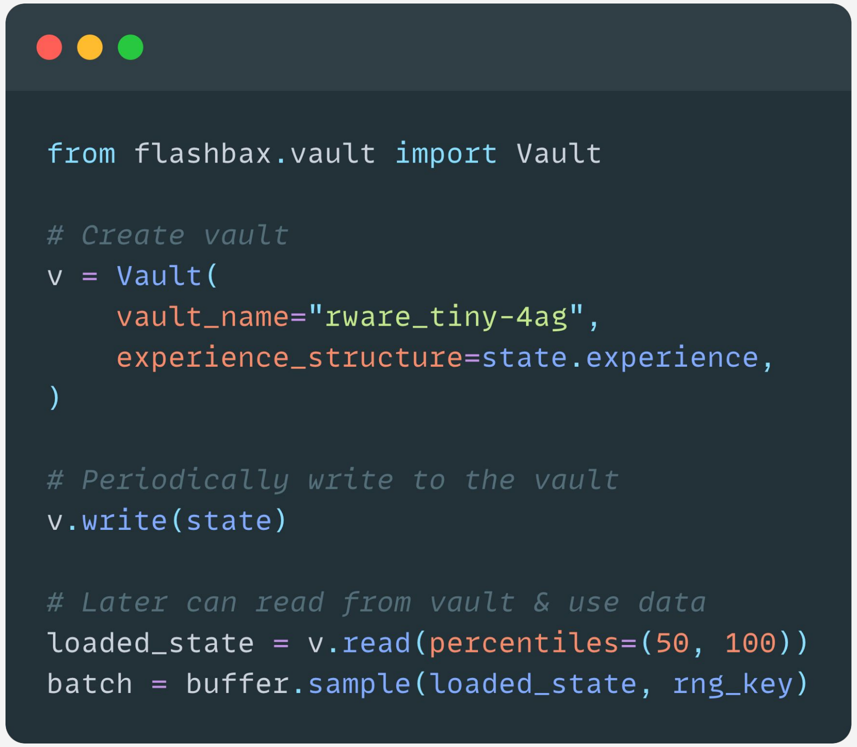
purely functional

jittable, with efficient memory management

jittable sampling

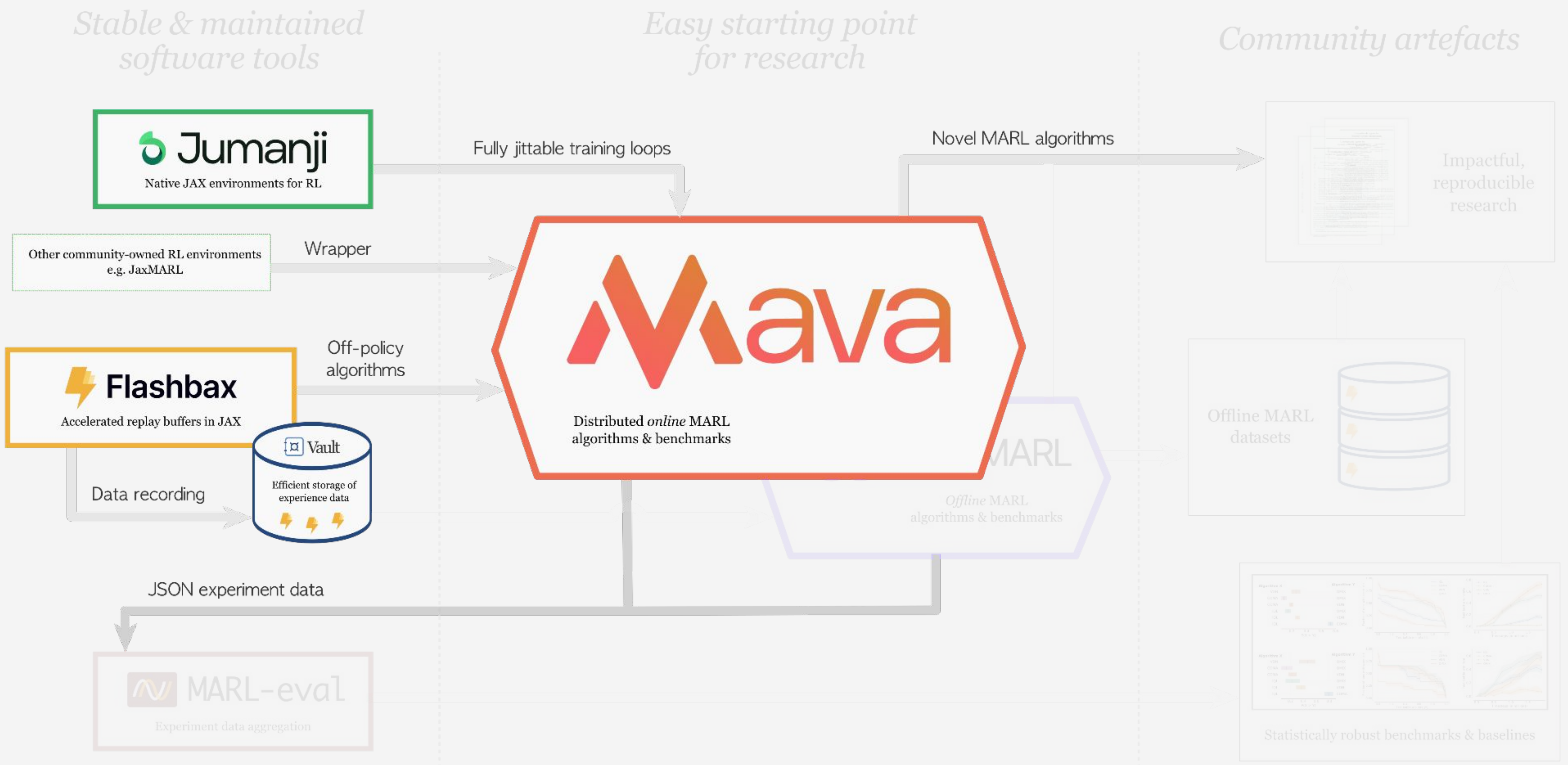


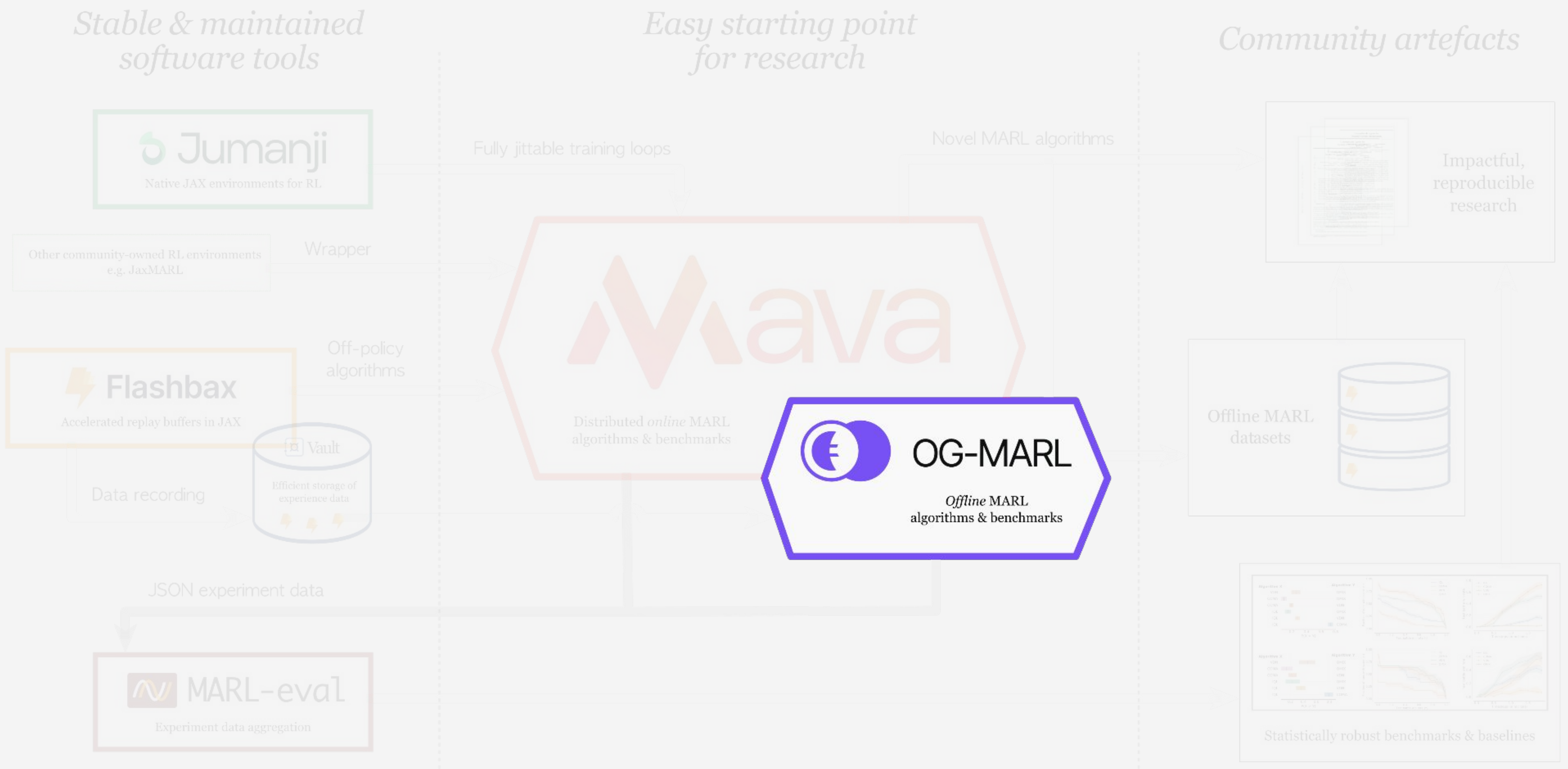
```python
import flashbax as fbx

# Create pure functions for buffer
buffer = fbx.make_trajectory_buffer(...)

# Initialise buffer → get state
state = buffer.init(example_timestep)

# Add data
state = jax.jit(buffer.add, donate_argnums=0)(
    state, timesteps
)

# Sample data
batch = jax.jit(buffer.sample)(state, rng_key)
```



```python
from flashbax.vault import Vault

# Create vault
v = Vault(
    vault_name="rware_tiny-4ag",
    experience_structure=state.experience,
)

# Periodically write to the vault
v.write(state)

# Later can read from vault & use data
loaded_state = v.read(percentiles=(50, 100))
batch = buffer.sample(loaded_state, rng_key)
```

bit.ly/
id-flashbax


Stable & maintained software tools
Easy starting point for research
Community artefacts
Jumanji
Native JAX environments for RL
Fully jittable training loops
Other community-owned RL environments e.g. JaxMARL
Wrapper
Flashbax
Accelerated replay buffers in JAX
Off-policy algorithms
Vault
Efficient storage of experience data
Data recording
Mava
Distributed online MARL algorithms & benchmarks
Novel MARL algorithms
Impactful, reproducible research
MARL
Offline MARL algorithms & benchmarks
Offline MARL datasets
JSON experiment data
MARL-eval
Experiment data aggregation
Statistically robust benchmarks & baselines

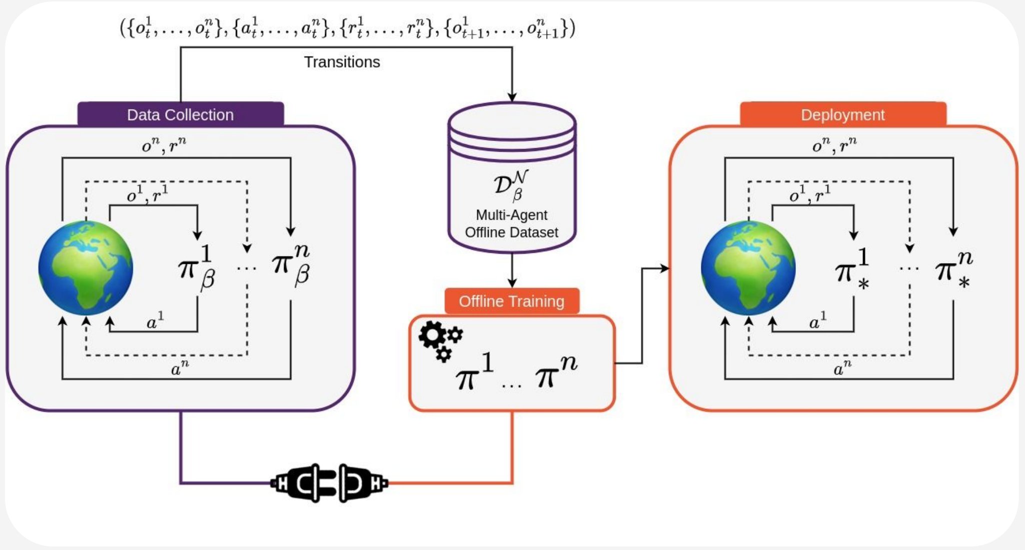

Stable & maintained software tools
Easy starting point for research
Community artefacts
Jumanji
Native JAX environments for RL
Fully jittable training loops
Novel MARL algorithms
Impactful, reproducible research
Other community-owned RL environments e.g. JaxMARL
Wrapper
Mava
Off-policy algorithms
Flashbax
Accelerated replay buffers in JAX
Distributed online MARL algorithms & benchmarks
OG-MARL
Offline MARL algorithms & benchmarks
Offline MARL datasets
Vault
Efficient storage of experience data
Data recording
JSON experiment data
MARL-eval
Experiment data aggregation
Statistically robust benchmarks & baselines

InstaDeep™

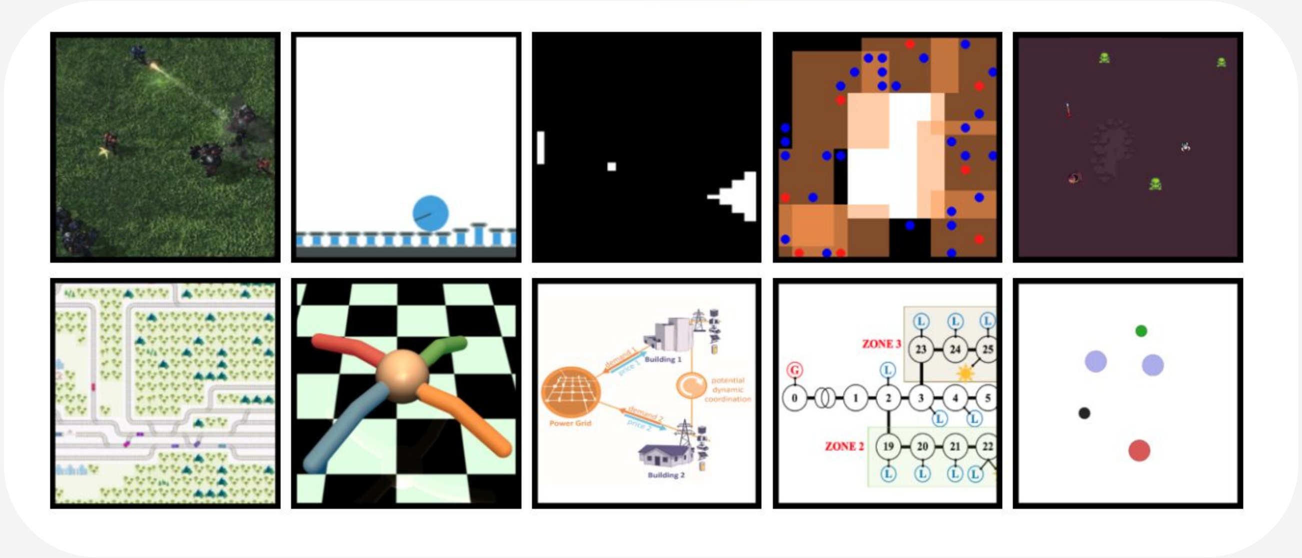
Off-the-Grid MARL

Datasets & baselines for offline MARL

Offline MARL



MARL Datasets

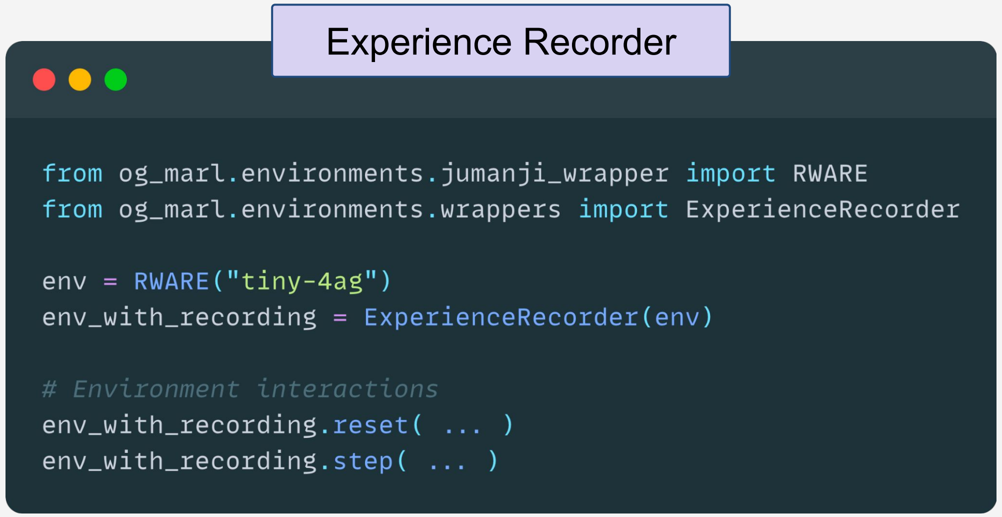


bit.ly/
id-ogmarl

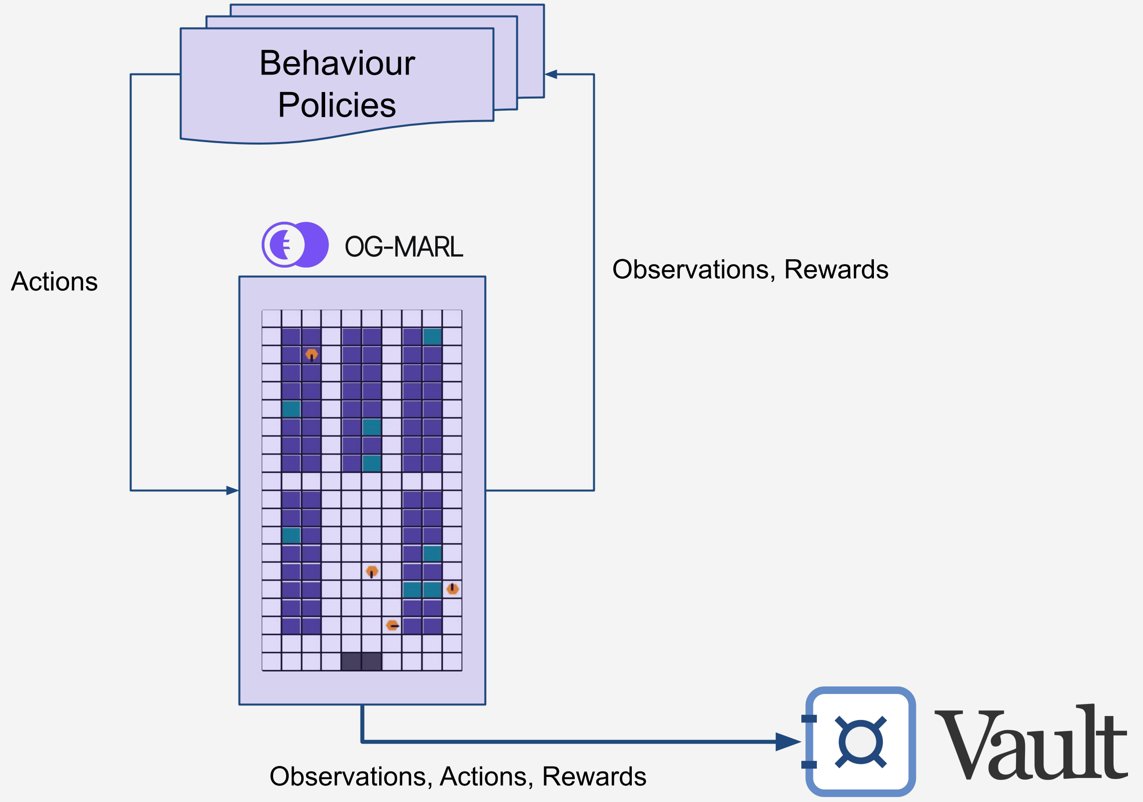
Formanek et al., *Off-the-Grid MARL: Datasets and Baselines for Offline Multi-Agent Reinforcement Learning*,
Extended Abstract at the 2023 International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2023)

Off-the-Grid MARL

Datasets & baselines for offline MARL


Experience Recorder

```
from og_marl.environments.jumanji_wrapper import RWARE
from og_marl.environments.wrappers import ExperienceRecorder

env = RWARE("tiny-4ag")
env_with_recording = ExperienceRecorder(env)

# Environment interactions
env_with_recording.reset( ... )
env_with_recording.step( ... )
```



bit.ly/
id-ogmarl

Formanek et al., *Off-the-Grid MARL: Datasets and Baselines for Offline Multi-Agent Reinforcement Learning*,

Off-the-Grid MARL

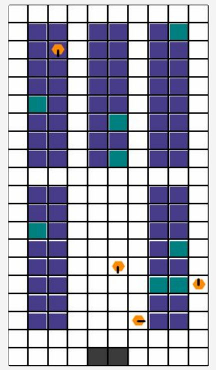
Datasets & baselines for offline MARL

Vault

OG-MARL

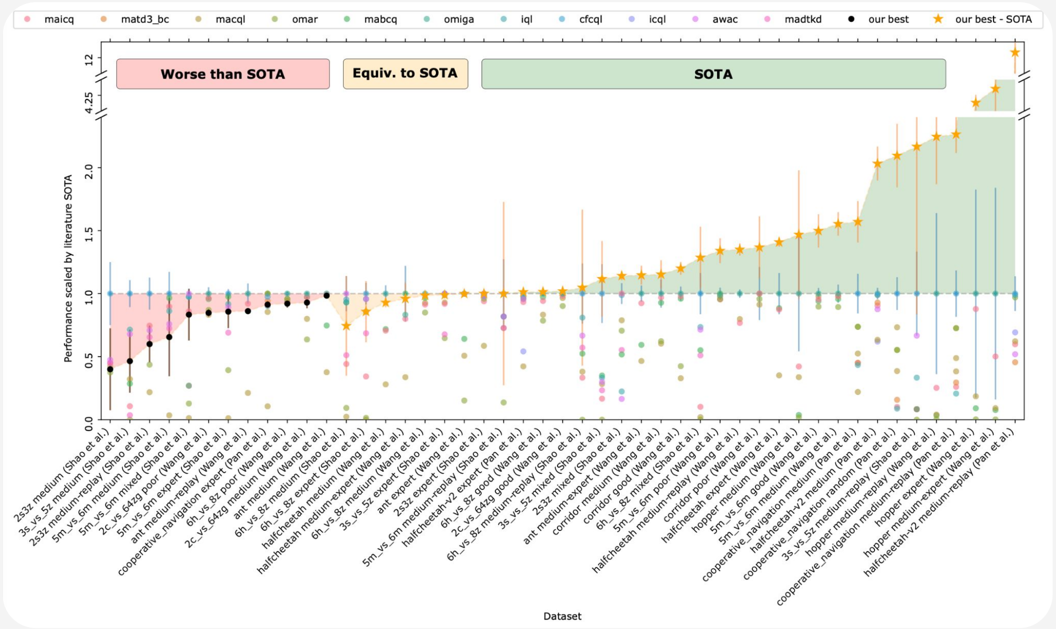
Baseline Algorithm Implementations

Evaluation



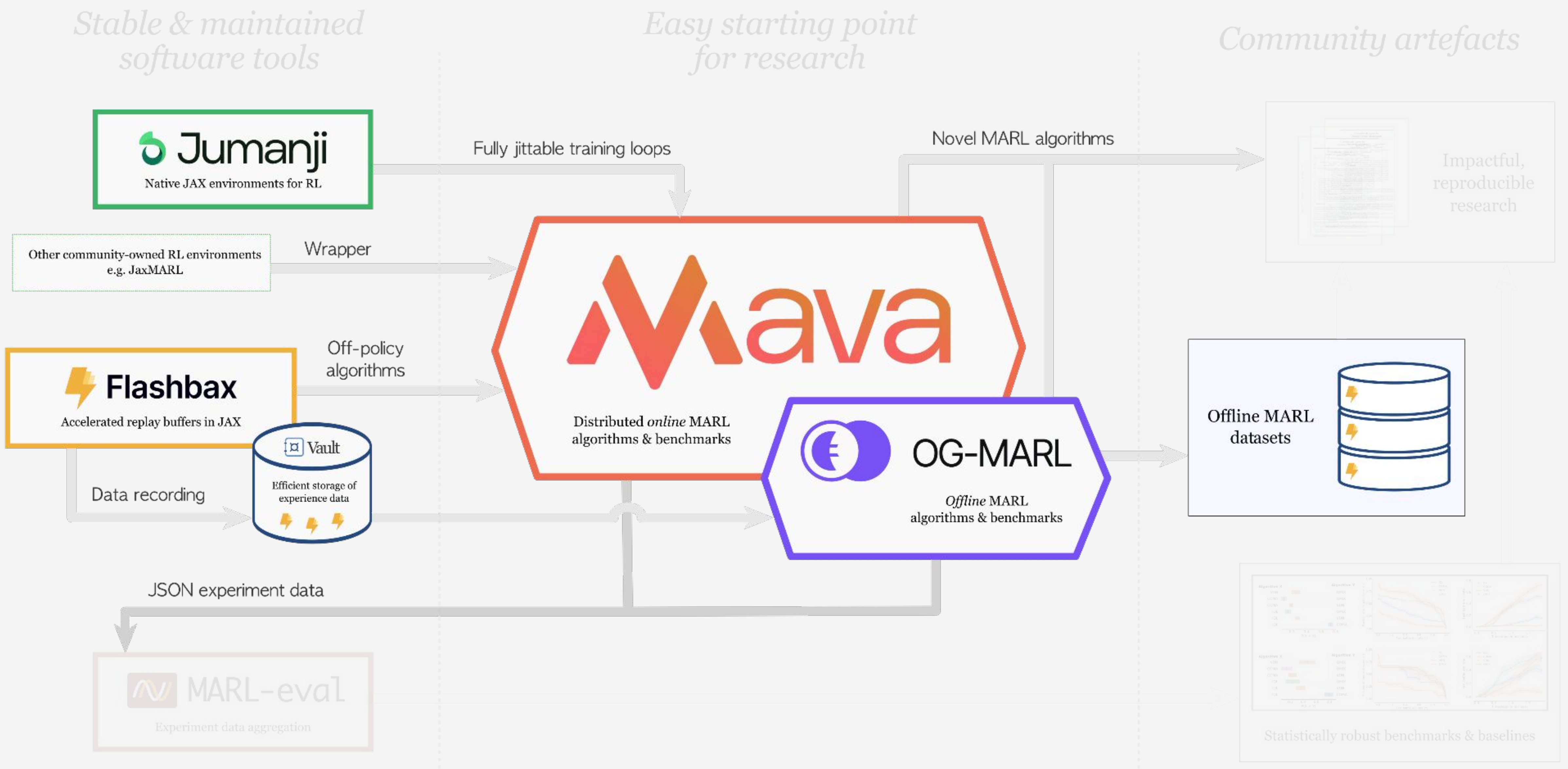
Baselines

```
$ python examples/download_vault.py
      --env=rware
      --scenario=tiny-4ag

$ python baselines/main.py
      --env=rware
      --scenario=tiny-4ag
      --dataset=Good
      --system=maicq
```



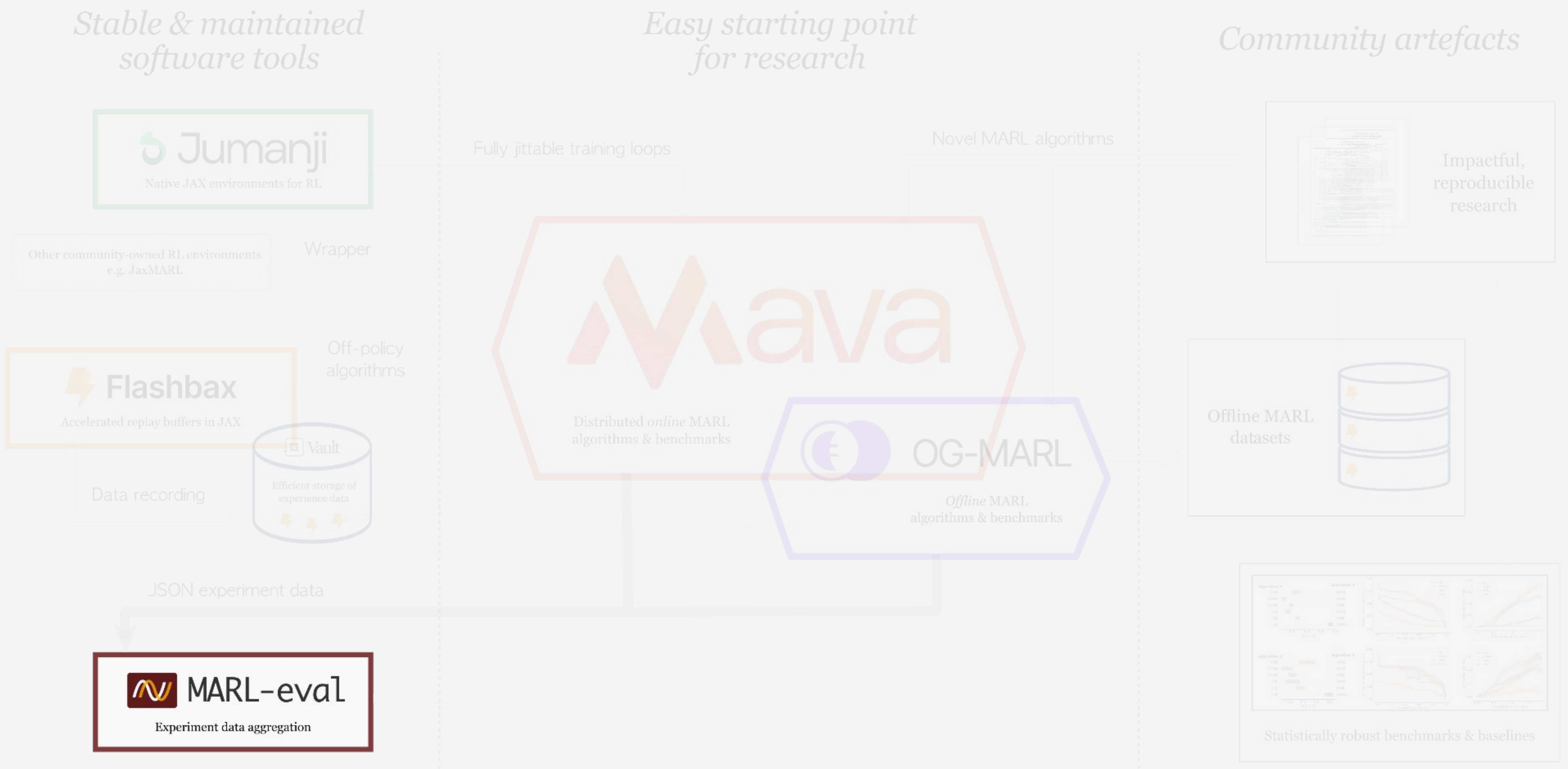
bit.ly/
id-ogmarl

Formanek et al., *Off-the-Grid MARL: Datasets and Baselines for Offline Multi-Agent Reinforcement Learning*,
Extended Abstract at the 2023 International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2023)



Stable & maintained software tools
Easy starting point for research
Community artefacts
Jumanji
Native JAX environments for RL
Fully jittable training loops
Other community-owned RL environments e.g. JaxMARL
Wrapper
Flashbax
Accelerated replay buffers in JAX
Off-policy algorithms
Vault
Efficient storage of experience data
Data recording
Mava
Distributed *online* MARL algorithms & benchmarks
OG-MARL
Offline MARL algorithms & benchmarks
Novel MARL algorithms
Impactful, reproducible research
Offline MARL datasets
JSON experiment data
MARL-eval
Experiment data aggregation
Statistically robust benchmarks & baselines

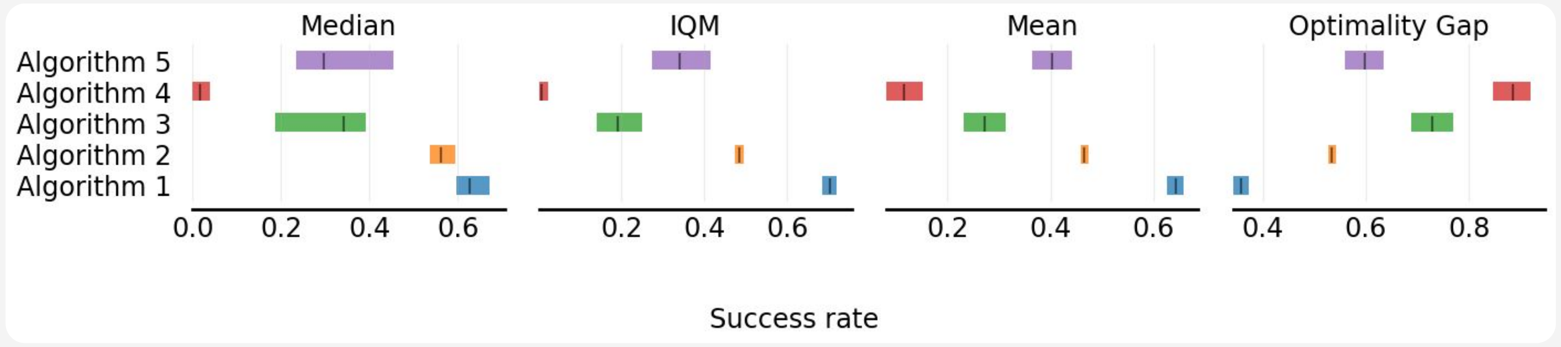
InstaDeep™


Stable & maintained software tools
Easy starting point for research
Community artefacts
Jumanji
Native JAX environments for RL
Fully jittable training loops
Novel MARL algorithms
Impactful, reproducible research
Other community-owned RL environments e.g. JaxMARL
Wrapper
Mava
Off-policy algorithms
Flashbax
Accelerated replay buffers in JAX
Vault
Efficient storage of experience data
Data recording
Distributed online MARL algorithms & benchmarks
OG-MARL
Offline MARL algorithms & benchmarks
Offline MARL datasets
JSON experiment data
MARL-eval
Experiment data aggregation
Statistically robust benchmarks & baselines

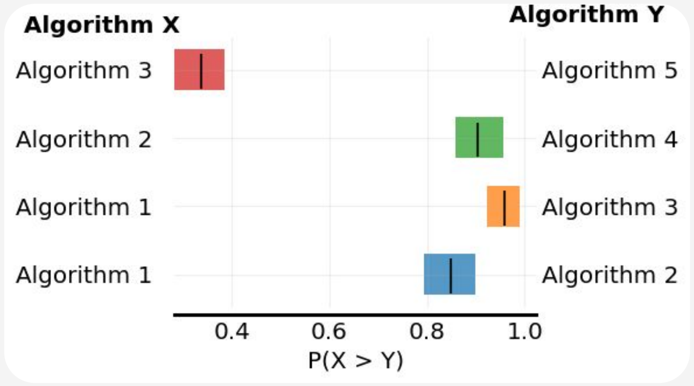
InstaDeep™

MARL-eval

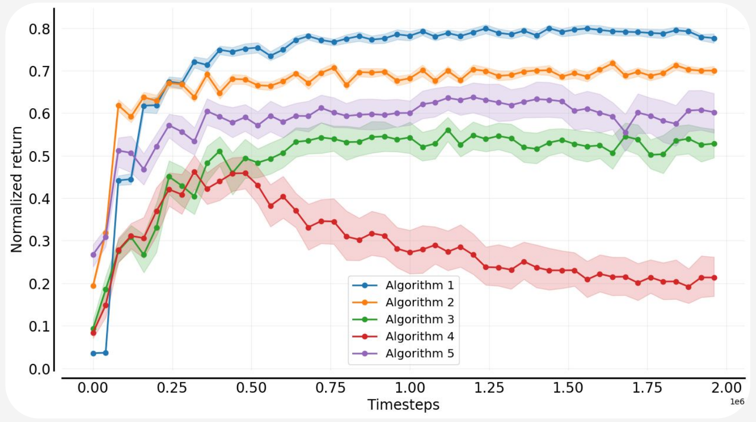
Statistically robust experiment result aggregation



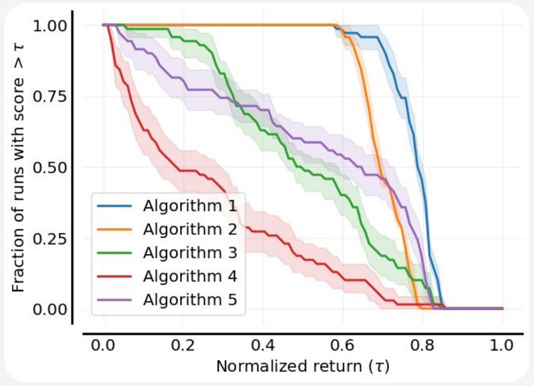
Aggregate Scores Plot



Probability of Improvement



Sample Efficiency Curve



Performance profile

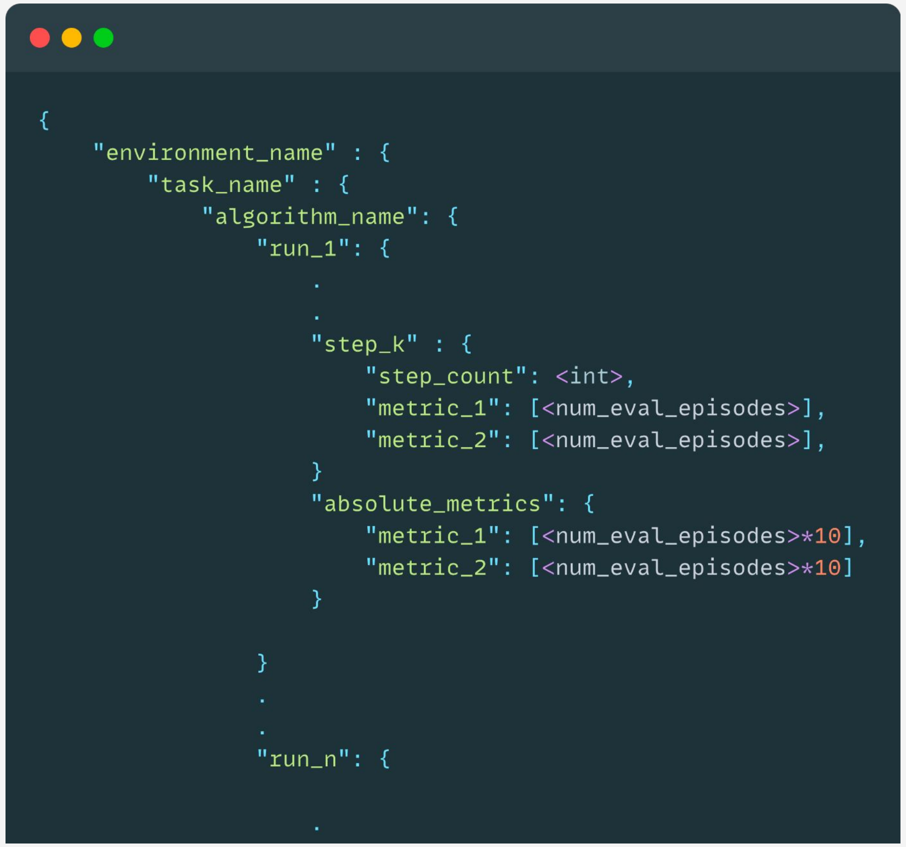
Agarwal et al., *Deep Reinforcement Learning at the Edge of the Statistical Precipice*, 35th Conference on Neural Information Processing Systems (NeurIPS 2021)
Gorsane et al., *Towards a Standardised Performance Evaluation Protocol for Cooperative MARL*, 36th Conference on Neural Information Processing Systems (NeurIPS 2022)

bit.ly/
id-marleval

MARL-eval

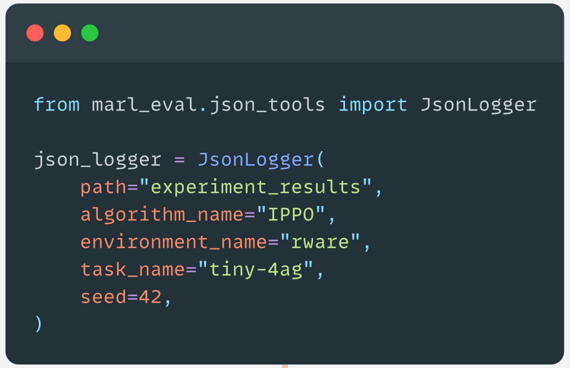
Statistically robust experiment result aggregation

Standardised data structure for raw experiment data

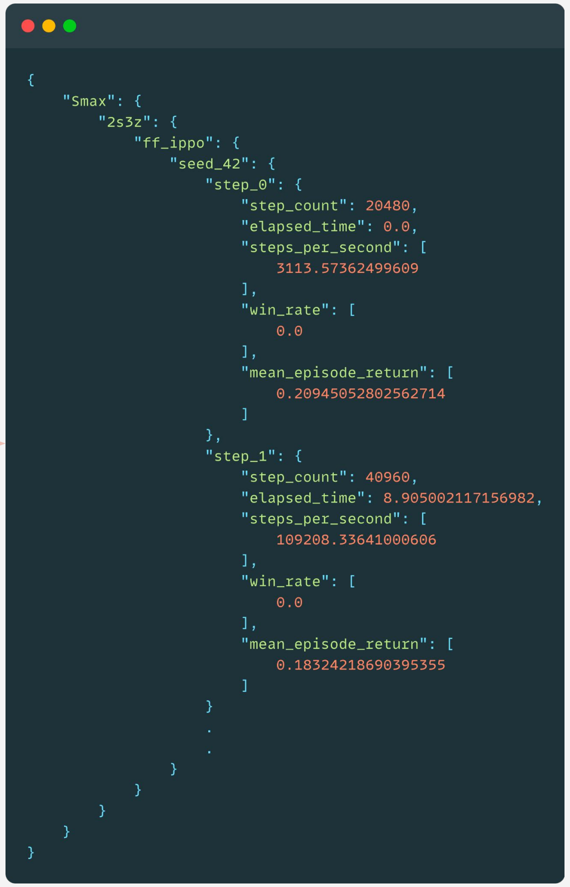


```
{
    "environment_name" : {
        "task_name" : {
            "algorithm_name": {
                "run_1": {
                    .
                    .
                    "step_k" : {
                        "step_count": <int>,
                        "metric_1": [<num_eval_episodes>],
                        "metric_2": [<num_eval_episodes>],
                    }
                    "absolute_metrics": {
                        "metric_1": [<num_eval_episodes>*10],
                        "metric_2": [<num_eval_episodes>*10]
                    }

                }
                .
                .
                "run_n": {

                    .
```

JSONLogger to match standardised format



```
from marl_eval.json_tools import JsonLogger

json_logger = JsonLogger(
    path="experiment_results",
    algorithm_name="IPPO",
    environment_name="rware",
    task_name="tiny-4ag",
    seed=42,
)
```



```
{
    "Smax": {
        "2s3z": {
            "ff_ippo": {
                "seed_42": {
                    "step_0": {
                        "step_count": 20480,
                        "elapsed_time": 0.0,
                        "steps_per_second": [
                            3113.57362499609
                        ],
                        "win_rate": [
                            0.0
                        ],
                        "mean_episode_return": [
                            0.20945052802562714
                        ]
                    },
                    "step_1": {
                        "step_count": 40960,
                        "elapsed_time": 8.905002117156982,
                        "steps_per_second": [
                            109208.33641000606
                        ],
                        "win_rate": [
                            0.0
                        ],
                        "mean_episode_return": [
                            0.18324218690395355
                        ]
                    }
                    .
                    .
                }
            }
        }
    }
}
```

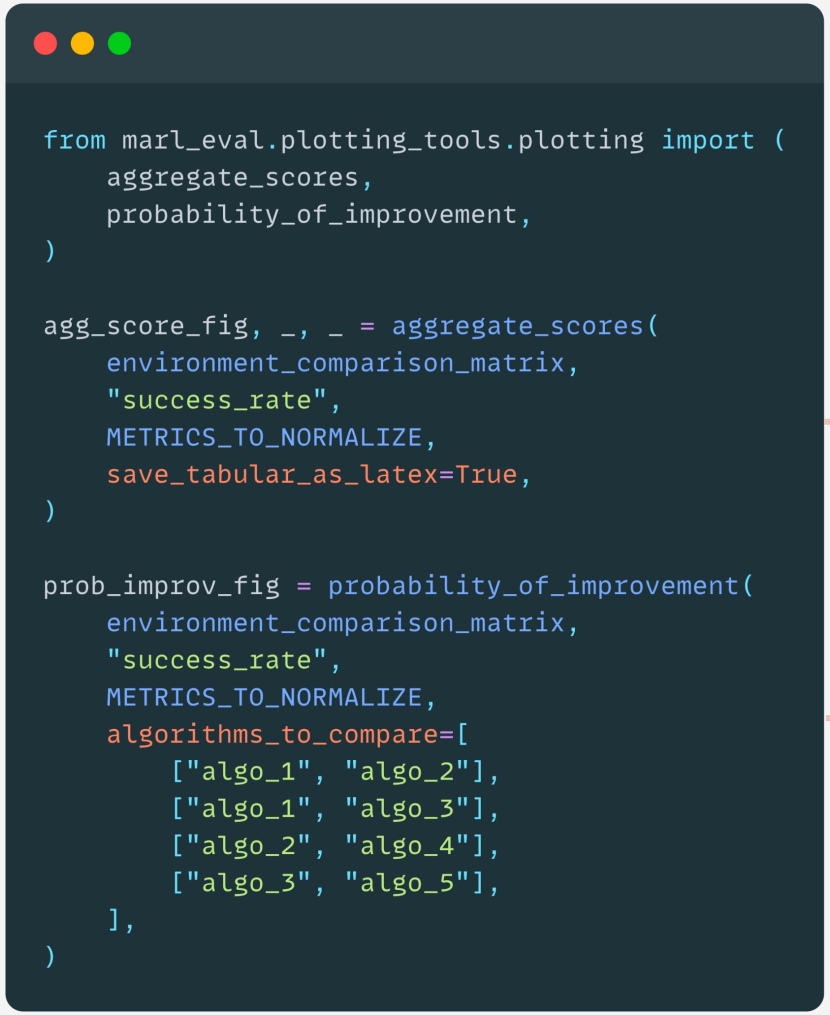
Gorsane et al., *Towards a Standardised Performance Evaluation Protocol for Cooperative MARL*, 36th Conference on Neural Information Processing Systems (NeurIPS 2022)

Bettini et al. BenchMARL: Benchmarking Multi-Agent Reinforcement Learning (2023)

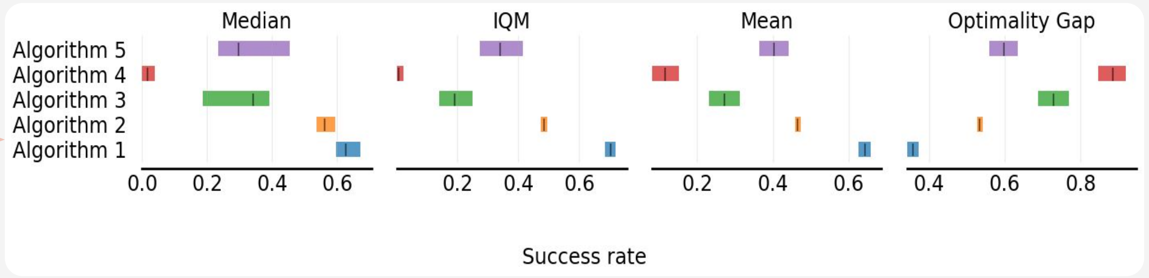
bit.ly/
id-marleval

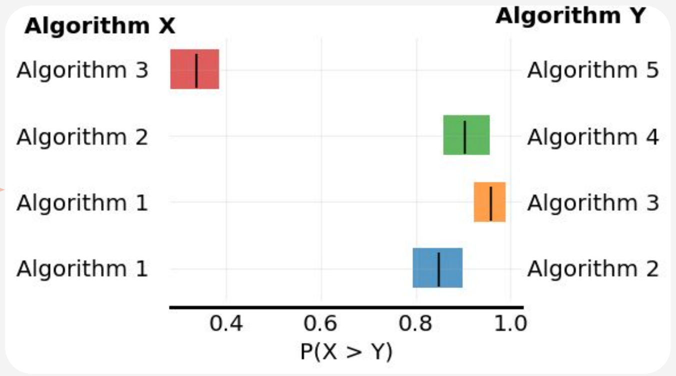
MARL-eval

Statistically robust experiment result aggregation



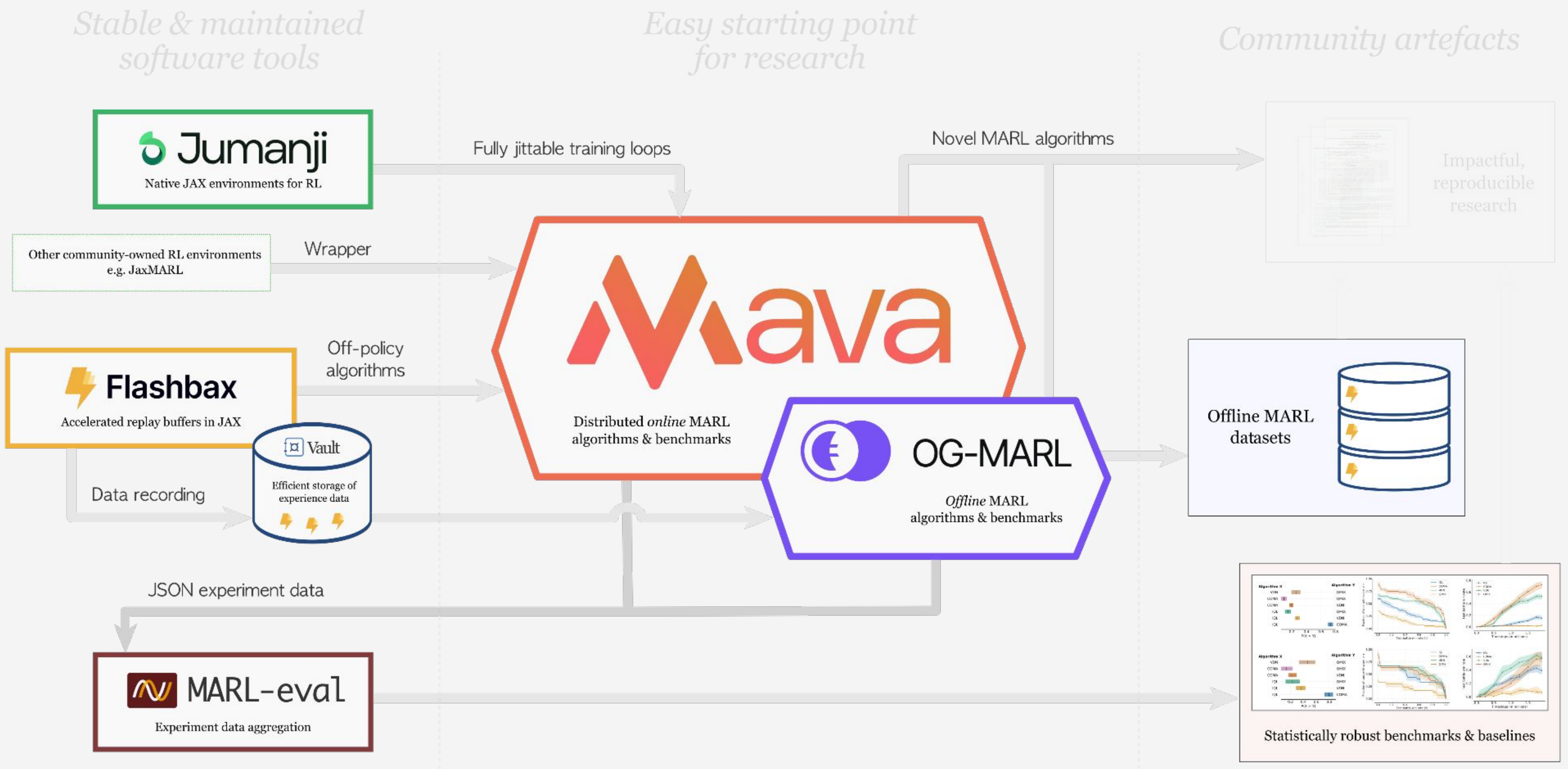
```python
from marl_eval.plotting_tools.plotting import (
    aggregate_scores,
    probability_of_improvement,
)

agg_score_fig, _, _ = aggregate_scores(
    environment_comparison_matrix,
    "success_rate",
    METRICS_TO_NORMALIZE,
    save_tabular_as_latex=True,
)

prob_improv_fig = probability_of_improvement(
    environment_comparison_matrix,
    "success_rate",
    METRICS_TO_NORMALIZE,
    algorithms_to_compare=[
        ["algo_1", "algo_2"],
        ["algo_1", "algo_3"],
        ["algo_2", "algo_4"],
        ["algo_3", "algo_5"],
    ],
)
```



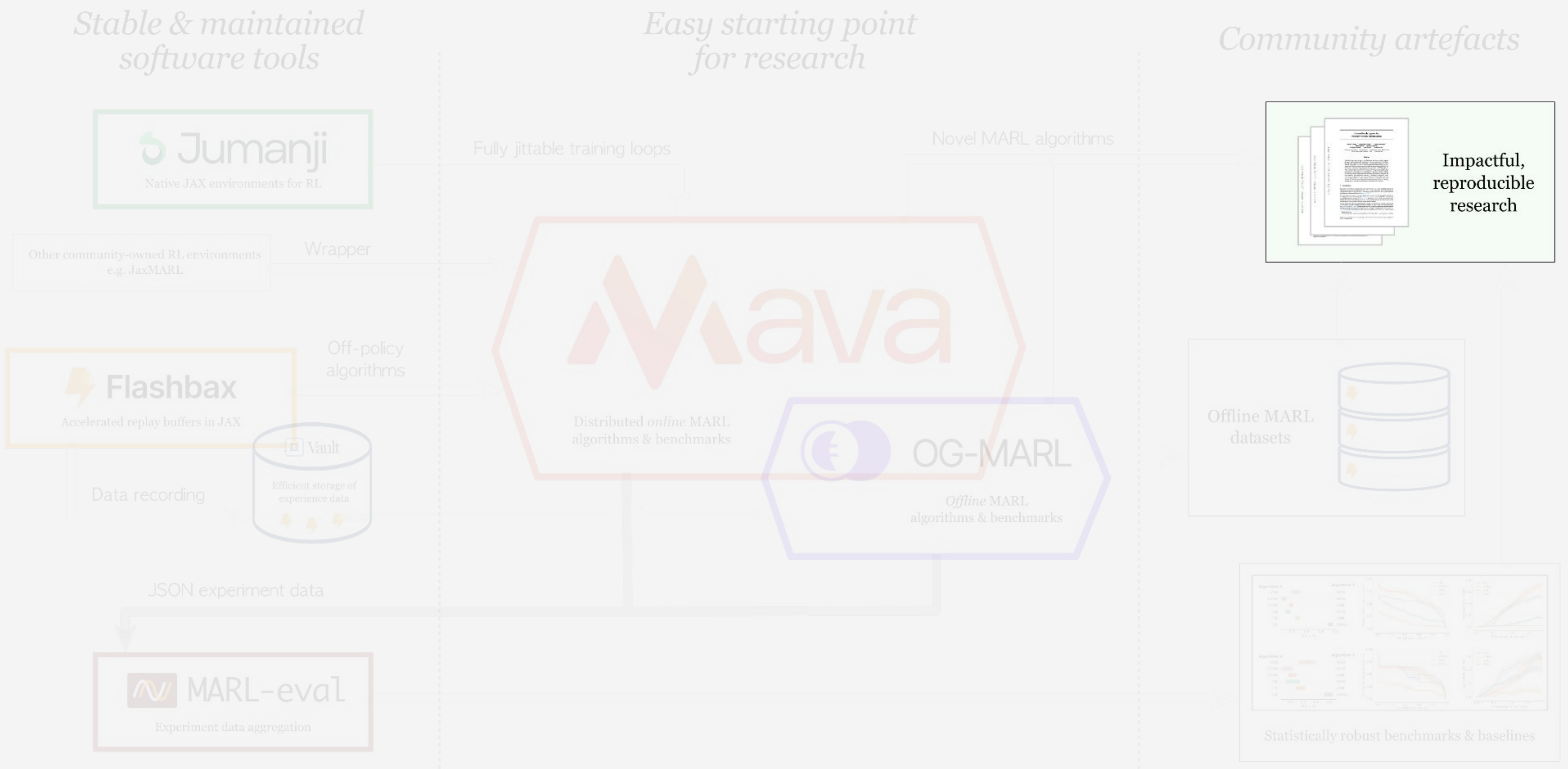


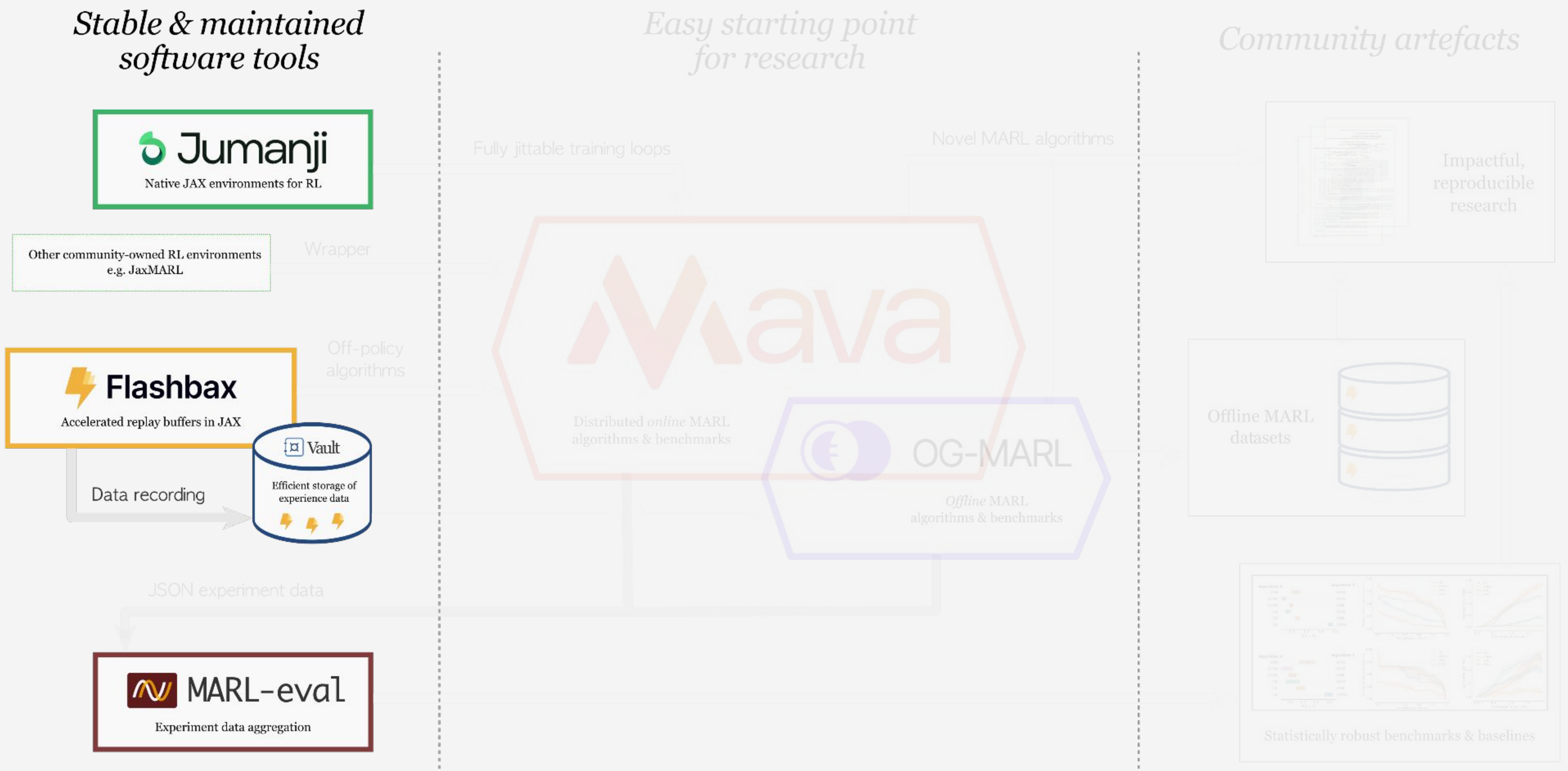
bit.ly/
id-marleval

Gorsane et al., *Towards a Standardised Performance Evaluation Protocol for Cooperative MARL*, 36th Conference on Neural Information Processing Systems (NeurIPS 2022)


Stable & maintained software tools
Easy starting point for research
Community artefacts
Jumanji
Native JAX environments for RL
Fully jittable training loops
Novel MARL algorithms
Impactful, reproducible research
Other community-owned RL environments e.g. JaxMARL
Wrapper
Mava
Distributed *online* MARL algorithms & benchmarks
Flashbax
Accelerated replay buffers in JAX
Off-policy algorithms
Vault
Efficient storage of experience data
Data recording
OG-MARL
Offline MARL algorithms & benchmarks
Offline MARL datasets
JSON experiment data
MARL-eval
Experiment data aggregation
Statistically robust benchmarks & baselines

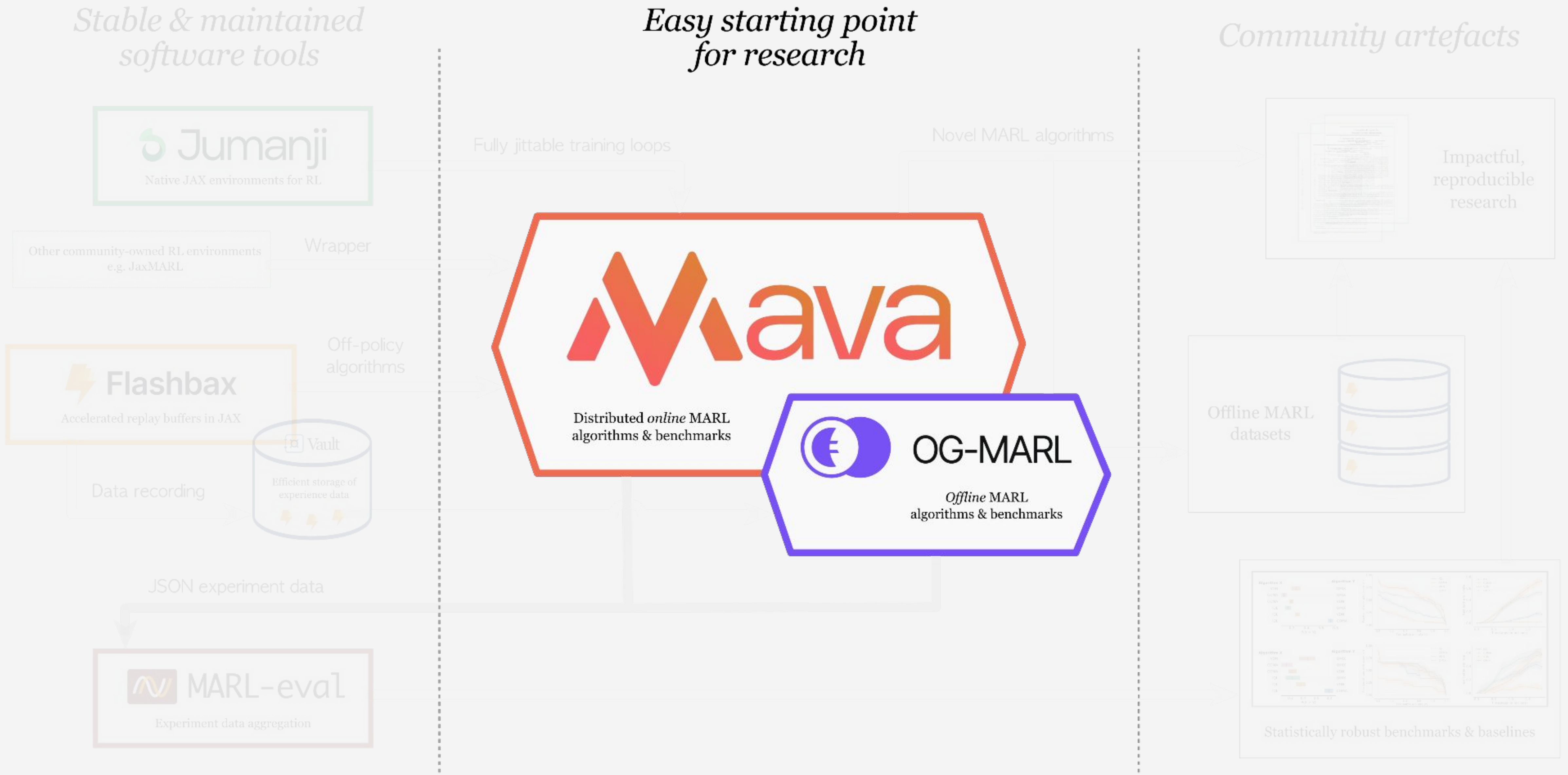
InstaDeep™


Stable & maintained software tools
Easy starting point for research
Community artefacts
Jumanji
Native JAX environments for RL
Fully jittable training loops
Novel MARL algorithms
Impactful, reproducible research
Other community-owned RL environments e.g. JaxMARL
Wrapper
Mava
Off-policy algorithms
Flashbax
Accelerated replay buffers in JAX
Vault
Efficient storage of experience data
Data recording
Distributed *online* MARL algorithms & benchmarks
OG-MARL
Offline MARL algorithms & benchmarks
Offline MARL datasets
JSON experiment data
MARL-eval
Experiment data aggregation
Statistically robust benchmarks & baselines


Stable & maintained software tools
Easy starting point for research
Community artefacts
Jumanji
Native JAX environments for RL
Fully jittable training loops
Novel MARL algorithms
Impactful, reproducible research
Other community-owned RL environments e.g. JaxMARL
Wrapper
Mava
Off-policy algorithms
Flashbax
Accelerated replay buffers in JAX
Distributed online MARL algorithms & benchmarks
Offline MARL datasets
Vault
Efficient storage of experience data
Data recording
OG-MARL
Offline MARL algorithms & benchmarks
JSON experiment data
MARL-eval
Experiment data aggregation
Statistically robust benchmarks & baselines

InstaDeep™

Easy starting point for research

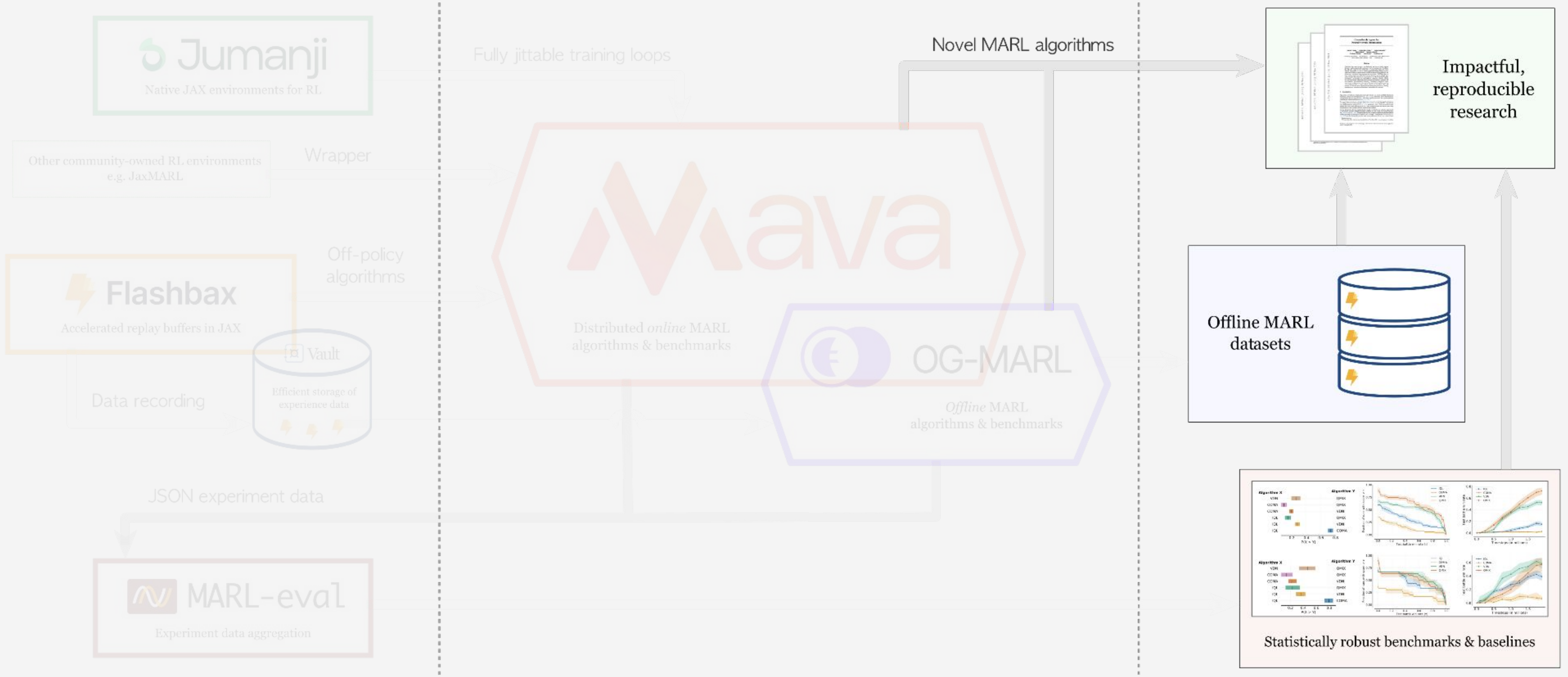

Stable & maintained software tools
Community artefacts
Jumanji
Native JAX environments for RL
Other community-owned RL environments e.g. JaxMARL
Wrapper
Fully jittable training loops
Novel MARL algorithms
Impactful, reproducible research
Flashbax
Accelerated replay buffers in JAX
Off-policy algorithms
Vault
Efficient storage of experience data
Data recording
Mava
Distributed *online* MARL algorithms & benchmarks
OG-MARL
Offline MARL algorithms & benchmarks
Offline MARL datasets
JSON experiment data
MARL-eval
Experiment data aggregation
Statistically robust benchmarks & baselines

InstaDeep™

Stable & maintained software tools

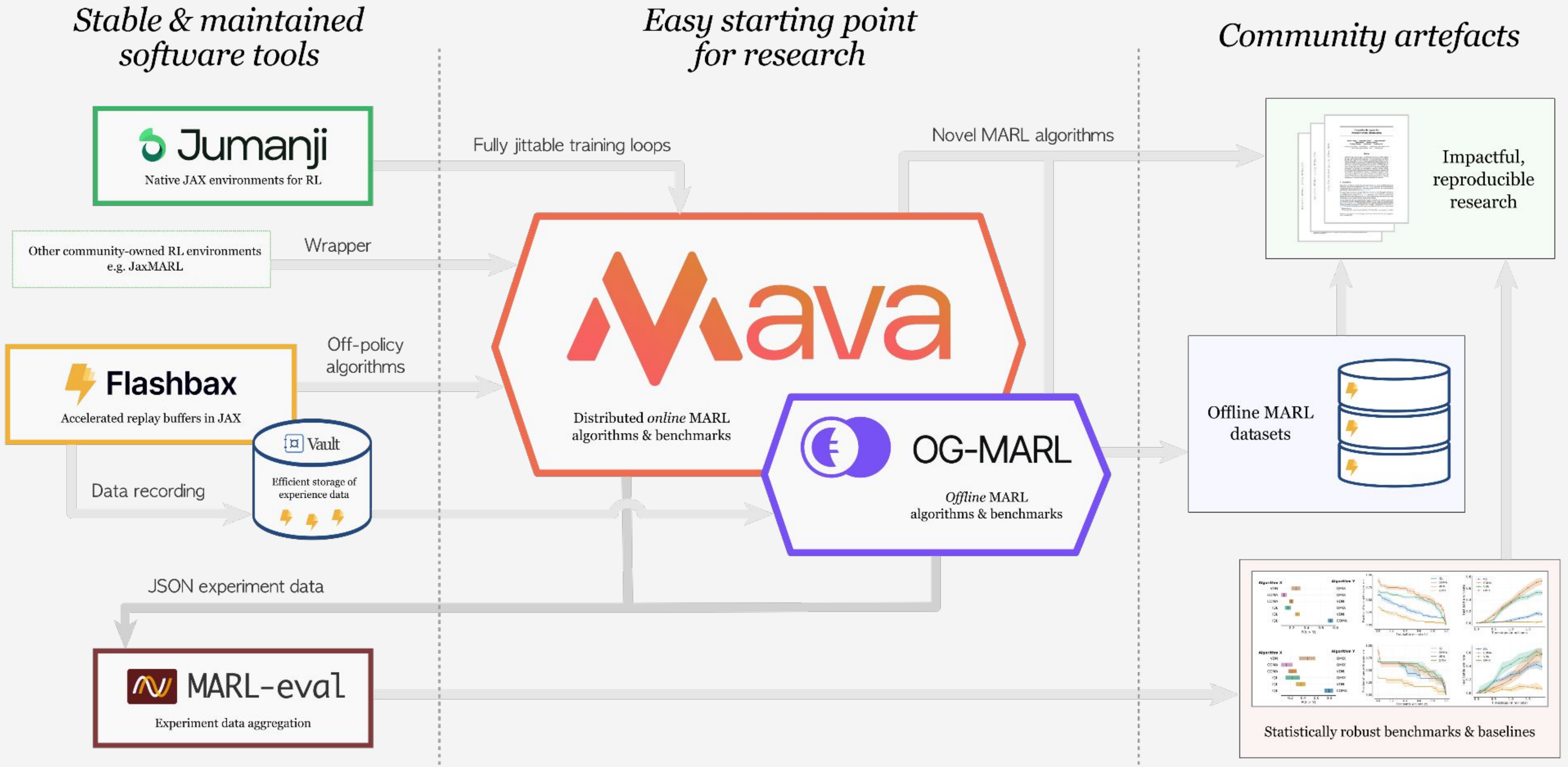
Easy starting point for research

Community artefacts



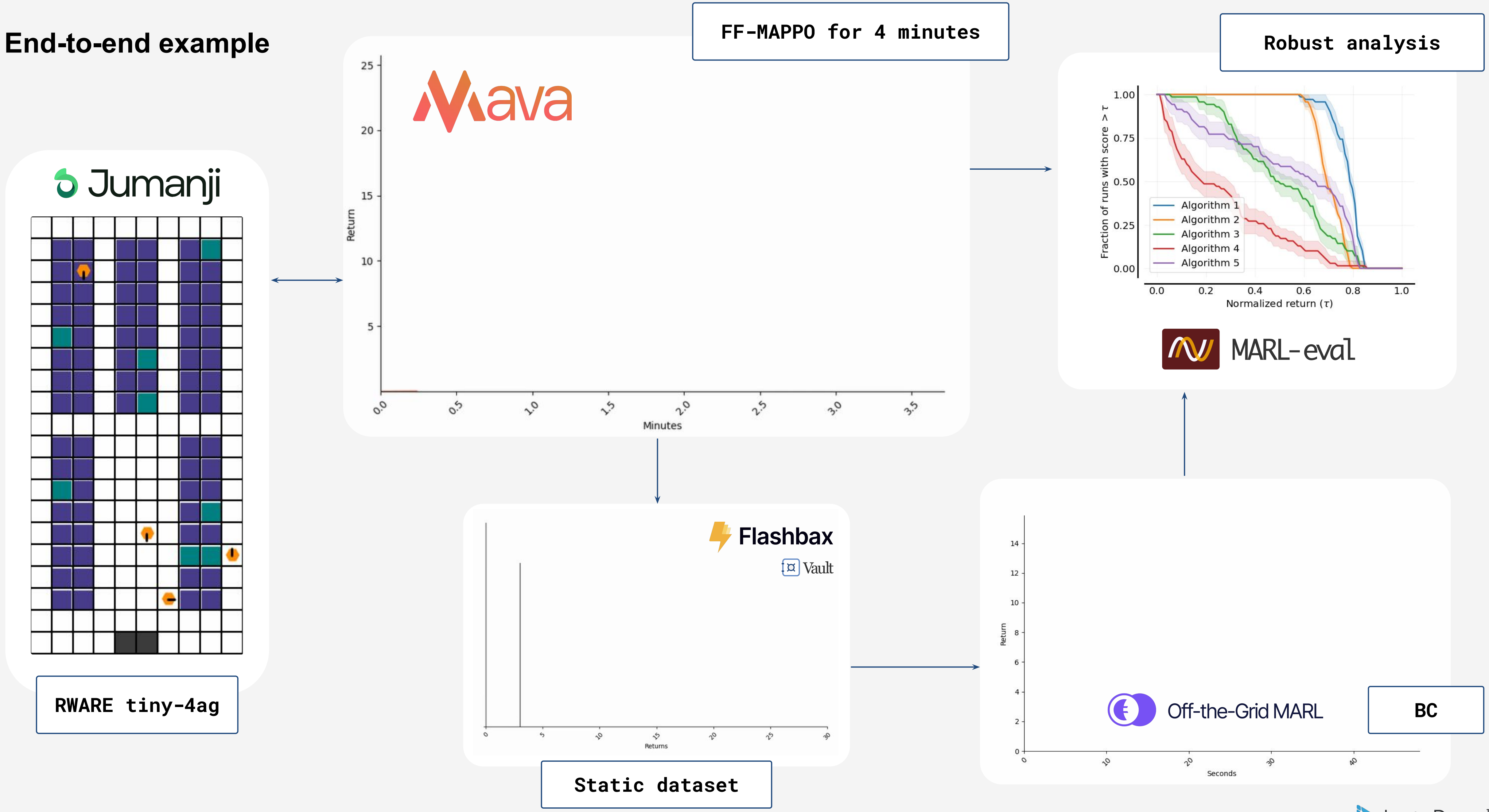
Jumanji
Native JAX environments for RL
Fully jittable training loops
Other community-owned RL environments e.g. JaxMARL
Wrapper
Flashbax
Accelerated replay buffers in JAX
Off-policy algorithms
Vault
Efficient storage of experience data
Data recording
Mava
Distributed *online* MARL algorithms & benchmarks
OG-MARL
Offline MARL algorithms & benchmarks
JSON experiment data
MARL-eval
Experiment data aggregation
Novel MARL algorithms
Impactful, reproducible research
Offline MARL datasets
Statistically robust benchmarks & baselines

InstaDeep™


Stable & maintained software tools
Easy starting point for research
Community artefacts
Jumanji
Native JAX environments for RL
Fully jittable training loops
Novel MARL algorithms
Impactful, reproducible research
Other community-owned RL environments e.g. JaxMARL
Wrapper
Mava
Flashbax
Accelerated replay buffers in JAX
Off-policy algorithms
Distributed online MARL algorithms & benchmarks
Vault
Efficient storage of experience data
Data recording
OG-MARL
Offline MARL algorithms & benchmarks
Offline MARL datasets
JSON experiment data
MARL-eval
Experiment data aggregation
Statistically robust benchmarks & baselines

InstaDeep™

End-to-end example



InstaDeep™

InstaDeep™

Questions?

bit.ly/
id-jumanji

bit.ly/
id-flashbax

bit.ly/
id-mava

bit.ly/
id-ogmarl

bit.ly/
id-marleval