

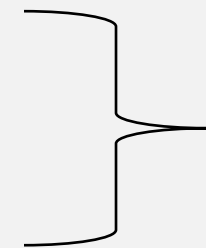
# Modernizacja aplikacji z Mono2Micro

Mikołaj Jaworski  
Partner Technical Specialist @ IBM



# Jak narzędzia modernizacyjne mogą pomóc?

Modernizacja architektury



IBM Mono2Micro

Planowanie  
(rady i rekomendacje)

Jak może wyglądać aplikacja:

- Dekompozycja aplikacji w ramach granic funkcjonalności i/lub potrzeb biznesowych
- Przekształcenie w zbiór mikroservisów.

Działanie  
(wsparcie w modernizacji)

- Generowanie kodu umożliwiającego komunikację pomiędzy mikroservisami Liberty.

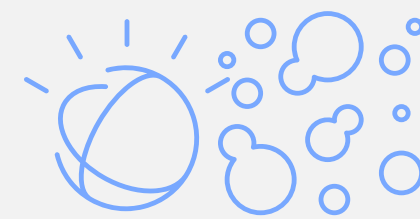
# IBM Mono2Micro

Zapewnia możliwość maksymalnego wykorzystania istniejącego kodu Java przy jednoczesnej transformacji aplikacji na mikroustugi.

Monolith



Silnik analizujący istniejące aplikacje.



Sztuczna inteligencja identyfikuje komponenty o wysokiej spójności i łatwym połączeniu.

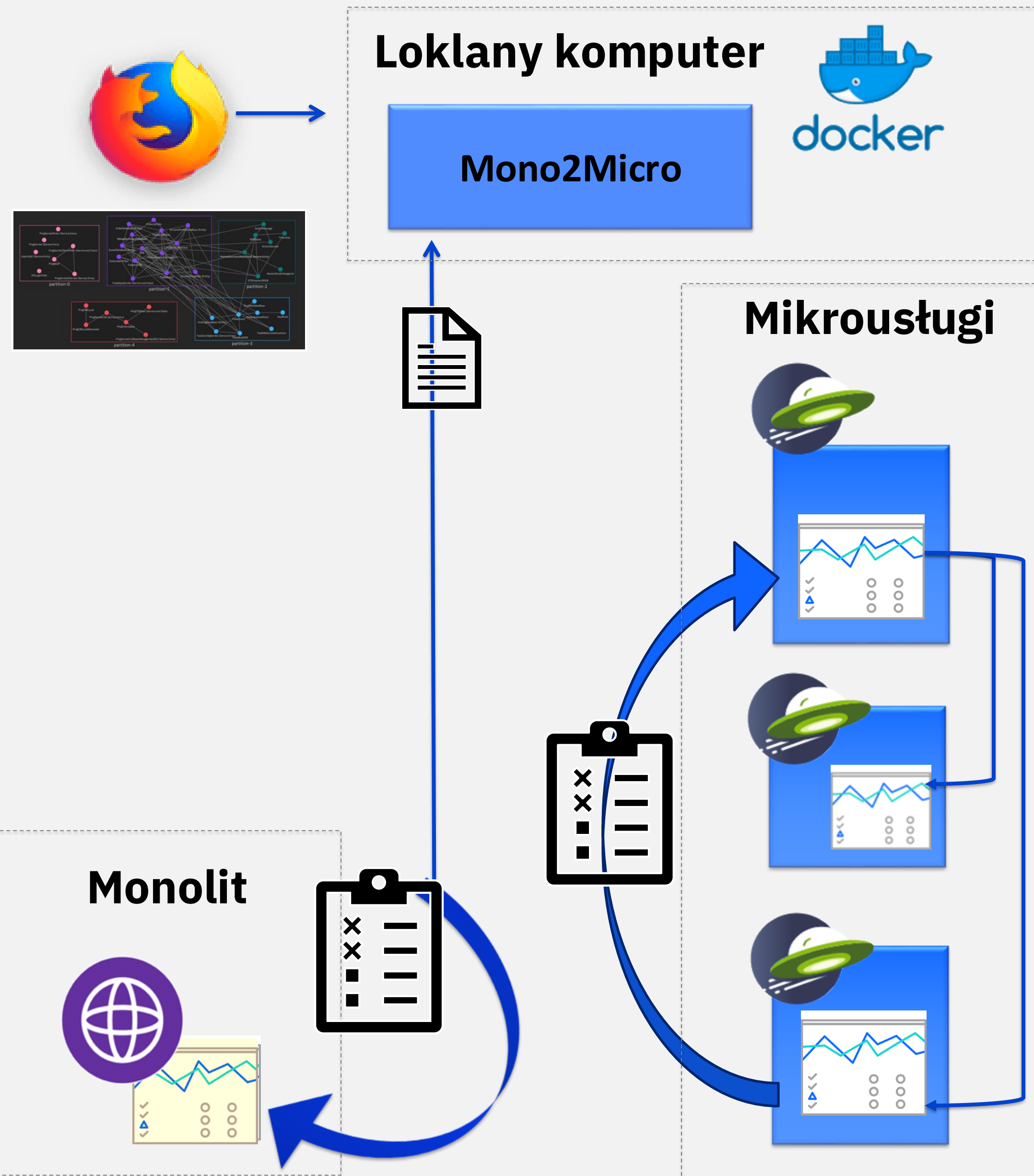
Microservices



Generowanie startowego kodu mikroustug do wdrożenia w Liberty.

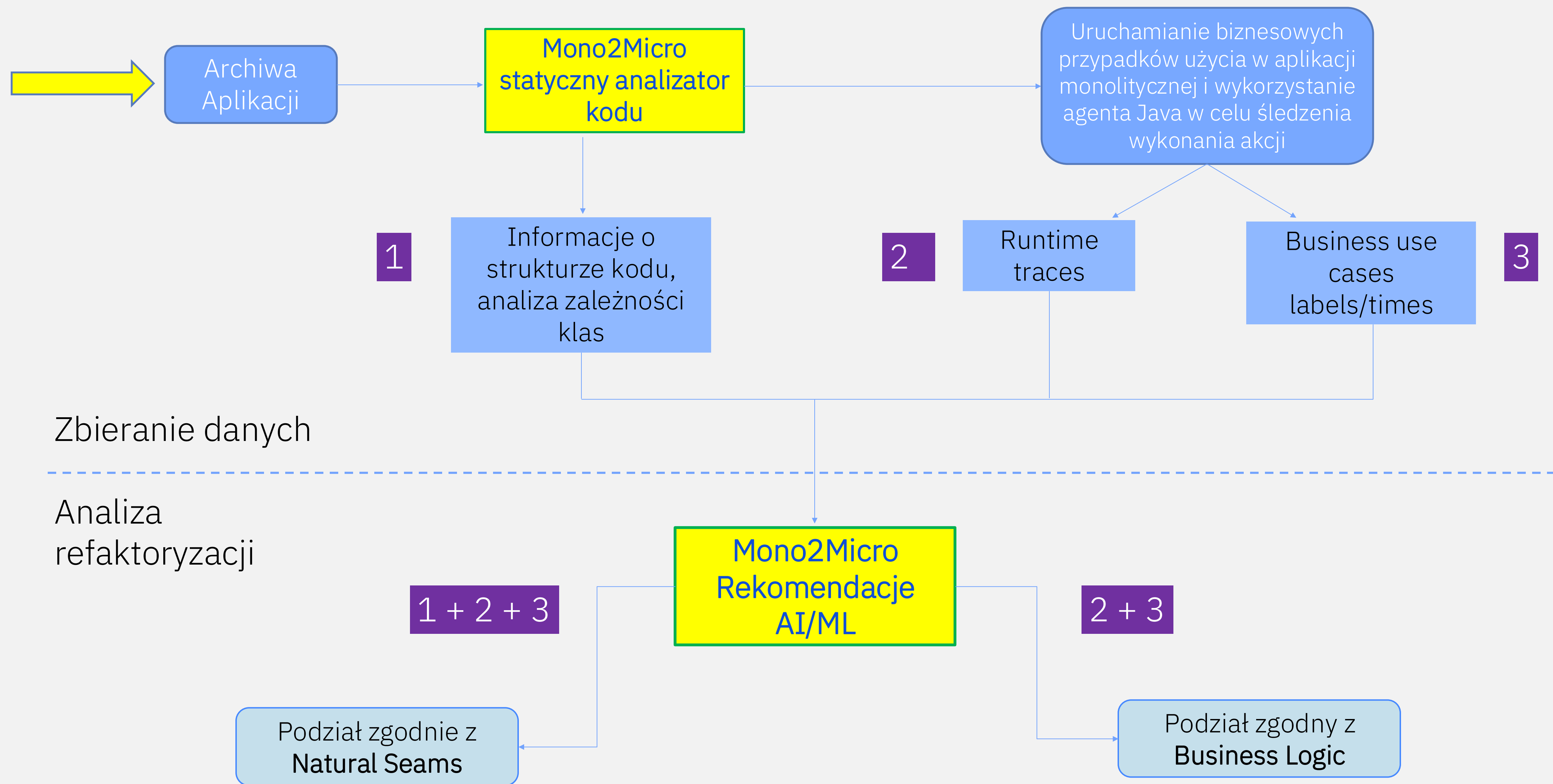
Transformacja wspomagana sztuczną inteligencją przy zmniejszonym ryzyku i niższych kosztach.

# Jak IBM Mono2Micro działa?



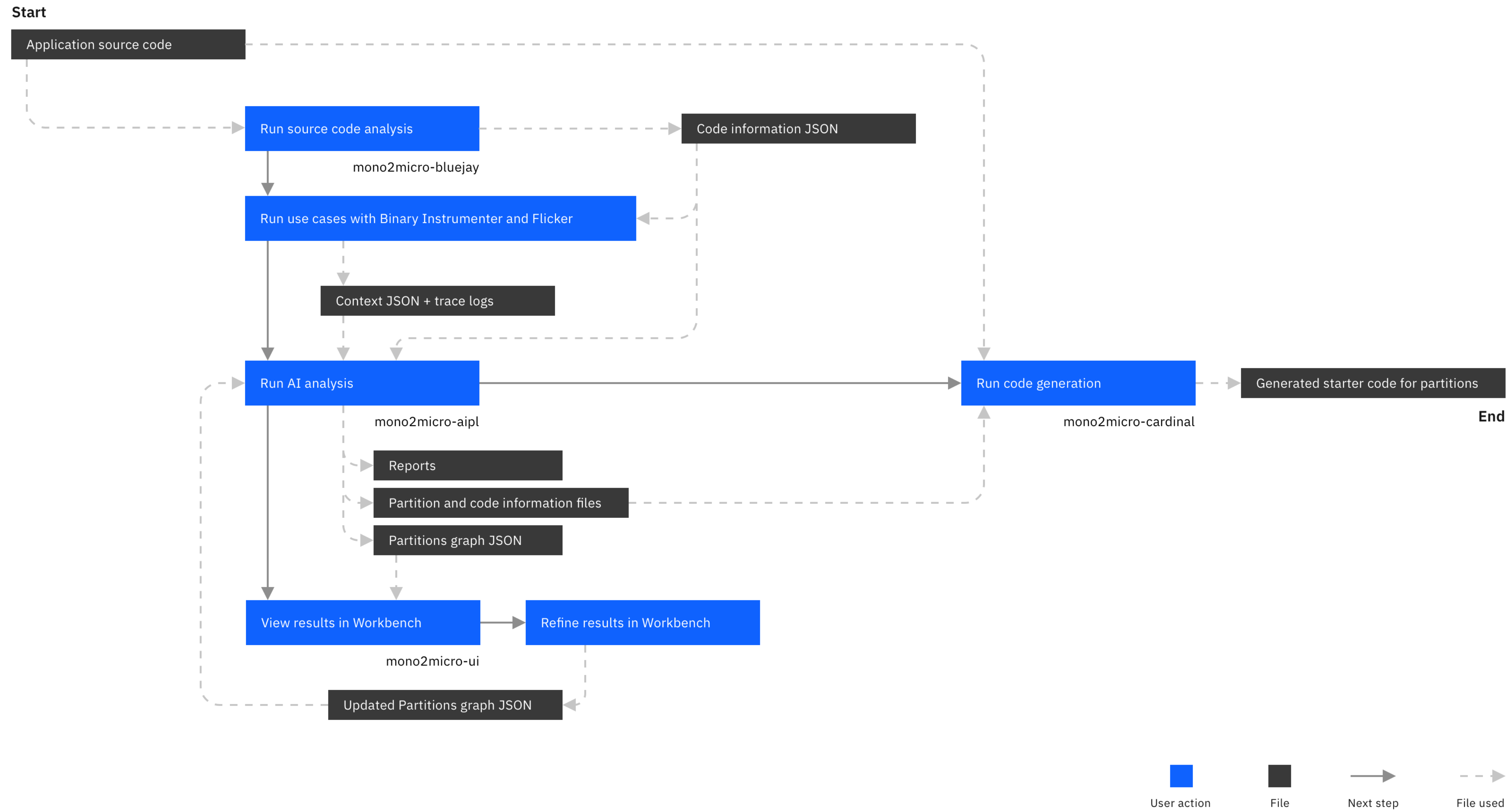
1. Instalacja Mono2Micro.
2. Statyczna analiza i opracowanie kodu źródłowego monolitu przez Mono2Micro.
3. Dynamiczna analiza wykonania testów biznesowych w odniesieniu do aplikacji monolitycznej, rejestrowana przez agenta Java.
4. Analiza powstałych plików przez AI Mono2Micro, aby zarekomendować podział na mikroustugi.
5. Przegląd i kustomizacja zalecanych partycji z poziomu interfejsu użytkownika w przeglądarce.
6. Mono2Micro tworzy nowe partycje mikroustug i generuje kod umożliwiający komunikację między partycjami.
7. Testy tych samych przypadki biznesowych na nowych partycjach mikroustug w Liberty.

# Mono2Micro – Refaktoryzacja aplikacji poprzez analizę statyczną i środowiska uruchomieniowego





# Przeptyw użycia Mono2Micro



# IBM Mono2Micro – Rekomendacje mikrouslug

## Wejście:

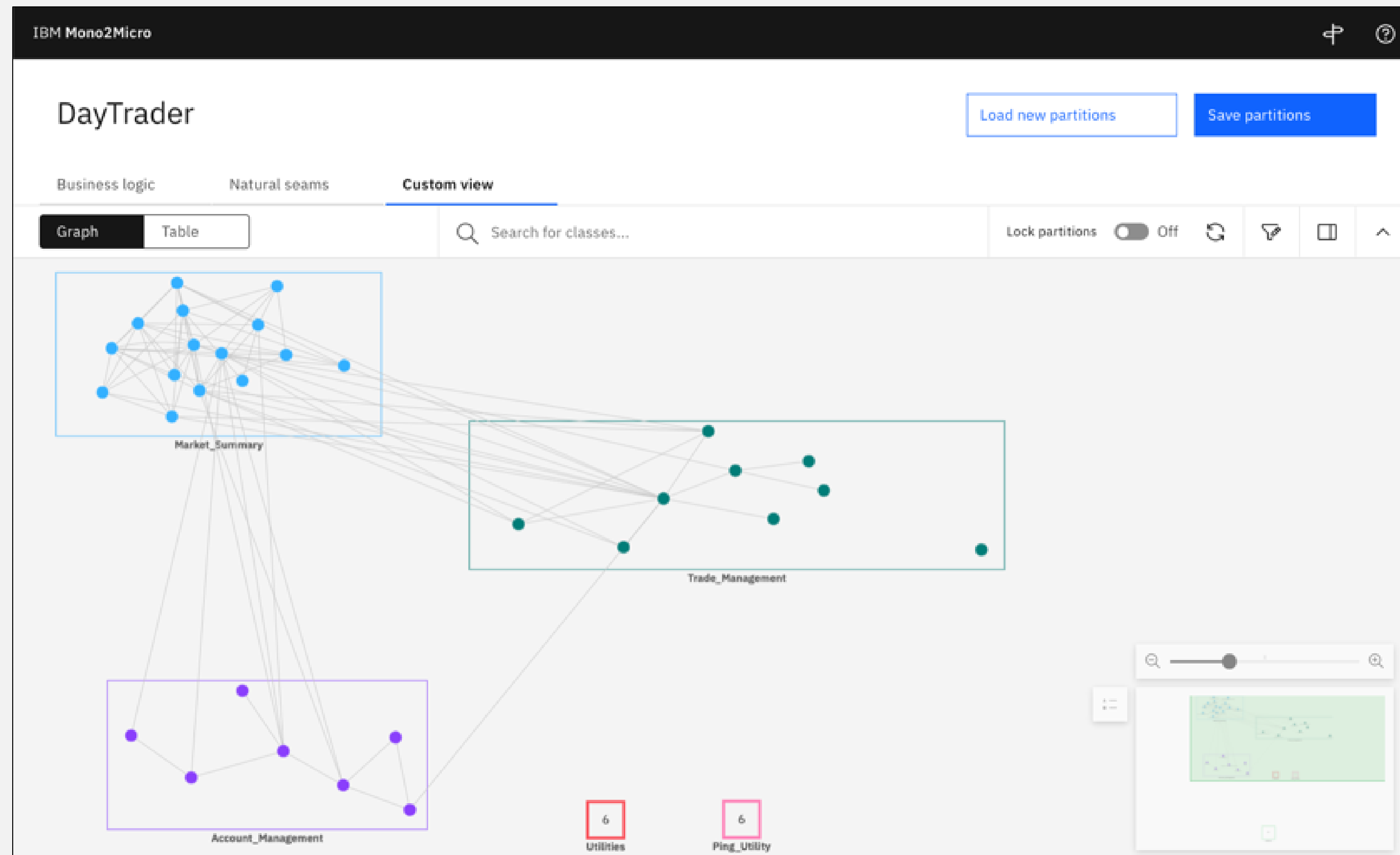
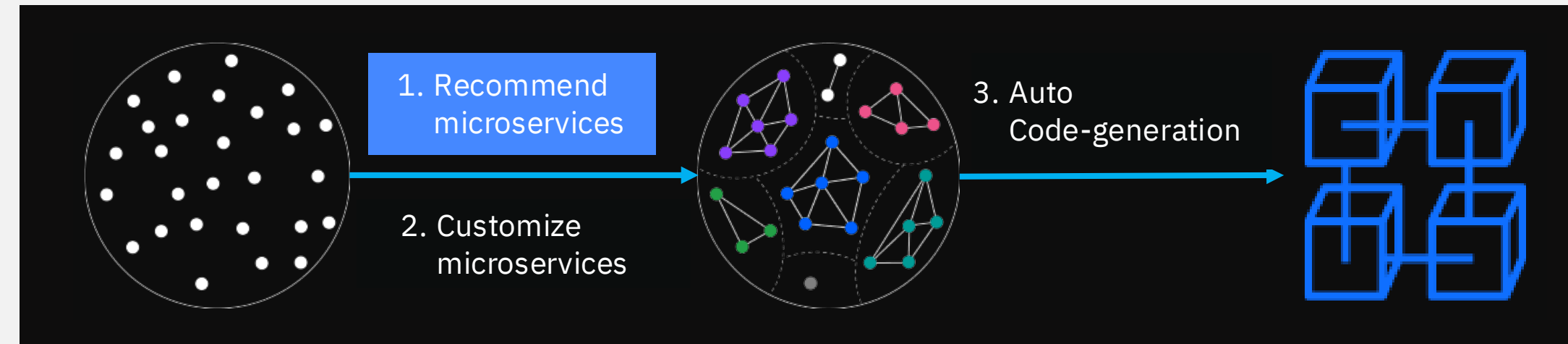
Runtime Traces,  
Use Cases, Code  
Metadata

Machine Learning &  
Clustering

## Wyjście:

Business Logic Seams

mono2micro-aip1 – oparty o  
AI generator rekomendacji  
dotyczących partycji, które  
ostatecznie mogą stać się  
mikrouslugami.



**Nodes:** Partitions / Groups of Classes - **Edges:** Call Volumes

# Automatyczne generowanie kodu

## Wejście



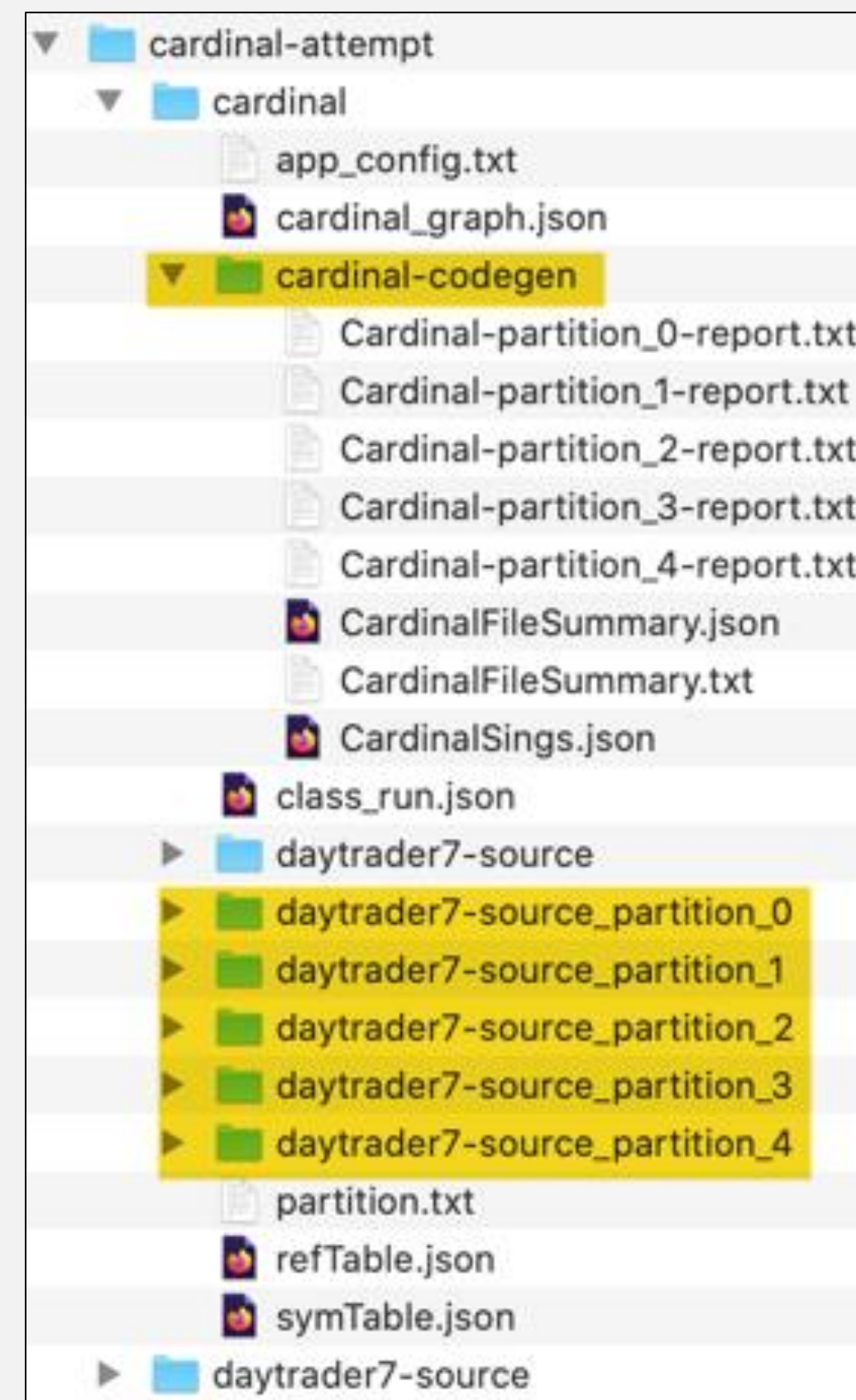
Recommended Microservice  
(Business Logic)

+

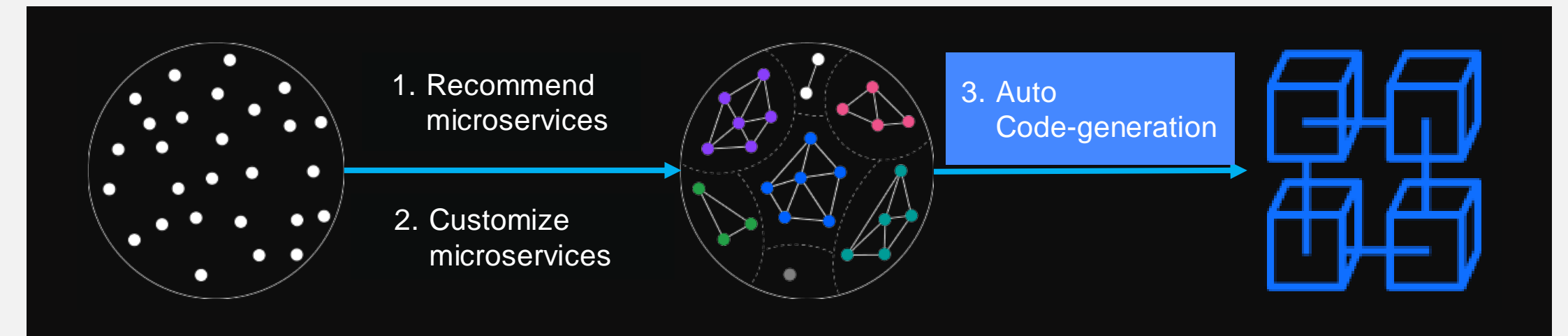
Original Source Code

mono2micro-cardinal – oparty o AI generator kodu, który posiada głęboką wiedzę na temat semantyki języka programowania Java.

## Wyjście



Codebase for each microservice



## Przykładowy automatycznie wygenerowany kod dla mikroustługi

```
. . .
public Collection<QuoteDataBean> getTopGainers() {

    Form form = new Form();
    form.param("referenceId",
String.valueOf(this.referenceId));

    JsonObject response_json =
client.target(DAYTRADER_BIZLOGIC_REST_URL)
        .path("getTopGainers")
        .request(MediaType.TEXT_PLAIN)
        .post(Entity.entity(form,
MediaType.APPLICATION_FORM_URLENCODED_TYPE), String.class);
    String response =
(String)response_json.get("return_value");

    Collection<String> response_decoded =
(Collection<String>)SerializationUtil.decode(response);
    String ret_type = (String)response.get("return_type");
. . .
```



# Transformacja monolitów w mikrouługi

## Analiza Manualna

- Trudny, kosztowny i czasochłonny proces
- Wymaga wysokiego poziomu umiejętności
- Wymaga dogłębnej znajomości istniejącego kodu
  - nie zawsze dostępna
  - twórcy aplikacji mogli odejść
- Znaczna szansa na popełnienie błędu
  - może być konieczna rearchitektura i ponowne wdrożenie
  - sprawdzany metodą „prób i błędów”

## Mono2Micro - Analiza oparta na AI

- Automatyczna analiza statyczna i środowiska uruchomieniowego
- Brak konieczności posiadania specjalnych umiejętności
- Dogłębna znajomość kodu nie jest wymagana
- ML jest kluczem do uzyskania właściwych rekomendacji dotyczących mikrouług
  - ML emuluje wiedzę SME
  - Refaktoryzacja, ale nie rearchitektura i ponowna implementacja
- Mono2Micro pomaga architektom w prawidłowym partycjonowaniu klas, gdy tylko jest to możliwe, ze świadomością funkcji biznesowych.

DEMO - [Mono2Micro](#)

