



Projekt współfinansowany ze środków Unii Europejskiej w ramach Europejskiego Funduszu Społecznego

Rozproszone systemy internetowe

Enterprise Java Beans: przykłady.

„Podniesienie potencjału uczelni wyższych jako czynnik rozwoju gospodarki opartej na wiedzy”

Nr projektu: UDA-POKL.04.01.01-00-143/09-00

Komponent EJB i jego otoczenie

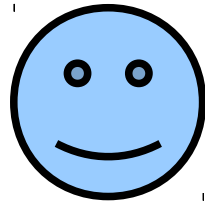
Kontener:

zarządzanie instancjami komponentu

Komponent EJB

Logika biznesowa
i/lub dane

Współpraca z klientem
(mediacja dostępu,
niezależność od platformy
serwera, redystrybucja bez
potrzeby ponownej kompilacji,
koncepcja „client view”)



Dostrajanie poprzez
specyfikację ustawień
środowiskowych

Realizacja mechanizmów
bezpieczeństwa i transakcji
(metadane w kodzie lub deskryptor)

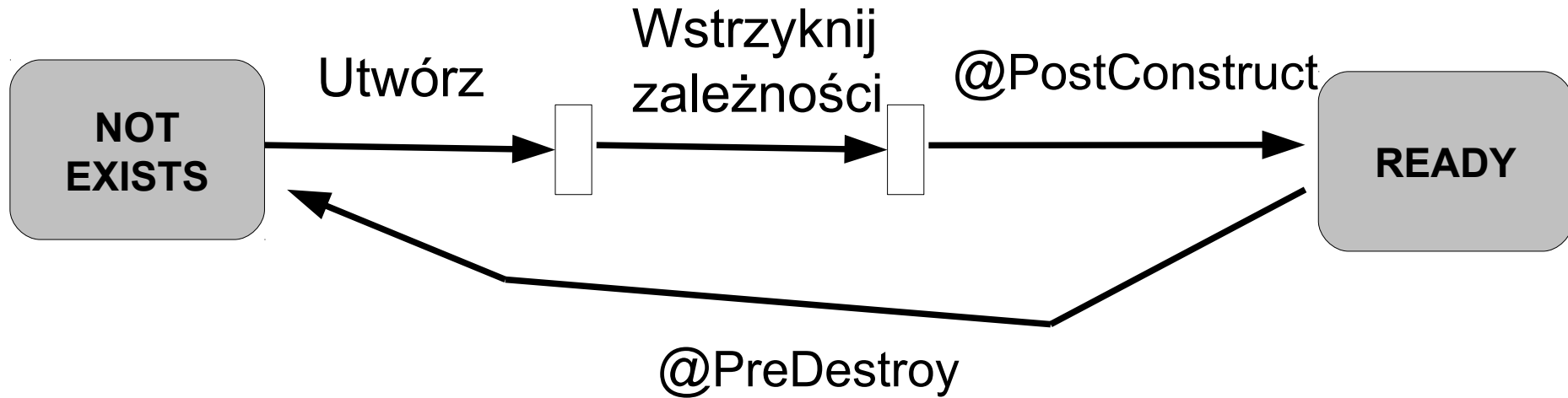
Typy obiektów

- **Session:** obsługuje pojedynczego klienta (sesja), zakładany krótki czas życia, nie reprezentuje danych, może korzystać z systemu transakcji, stanowe i bezstanowe
- **Message-Driven:** wykonanie operacji jest reakcją na otrzymany komunikat, wywołania asynchroniczne, bezstanowe, nie reprezentuje danych, może korzystać z systemu transakcji, zakładany krótki czas życia
- **Entity:** obiektowa reprezentacja danych, zakładany długi czas życia

Interfejsy i klienci

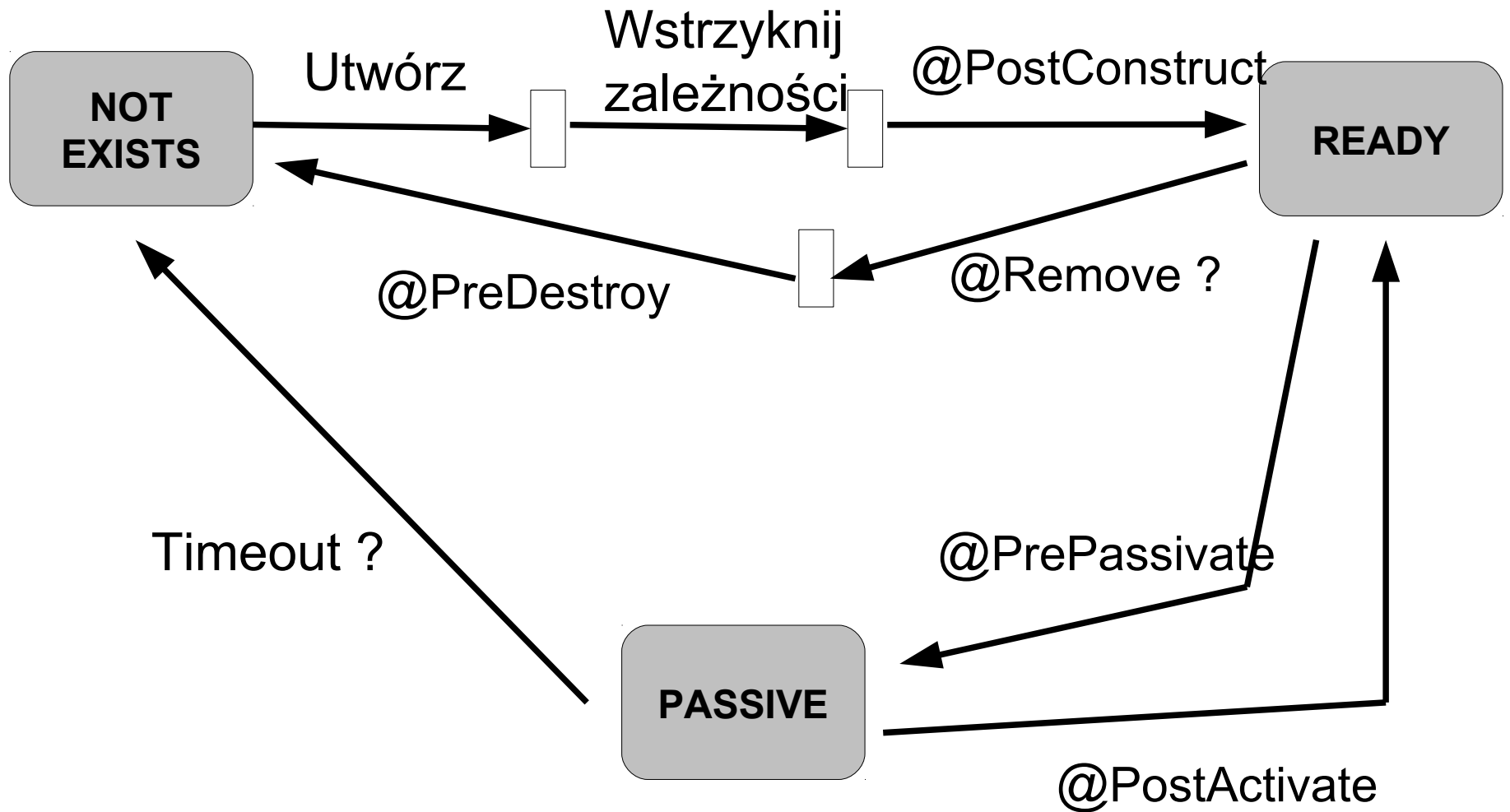
- Interfejs typu Local i Remote
 - przekazywanie parametrów
 - wady i zalety poszczególnych rozwiązań
- Pojęcie „*Client View*”
 - Local, Remote, Web Service
- Wstrzykiwanie zależności (dependency injection)
- Porównywanie obiektów (stateless vs stateful)

Stateless Session Bean

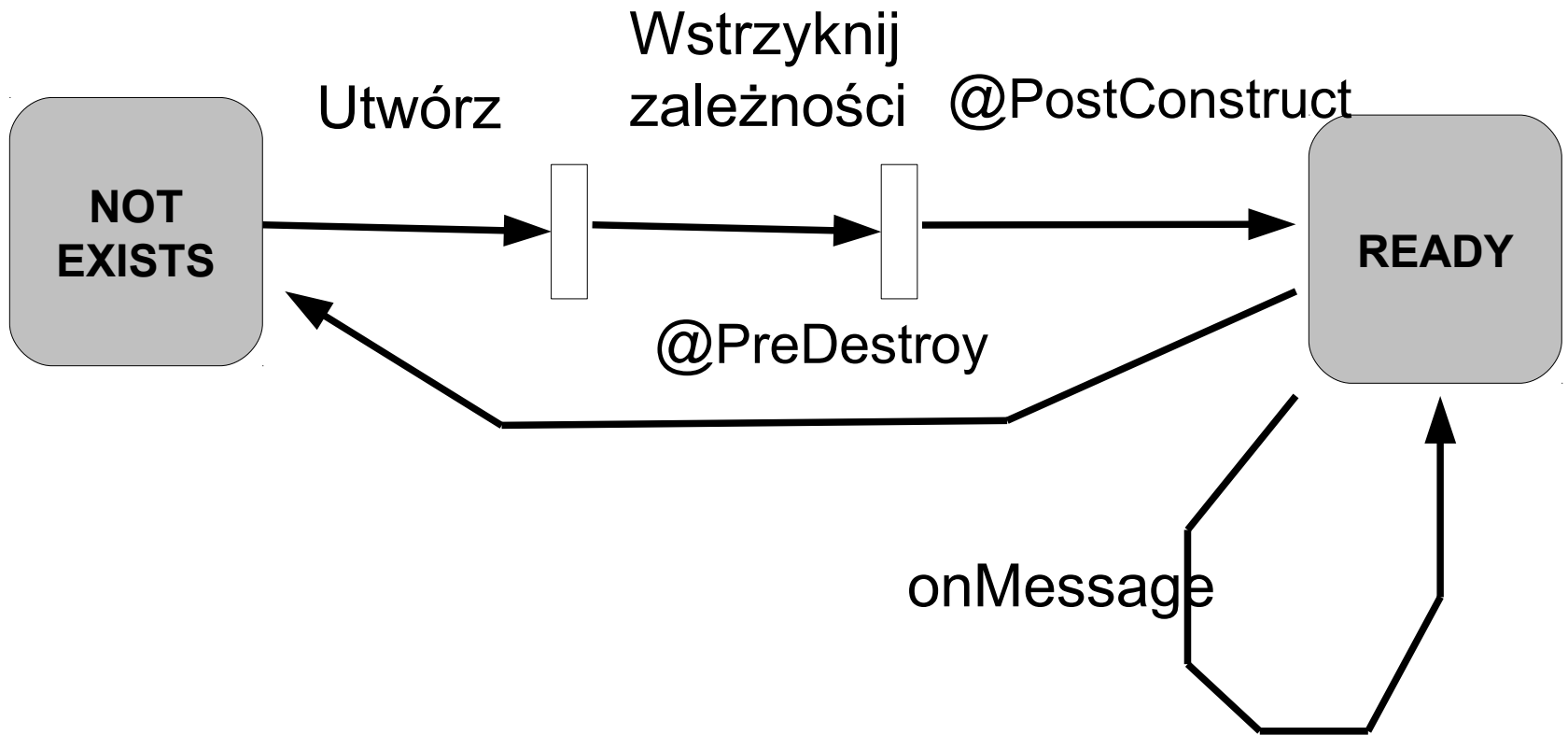


- Stateless = brak stanu dla konkretnego klienta
- Conversational Bean State
- Operacje: *passivate* i *activate*
- Timeout callbacks (tylko stateless)

Stateful Session Bean



Message-Driven Bean



Elementy dodatkowe

- **Interceptors:** metody uruchamiane przed metodami komponentu

```
@javax.ejb.AroundInvoke
```

```
public Object <name>(javax.ejb.InvocationContext ctx)
```

```
throws java.lang.Exception
```

```
@javax.ejb.Interceptors(X.class, Y.class ...)
```

- **Timer Service**

```
@Timeout
```

```
public void timeout(Timer timer)
```


EJB 3.1

- Opcjonalny interfejs dla postaci lokalnej
- Zdefiniowana składnia nazw obiektów
- Singleton Bean
- Rozszerzenia Timer Service
- Wywołania asynchroniczne (@Asynchronous)
- EJB 3.1 Lite