

# Informatyka w Mechatronice

## Serwer danych pomiarowych w technologii Java RMI

April 11, 2017

Dr inż. Lucjan Miękina

### Ćw. 3

Technologia Java RMI w zastosowaniu do serwera danych pomiarowych.

### Cel ćwiczenia

Celem ćwiczenia jest implementacja w języku Java klas klienta i serwera danych pomiarowych przesyłanych w systemie monitorowania, podobnie jak w poprzednim ćwiczeniu, ale **z użyciem technologii Java RMI** w miejsce protokołu UDP.

Całość aplikacji składa się z dwu modułów:

- serwera
- klienta.

### Funkcjonalność aplikacji

Należy stworzyć aplikację serwera danych pomiarowych, który udostępnia funkcje:

1. rejestracja ramki danych typu *TimeHistory* i *Spectrum* w pojemniku (liście lub mapie). Każdy typ danych będzie rejestrowany w oddzielnej liście lub mapie, a następnie może być zapisany w pliku w katalogu serwera, w sposób opisany w funkcji numer 3
2. zwrócenie numerowanej listy danych określonego typu, spełniających kryterium wyszukiwania oparte na nazwie urządzenia i/lub opisie kanału i/lub czasie rejestracji
3. zapis ramki danych odpowiadających wybranej pozycji z listy zwróconej przez funkcję numer 2 w pliku o nazwie opartej na typie danych, nazwie urządzenia, opisie kanału i czasie rejestracji, zawartych w podklasie *Packet*. Należy zastosować rozszerzenie *.thi* dla plików binarnych zawierających zserializowane dane *TimeHistory*, a rozszerzenie *.spc* dla danych typu *Spectrum*.
4. wyszukanie i przesłanie do klienta ramki danych odpowiadających wybranej pozycji z listy zwróconej przez funkcję numer 2
5. zastosowanie osobnych wątków dla realizacji w.w. zadań i objęcie sekcją krytyczną wykluczających się operacji

UWAGA: Wymagana jest realizacja punktu numer 1 dla uzyskania minimalnej oceny pozytywnej. Realizacja kolejnych punktów podnosi ocenę o 0.5 stopnia.

### Realizacja

W trakcie realizacji można się wzorować na rozwiązaniach zastosowanych w przykładzie aplikacji mini-czat w wersji Java RMI.

Jednak **odmiennie niż w owym przykładzie**, aplikacja budowana w ramach niniejszego ćwiczenia składa się z trzech projektów w środowisku Eclipse:

1. projekt *Interfejsy* zawiera jedynie deklaracje interfejsu serwera, przeznaczone do wykorzystania w kodzie serwera i klienta. Dlatego projekt *Interfejsy* musi być włączony w projekty *Serwer* i *Klient* za pomocą polecenia *Properties* → *Build path* → *Projects* → *Add*, zakończonego wskazaniem projektu *Interfejsy* w okienku dialogowym
2. projekt *Serwer* zawiera uniwersalną klasę serwera i dedykowaną klasę serwanta implementującego zdalny obiekt. Projekt jest zależny od projektu *Interfejsy*.
3. projekt *Klient* dostarcza możliwości wyszukania usługi na serwerze, pozyskania zdalnej referencji i wywołania za jej pośrednictwem poszczególnych funkcji serwera. Projekt jest zależny od projektu *Interfejsy*.

Należy użyć klas *Packet*, *Sequence*, *TimeHistory* i *Spectrum* z projektu *TypyDanych*, włączając go do projektów serwera i klienta, podobnie jak dla projektu *Interfejsy*.