

# Projekt: system do zarządzania oświetleniem

Zaawansowane Technologie  
Bazodanowe

2014/2015

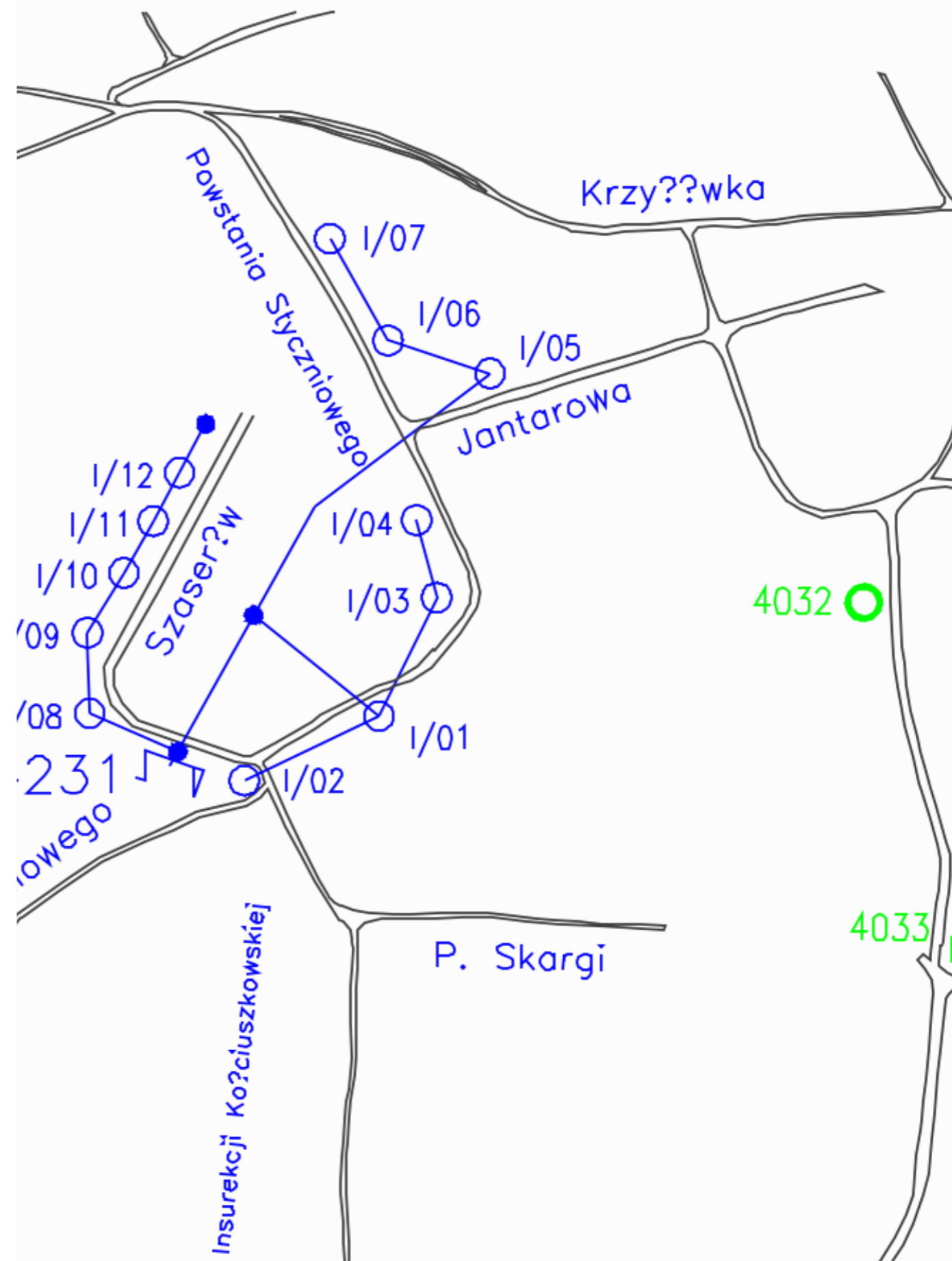
# Nowoczesne systemy oświetlenia ulicznego

- Zastosowanie LED: możliwość dowolnego sterowania – włączanie, wyłączenie, ściemnianie (*dimming*) bez ograniczeń np. lamp sodowych.
- Warstwa sensoryki/telemetrii: określa natężenie ruchu, wykrywa zdarzenia, itd.
- Komunikacja: możliwość sterowania każdą lampą osobno.

# Fizyczna struktura systemu

Struktura hierarchiczna:

- węzły zasilające – dla nas nieistotne,
- szafy – zasilanie + sterowanie, *segment controllers*,
- oprawy oświetleniowe – sterowniki lokalne, *drivers*.



# Przykład: Owlet Nightshift

The screenshot displays the Owlet monitoring interface. The browser address bar shows `plkrak1.myowlet.net`. The interface includes a navigation menu with 'Strona główna', 'Monitorowanie', 'Pulpit', and 'Informacja'. A language dropdown is set to 'Polski' and a 'Wyloguj' button is visible. The main content area features a map of Krakow with various network status indicators. On the left, there is a sidebar with 'Dane historyczne' and 'Status - wybór' sections. The 'Status - wybór' section contains several checkboxes for filtering data.

**Dane historyczne**

**Najnowsze**

22.10.2014

**Status - wybór**

- Wyświetl wszystkie
- Uszkodzony element świecący (obciążenie)
- Brakujące urządzenie
- Polecenie ręczne
- Serwis
- Pokaż SeCo
- Niekompletne urządzenia

**Ostatnia aktualizacja : 22.10.2014 11:39:43**

Map Satellite

PL\_KRK\_0010001\_991

Znacznik czasu	22.10.2014 10:55:18
Opis	SeCo - DIETLA

PL\_KRK\_0010001\_IV\_10

Import danych	22.10.2014 11:57:00
Opis	IV/10
Status sieci (FLN)	Brakujące
Status wyszukiwania danych (FDR)	O.K.
Komunikaty o błędach (FEM)	Oczekiwanie na dane
Poziom ściemniania (FDL)	30%
Ostatni czas włączenia (FON)	21.10.2014 17:47:09
Ostatni czas wyłączenia (FOF)	21.10.2014 07:00:59

PL\_KRK\_0010001\_IV\_16

Import danych	22.10.2014 11:57:00
Opis	IV/16
Status sieci (FLN)	Brakujące
Status wyszukiwania danych (FDR)	O.K.
Komunikaty o błędach (FEM)	Oczekiwanie na dane
Poziom ściemniania (FDL)	0%
Ostatni czas włączenia (FON)	21.10.2014 18:02:42
Ostatni czas wyłączenia (FOF)	14.04.2014 05:33:01

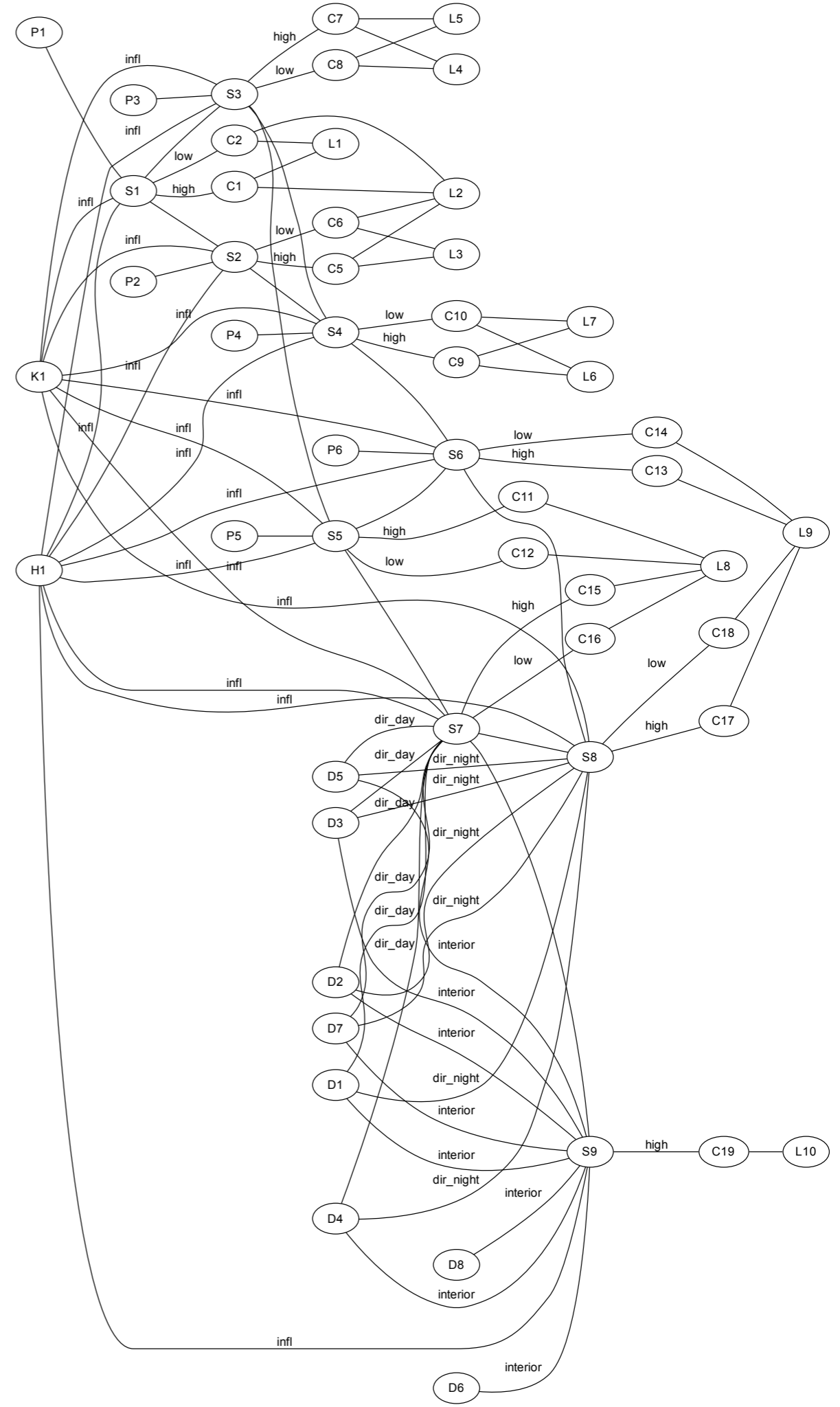
Map data ©2014 Google Terms of Use

# Inteligentne projektowanie

- Istotne osiągnięcia AGH.
- Projekt wygenerowany przez algorytmy AI (w oparciu o heurystyki) zamiast "ręcznie" przez projektanta (w oparciu o intuicję).
- Duża liczba konfiguracji oświetlenia dla różnych sytuacji (poziom światła, ruch drogowy, wykryte zagrożenia, itd.).

# Sterowanie

- Przestrzeń do oświetlenia dzielimy na **obszary** (*areas*).
- W **obszarach** wymagania dotyczące oświetlenia są **zbliżone** (np. jezdnia, chodnik, itd.).
- W zależności od wymagań, obszary dzielimy na **segmenty**.
- Dla segmentów określone są **konfiguracje lamp**, które realizują wymagany normami poziom oświetlenia w **segmentach**.



# Moduły

- Określono 7 modułów funkcyjnych oraz 3 moduły użytkowe.
- Moduły merytoryczne: warstwa GIS podkładu, warstwa GIS obiektów, model danych sensorów, model wymagań, reguły sterujące, profile i konfiguracje, logowanie.
- Moduły użytkowe: GUI, API REST, uwierzytelnianie +autoryzacja.

# Organizacja projektów

- 1 moduł = 1 projekt, grupy 3-osobowe, redundancja 2x.
- Specjalność (~60 osób) dzielimy na dwie połowy, każda przygotowuje wszystkie moduły określone wcześniej.
- Wybieramy dwie osoby, które będą koordynatorami ~30-osobowych grup.
- Pozostałe osoby dzielą się na grupy i podejmują poszczególne tematy.



Moduły funkcjonalne

# 1. Warstwa GIS podkładowa

- Wykorzystuje narzędzia GIS.
- Model danych do przechowywania informacji o „mapie”: ulice, budynki, drzewa, słupy, itd.
- Wstępny wsad z [OpenStreetMap](#).



## 2. Warstwa GIS obiektów

Przechowuje informacje o:

- oprawach (*fixtures*) i sterownikach (*drivers*),
- szafach (*cabinets*) i sterownikach segmentowych (*segment controllers*),
- lokalizacji czujników (ale bez szczegółowych informacji).

Baza przechowuje atrybuty stałe, chwilowe i parametry geolokalizacyjne.

# 3. Model danych sensorów

Określa katalog możliwych czujników

- np. pętle indukcyjne, czujniki ruchu, wideodetektory (kamery), czujniki oświetlenia zewnętrznego, liczniki pojazdów,
- ale również zegar astronomiczny, ręczne sterowanie, itd.

## 4. Model wymagań

Określa wymagania dla systemu dynamicznego sterowania w „idealnym” przypadku, tzn. przy założeniu, że mamy precyzyjną sensorykę.

Przykład: na drodze samochody poruszają się z prędkością 30 km/h (ok. 8,3 m/s), więc chcemy, aby „śledziła” je plama światła o długości 60 m.

Z tego wymagania wynika decyzja: dzielimy obszar na segmenty o długości 20 m i rozświetlany będzie segment z wykrytym pojazdem oraz po jednym przed i za nim.

## 5. Reguły sterujące

Moduł musi przechowywać reguły w logice predykatów, określające jaką konfigurację zaaplikować na lampach przy określonej kombinacji wartości i zdarzeń na sensorach.

```
r('1a1b') :- % reguła 1a1b
    v(d=dark,L,detected=false), % wierzchołek o indeksie
    L typu d=dark o etykiecie detected=false
    e(d=dark,L,s,J), % krawędź od wierzchołka d=dark o
    indeksie L do wierzchołka s o idneksie J
    + v(s,J,off), % wierzchołek s o indeksie J nie ma
    etykiety off
```

## 6. Profile i konfiguracje

Moduł przechowuje informacje wygenerowane przez system do projektowania oświetlenia, np.:

Aby uzyskać w danym segmencie normę oświetlenia ME3, przy 20% oświetleniu zewnętrznym, należy zapalić lampy L3 i L4 ze ściemnieniem 30%.

# 7. Logowanie

Moduł odpowiedzialny za gromadzenie historii działania systemu, w tym:

- zmian wartości odczytywanych z sensorów,
- zmian parametrów życiowych lamp,
- decyzji podejmowanych przez system,
- komunikatów wysyłanych do urządzeń.



Moduły użytkowe

# 8. GUI

Grupa odpowiedzialna będzie za przygotowanie prostej aplikacji webowej, pozwalającej na odczytywanie i wpisywanie wartości do baz poszczególnych modułów funkcjonalnych.

# 9. REST API

Grupa odpowiedzialna za wsparcie poszczególnych grup w przygotowaniu (tam gdzie to konieczne) API REST.

# 10. Uwierzytelnianie i autoryzacja

Grupa odpowiedzialna za stworzenie modułu uwierzytelniania i autoryzacji, w tym:

- przygotowanie bazy użytkowników,
- przygotowanie interfejsów,
- określenie możliwych uprawnień do poszczególnych modułów,
- wstępne zasilenie danymi.