Durham University

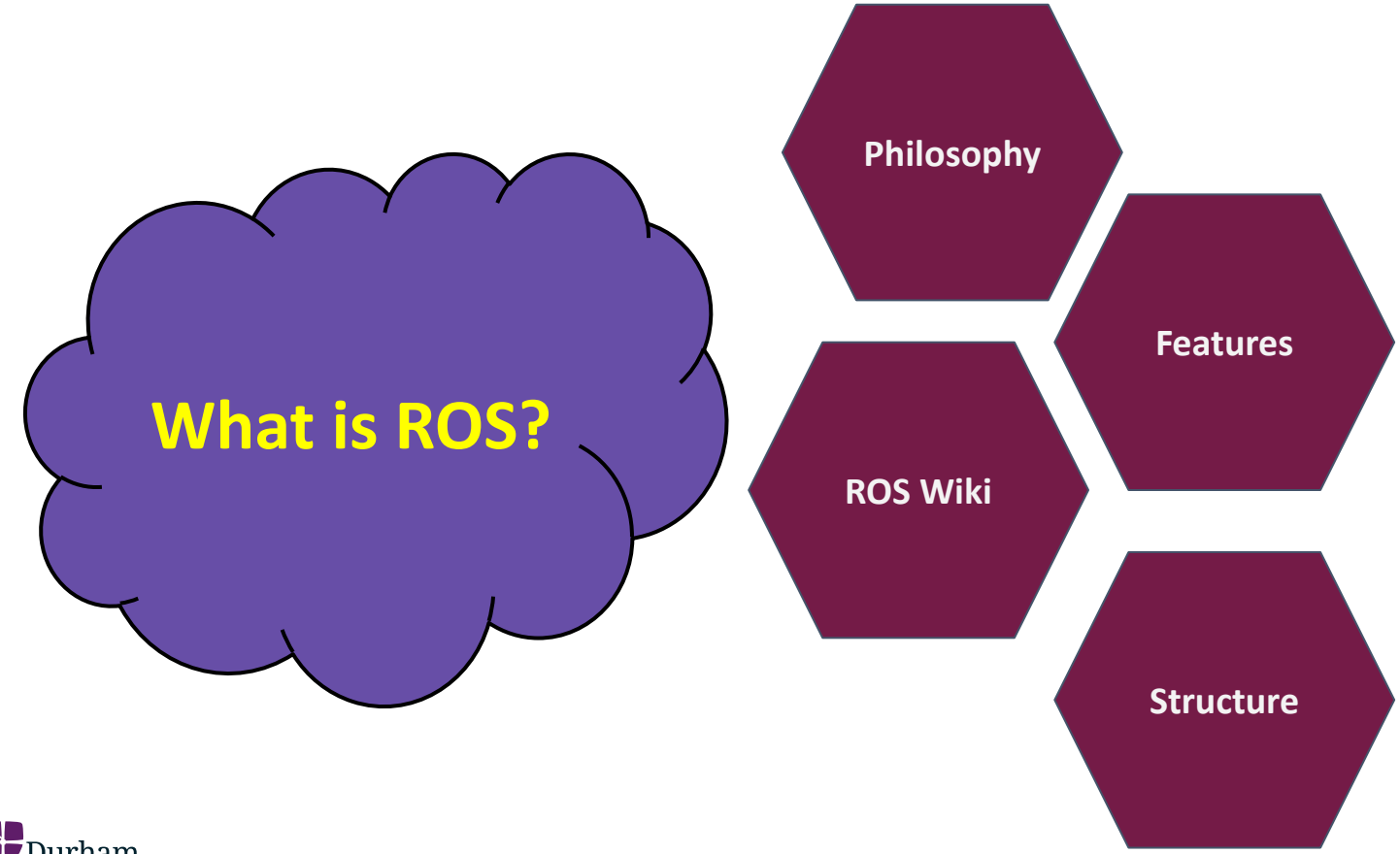
Robotics – Planning and Motion

COMP52815

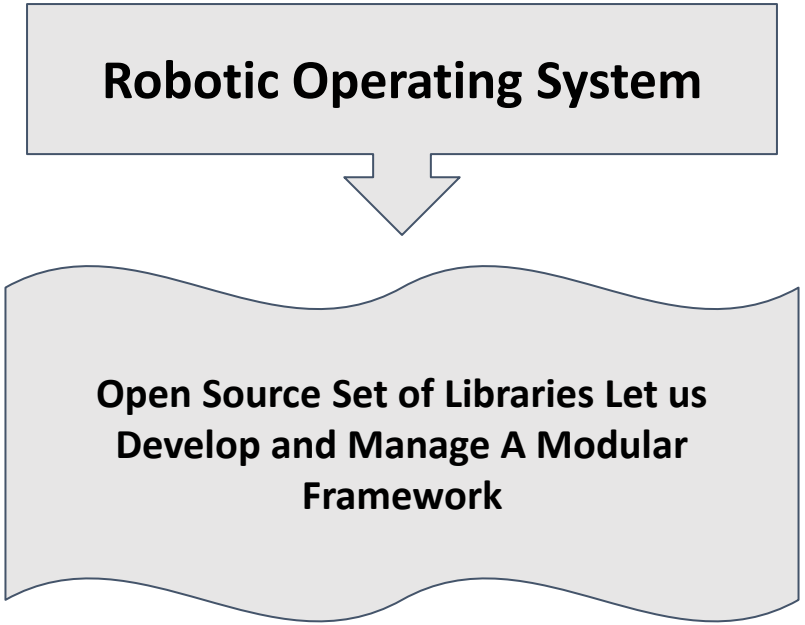
Dr Fatemeh Rekabi Bana

Email: fatemeh.Rekabi-bana@durham.ac.uk
Room: MCS 1005a

Learning Objectives



What is ROS?



Durham University

Philosophy:

The development of a new robotic system relies on:

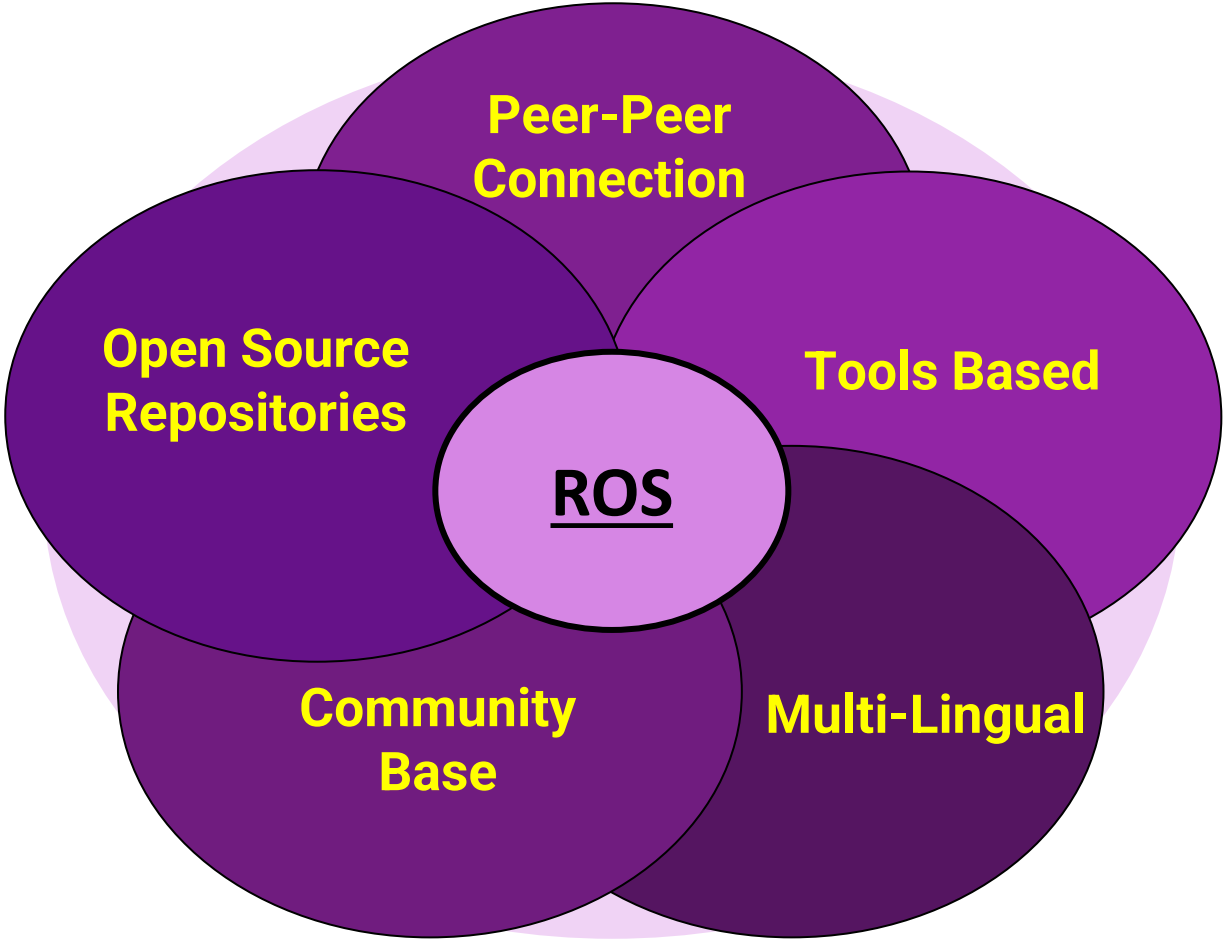
- **Modularity:** using ready modules (sensors, actuators, etc.) instead of making everything from scratch.

- **Distributed computation:** each module (software or hardware) may need an independent computational resource.

- **Robustness and Reliability:** it is necessary to ensure all the modules work together consistently regardless of uncertainties or disturbances.

- **Scalability:** adding new features, expanding the capability domain, and even making new products based on the current design led us to consider scalability in the development process.

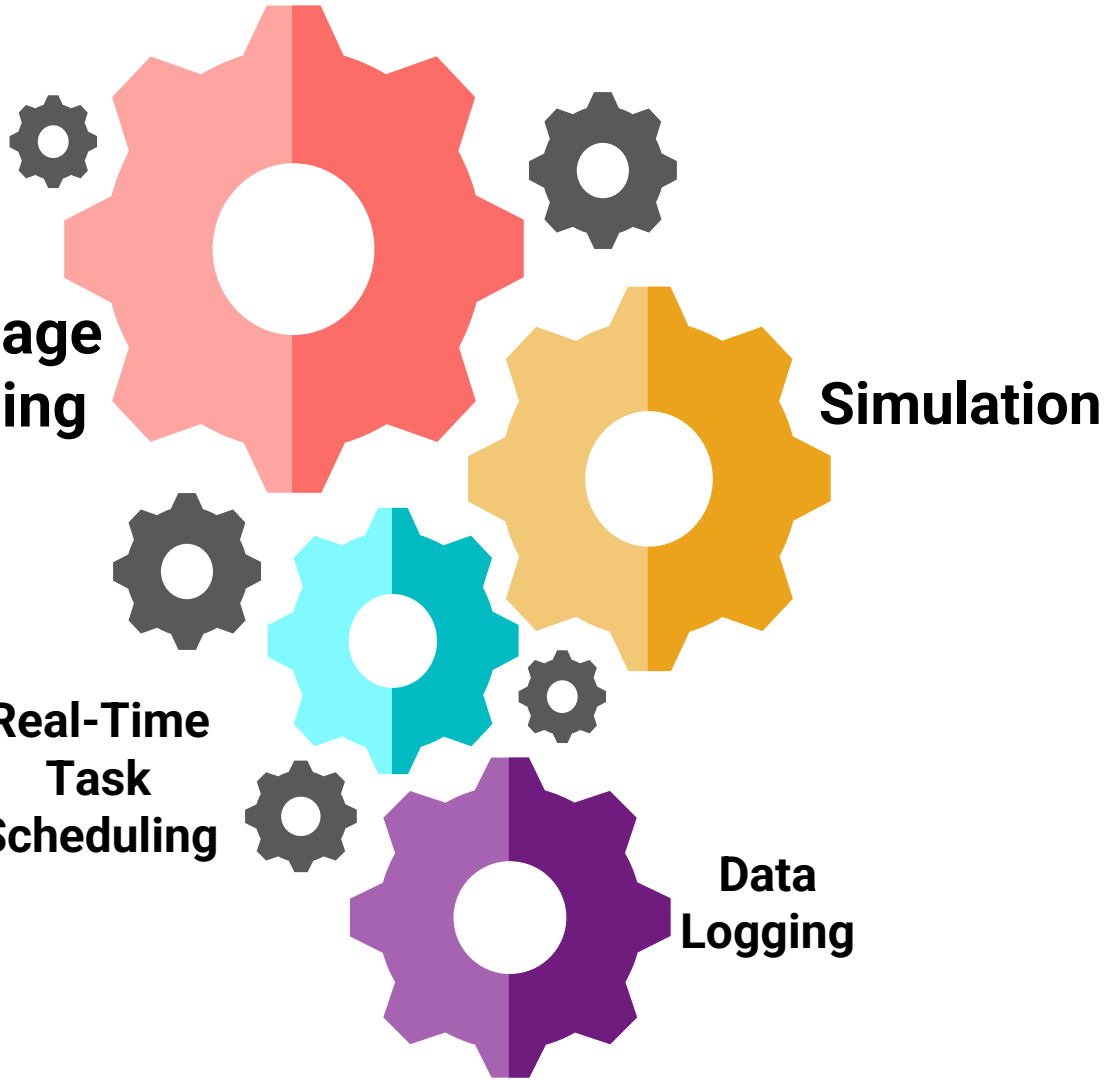
Features:



Durham University

Features:

Tools



Durham University

ROS Documents

ROS.org
Documentation

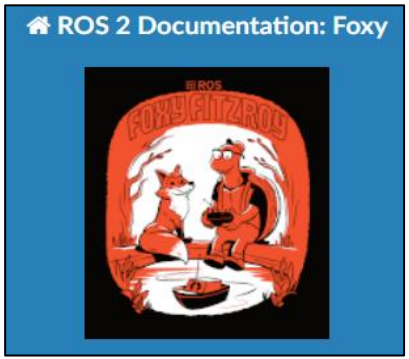
ROS Wiki:

https://wiki.ros.org/Documentation

ROS Robots

ROS Robots:

https://robots.ros.org/



ROS2 Documents:

https://docs.ros.org/en/foxy/index.html

Durham University

ROS Main Concepts:

Node

- Single-purposed executable programs
- Independently worked and managed
- They are written using a ROS library

Message

- Data structure for communication between nodes

Topics

- A customised message dedicated to transferer data on the network
- Nodes can subscribe/publish all the Topics on the network

ROS Main Concepts:

Service

- Synchronous inter node transactions
- (blocking RPC): ask for something and wait for it

Action

- standardized interface for interfacing with non-interrupting tasks

ROS Main Concepts:

Parameter Server

- A shared dictionary that is accessible via network
- Best used for static data such as configuration parameters

Master

- Provides connection information to nodes so that they can transmit messages to each other

Packages

- Software in ROS is organized into packages
- A package contains one or more nodes, documentation, messages, services, …

ROS2 Ecosystem:

Visualisation Tools (RVIZ)

Simulation Tools (GAZEBO)

Available Cross-Platform libraries and community support

ROS Applications in robotics:

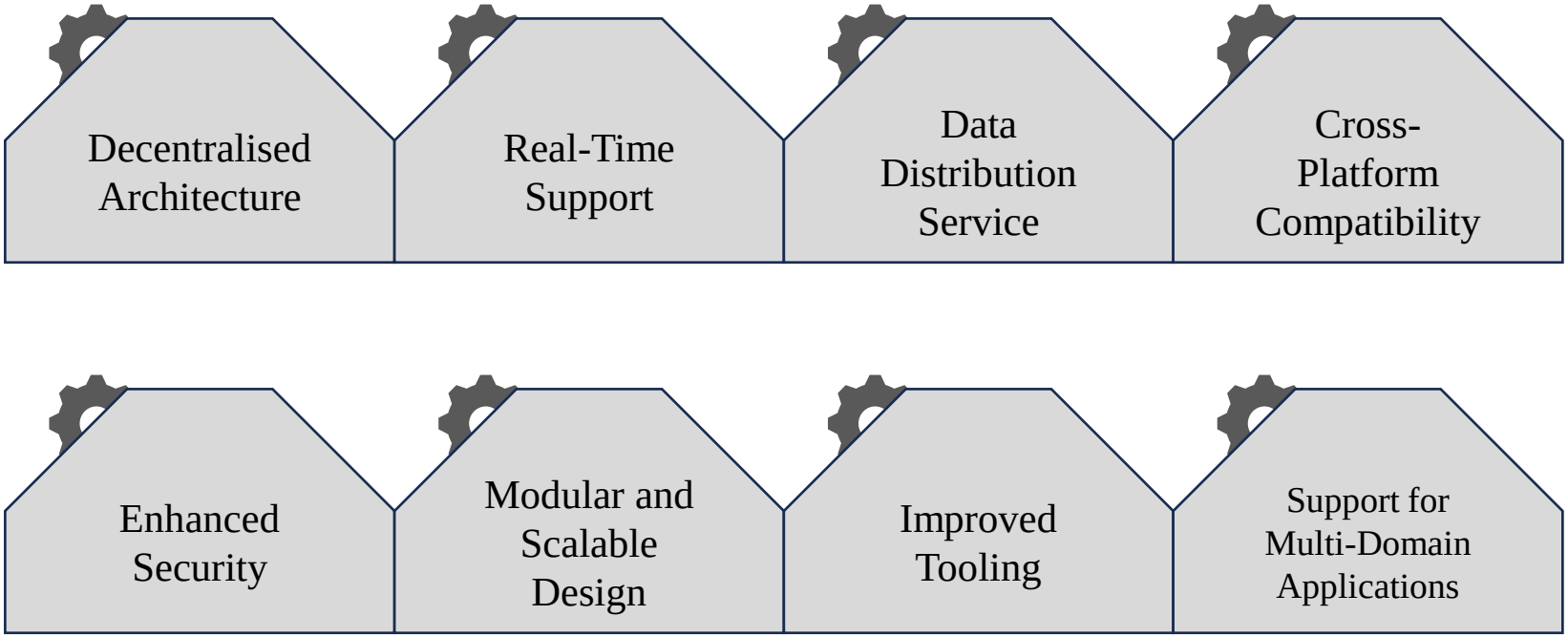
Algorithms: autonomous navigation, manipulation, and swarm robotics.

Real-world use cases: delivery robots, drones, and healthcare robots

Industrial applications: self-driving cars, precision agriculture, and collaborative robots

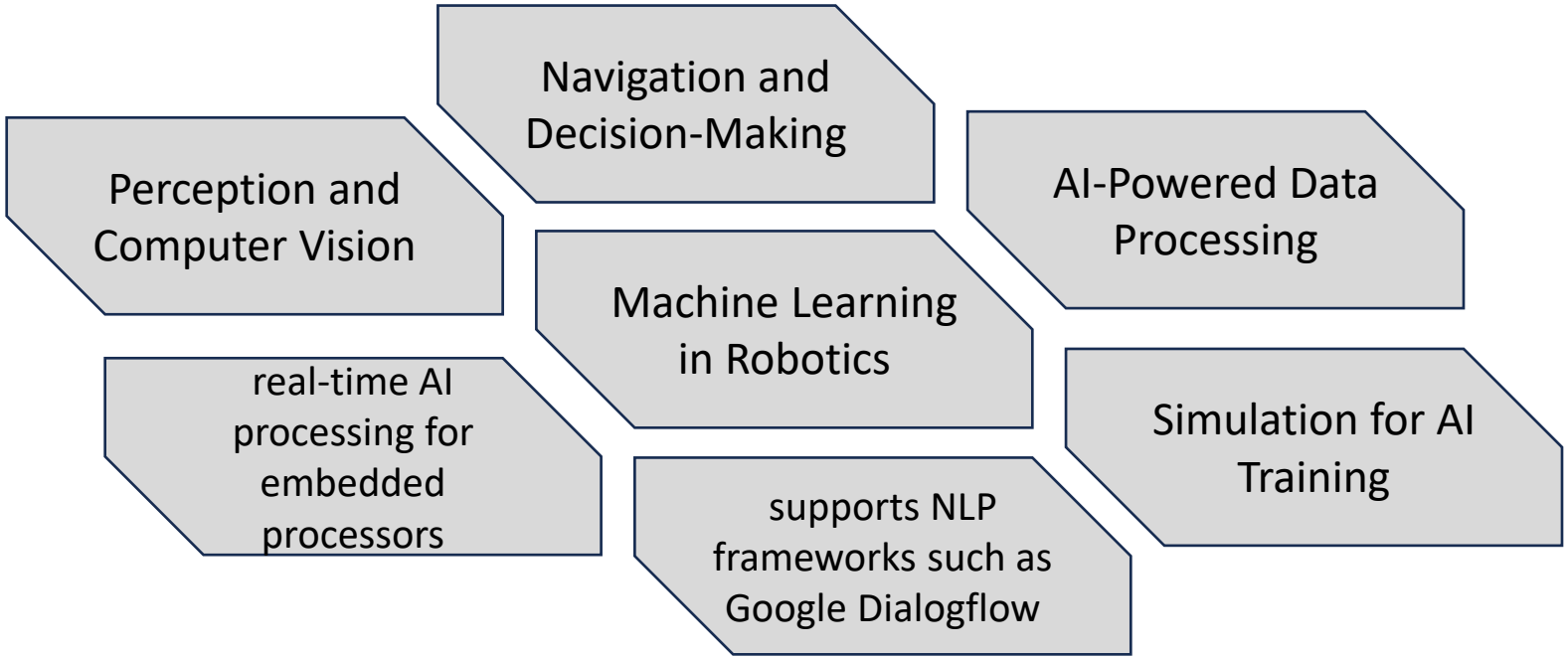
Advanced use cases in real-time systems (ROS2)

ROS2 and its advantages:



ROS/ROS2 and AI Integration:

- Tools and frameworks for AI integration into robotic systems for tasks like perception, decision-making, and learning



Summary

- Introduction to ROS
- Main ROS concept
- ROS2 features and advantages over ROS