JACK – Agent Software Handson I

Computer Applications in
Power Systems – Advance course

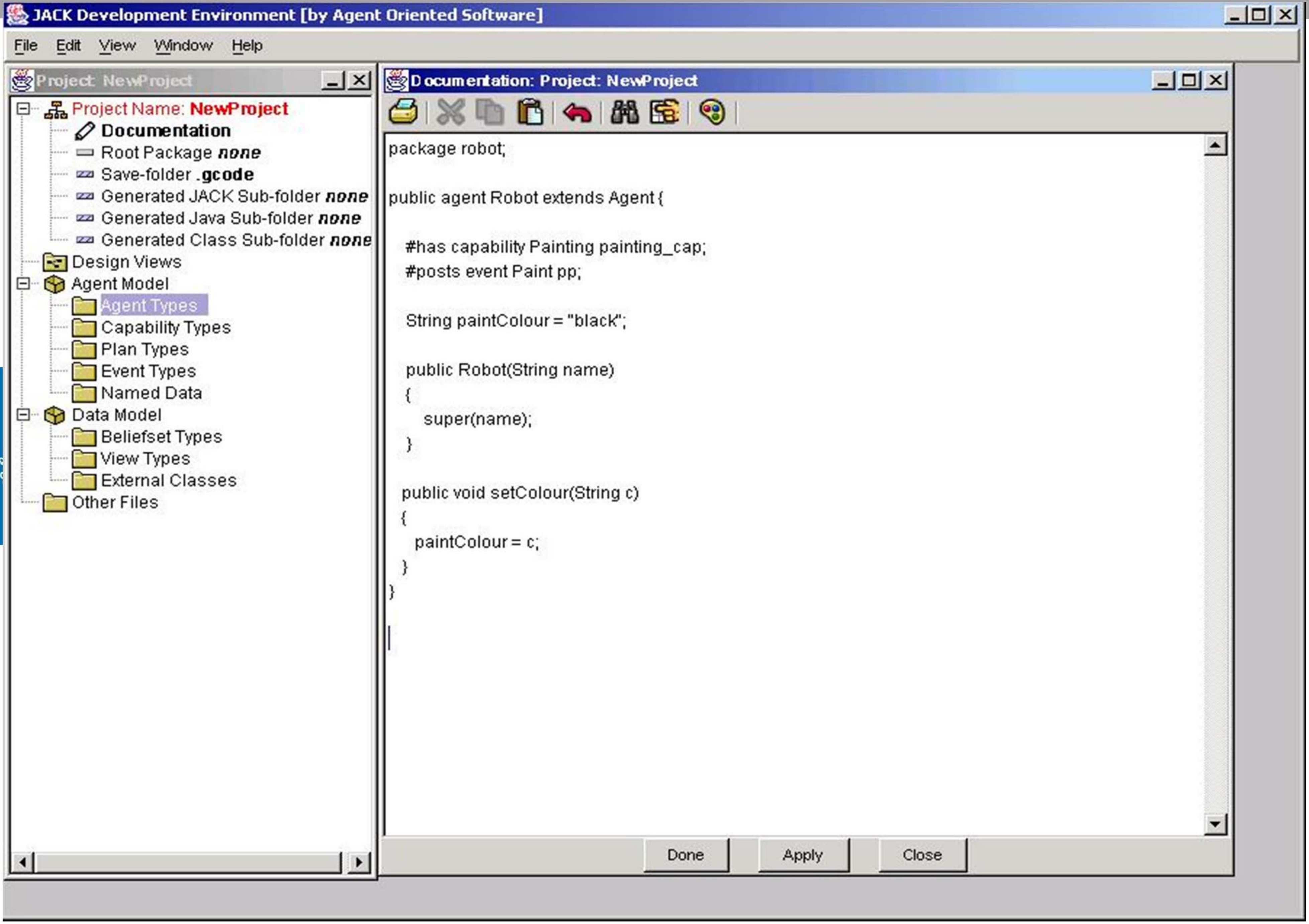
EH2750

Outline

- What is JACK
- JACK Architecture
- JACK Agent language and it’s concepts
- Todays work
- Some practicals

What is JACK

JACK Intelligent Agents is an *environment* for building, running and integrating commercial *Java-based* multi-agent software using a *component-based* approach.


JACK Development Environment [by Agent Oriented Software]
File Edit View Window Help
Project: NewProject
Documentation: Project: NewProject
Done
Apply
Close

Project Name: NewProject
- Documentation
- Root Package *none*
- Save-folder **.gcode**
- Generated JACK Sub-folder *none*
- Generated Java Sub-folder *none*
- Generated Class Sub-folder *none*

Design Views

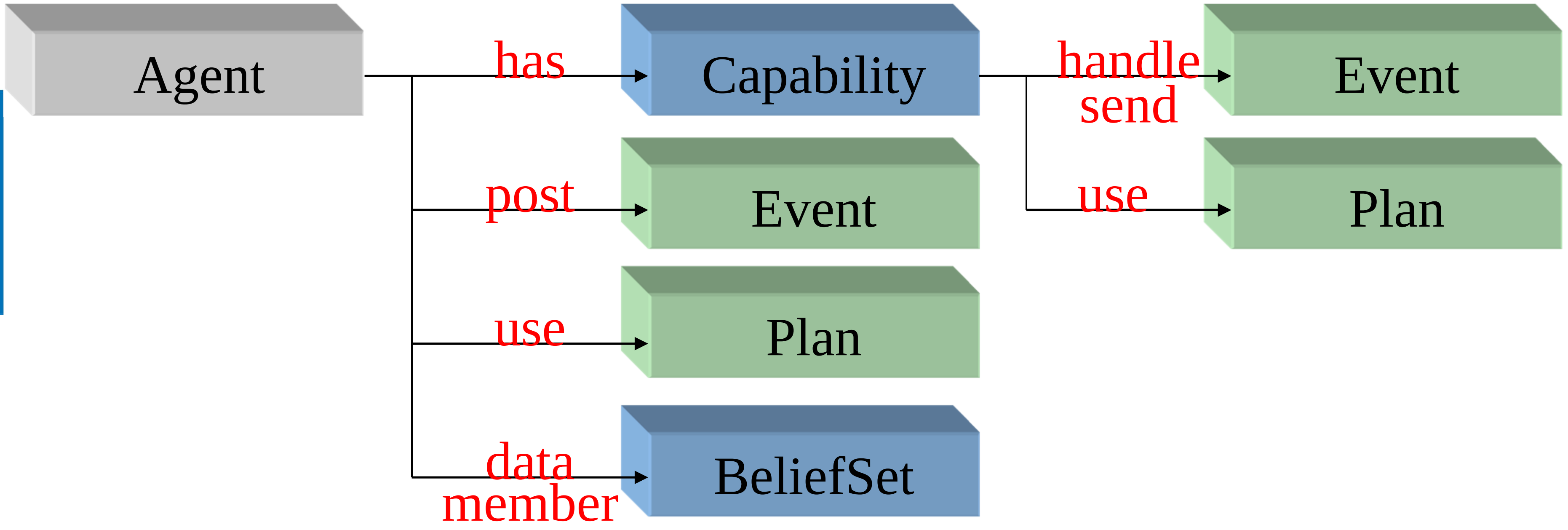
Agent Model
- Agent Types
- Capability Types
- Plan Types
- Event Types
- Named Data

Data Model
- Beliefset Types
- View Types
- External Classes

Other Files

```
package robot;

public agent Robot extends Agent {

  #has capability Painting painting_cap;
  #posts event Paint pp;

  String paintColour = "black";

  public Robot(String name)
  {
     super(name);
  }

 public void setColour(String c)
 {
    paintColour = c;
 }
}
```

JACK Architecture



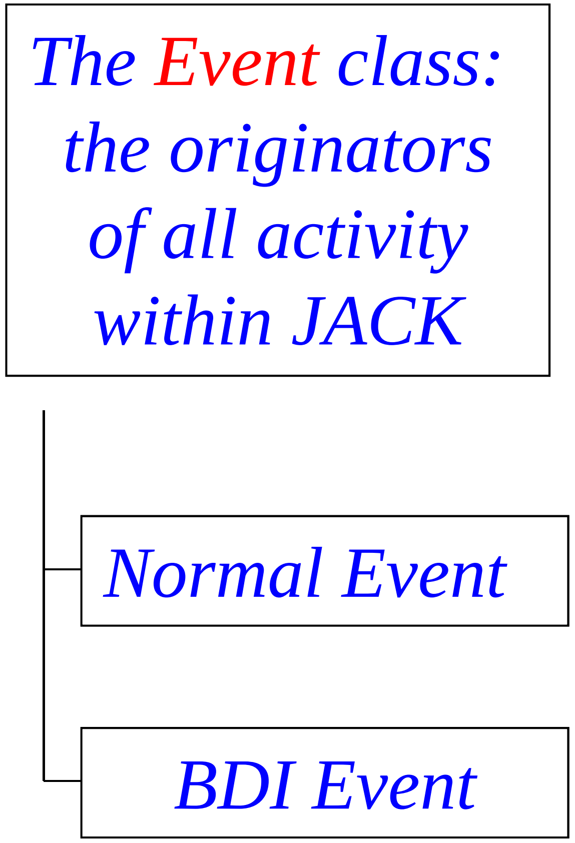
Most important concepts in JACK Agent Language(1)

The Agent class: embodies the functionality associated with a JACK intelligent agent

```
agent AgentType extends Agent {implements
    interface}
{
   #{private,agent,global} data Type Name (arglist);
    #handles event EventType;
    #uses plan PlanType;
    #posts event EventType reference;
    #has capability CapabilityType reference;
    //data members (Java data structures)
    :
    //Constructor method
    AgentType(arglist)
    {
    super("agent name");
    :
    }
    //Java methods that implement agent
    functionality
    :
}
```

Most important concepts in JACK Agent Language(2)

```
event EventType extends Event
{
    // Any members required as part of the event structure.
    // Any declarations required to give the agent access to
    data or
    // beliefsets within the enclosing agent or capability.
    #uses data DataType data_name;
    // Any #posted when declarations required so that the
    event
    // will be posted automatically when certain belief states
    arise.
    #posted when { condition }
    // Declarations specifying how the event is posted within
    an agent.
    // You can have as many posting methods as you require.
    #posted as postingMethodName(parameter list)
    {
    // method body
    }
}
```


The Event class: the originators of all activity within JACK
Normal Event
BDI Event

Most important concepts in JACK Agent Language(3)

The Plan class: describes a sequence of actions that an agent can take when an event occurs

```
plan PlanType extends Plan
{
     #handles event EventType eventref;
     #posts event EventType eventref2;
     #reasoning method pass(){
     // Post-processing and clean up steps when the plan
     has
     // succeeded
     }
     #reasoning method fail(){
     // Post-processing and clean up steps when the plan
     has failed
     }
     static boolean relevant (EventType eventref){
     }
     context(){
     }
     body(){
     }
}
```

Exercises today

- HelloWorld
- Planner Agent
- Choosing from multiple plans

Practicals

- Make a new folder with your name/group on C drive of computer and save project for each exercise there
- Basic steps are explained in more detail in exercise 1 handout. While going through later exercises refer back to the handout of the first one for any explainations