



Universidad de Granada

decsai.ugr.es

Practical Process Mining (II)

Master CD&IC

Luis Castillo Vidal

l.castillo@decsai.ugr.es

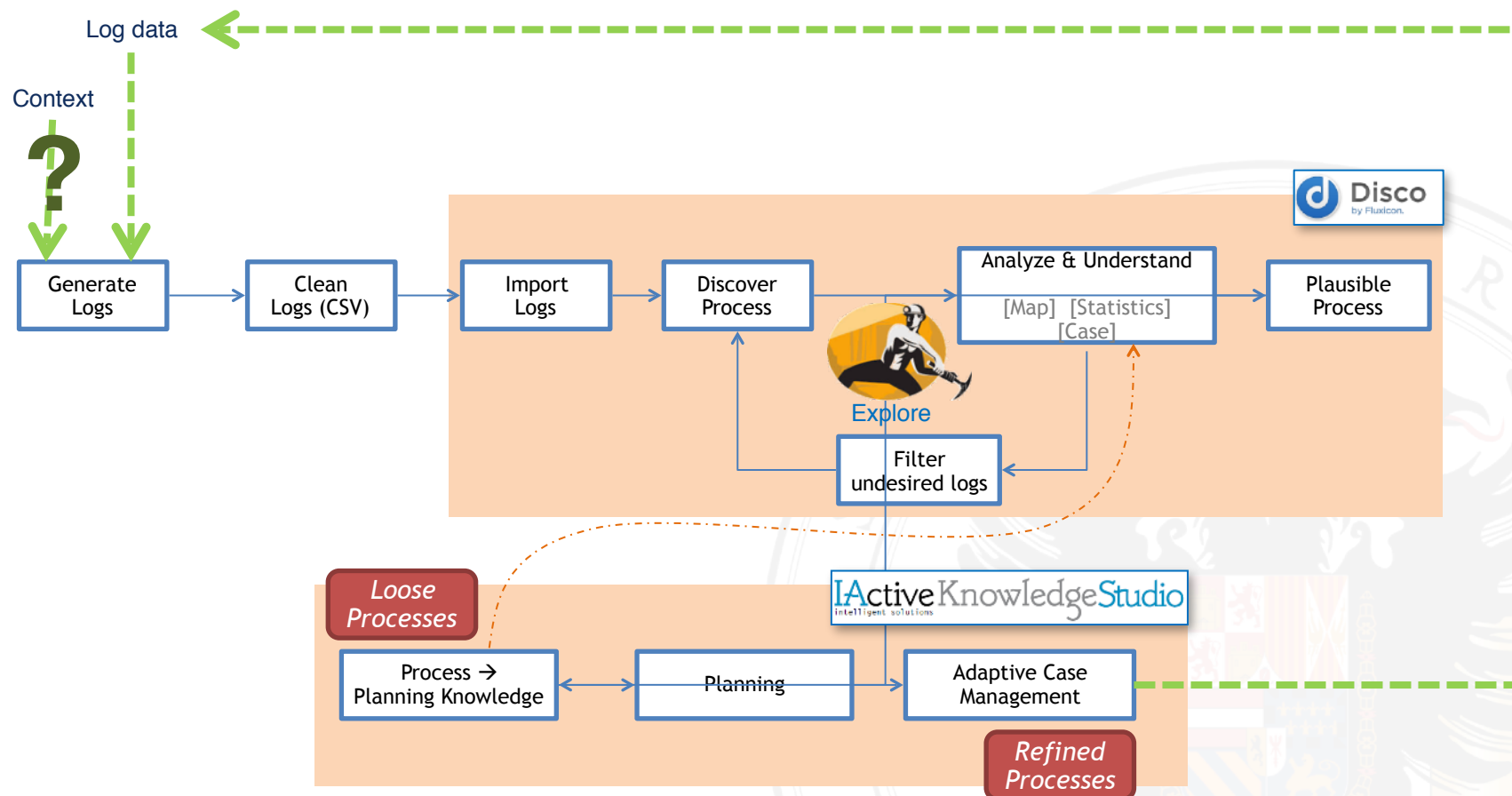
Skype: l.castillovidal

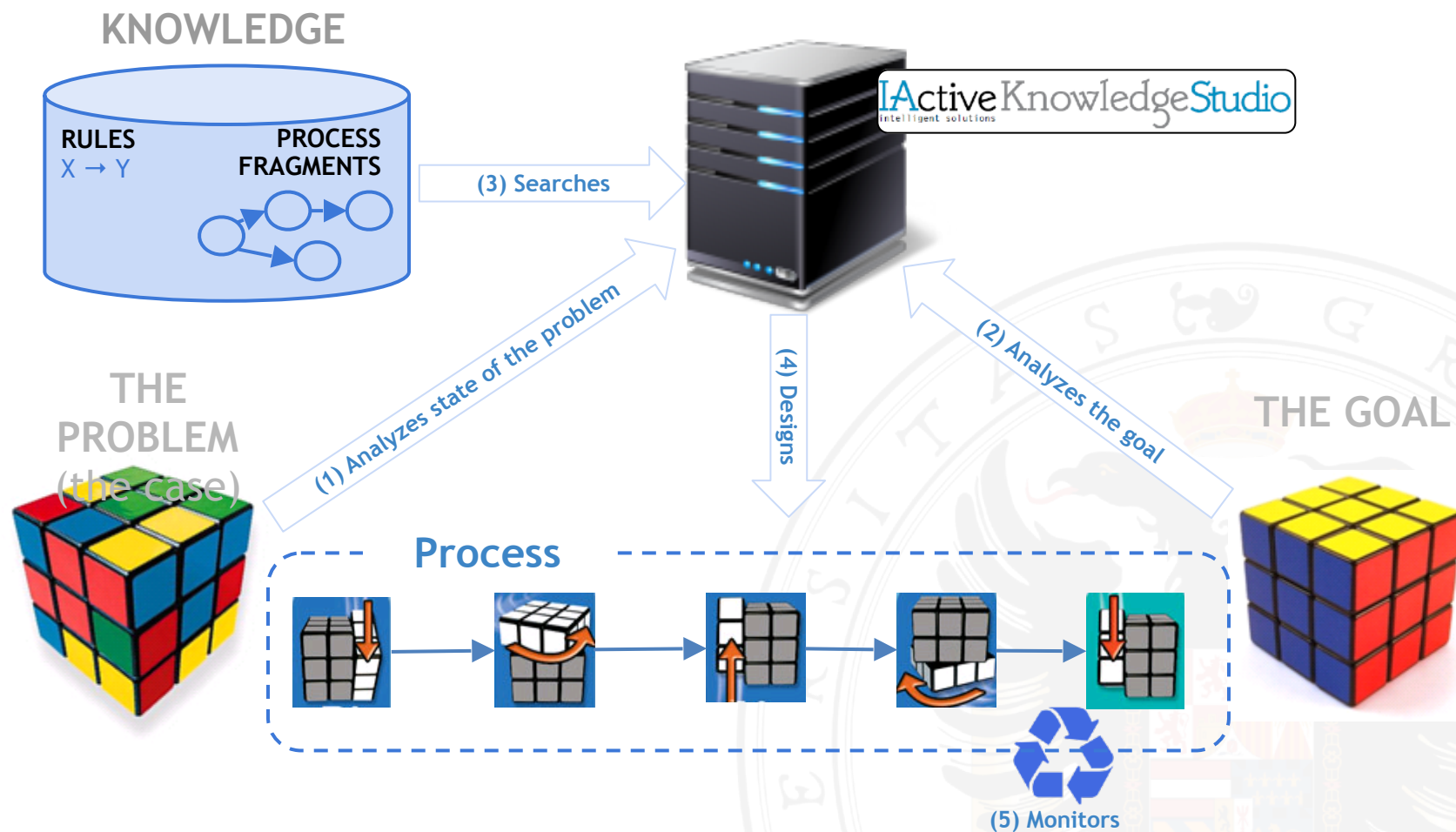


DECSAI

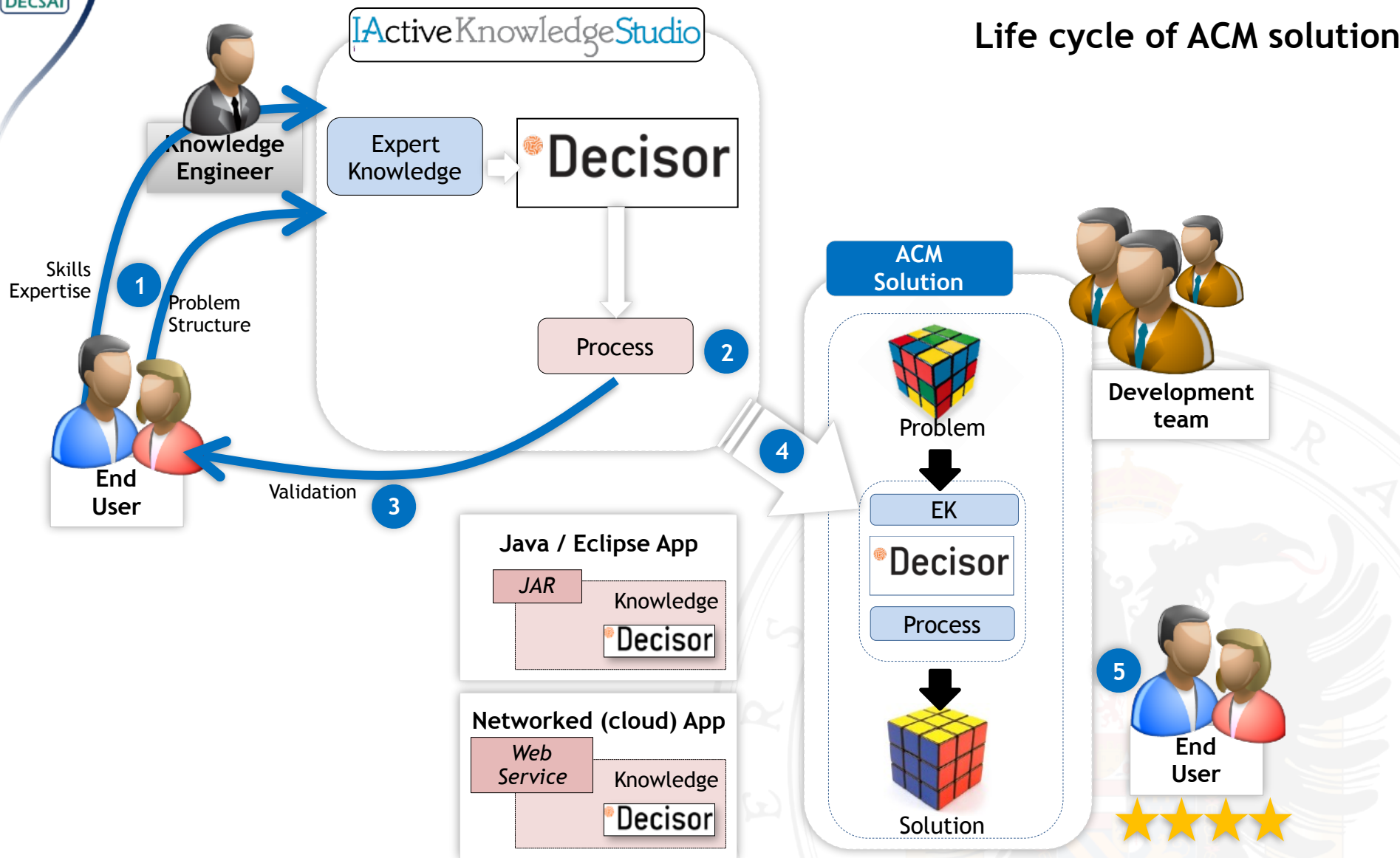
**Departamento de Ciencias de la
Computación e Inteligencia Artificial**

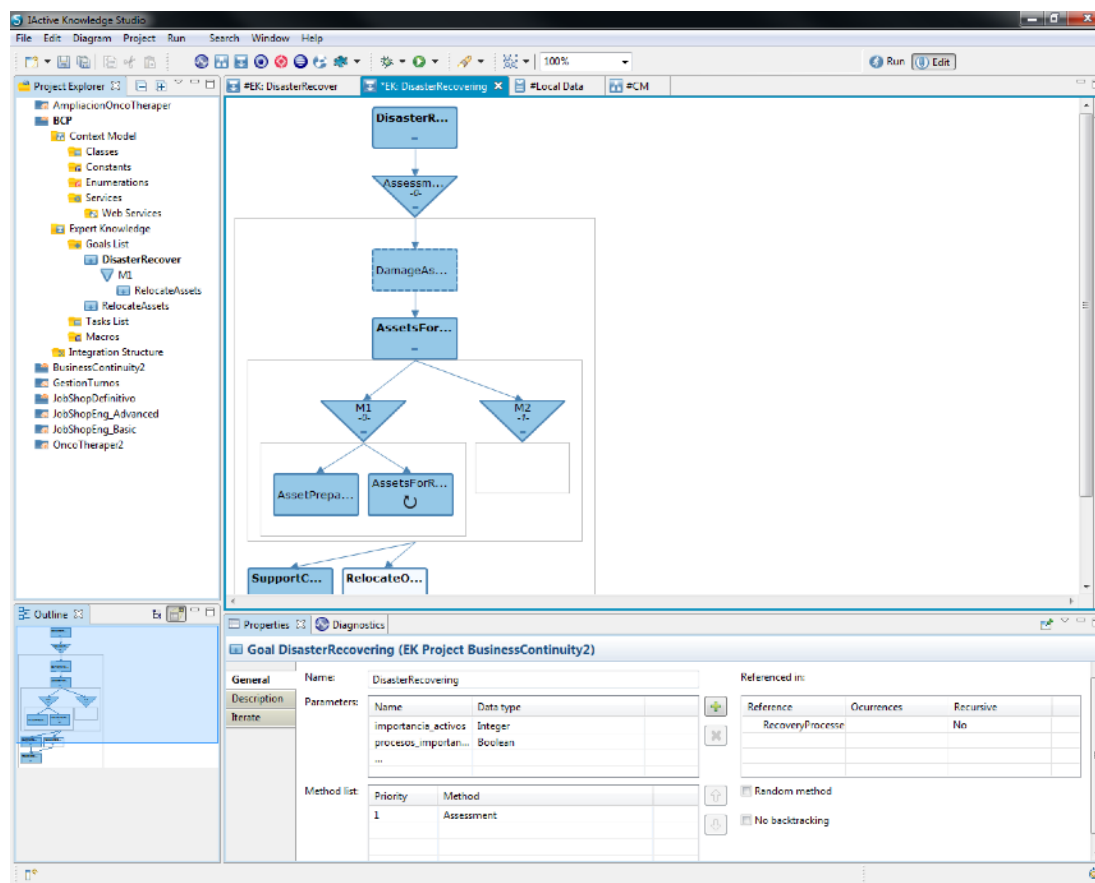
1. Introduction to practical Process Generation and Adaptive Case Management with Knowledge Studio

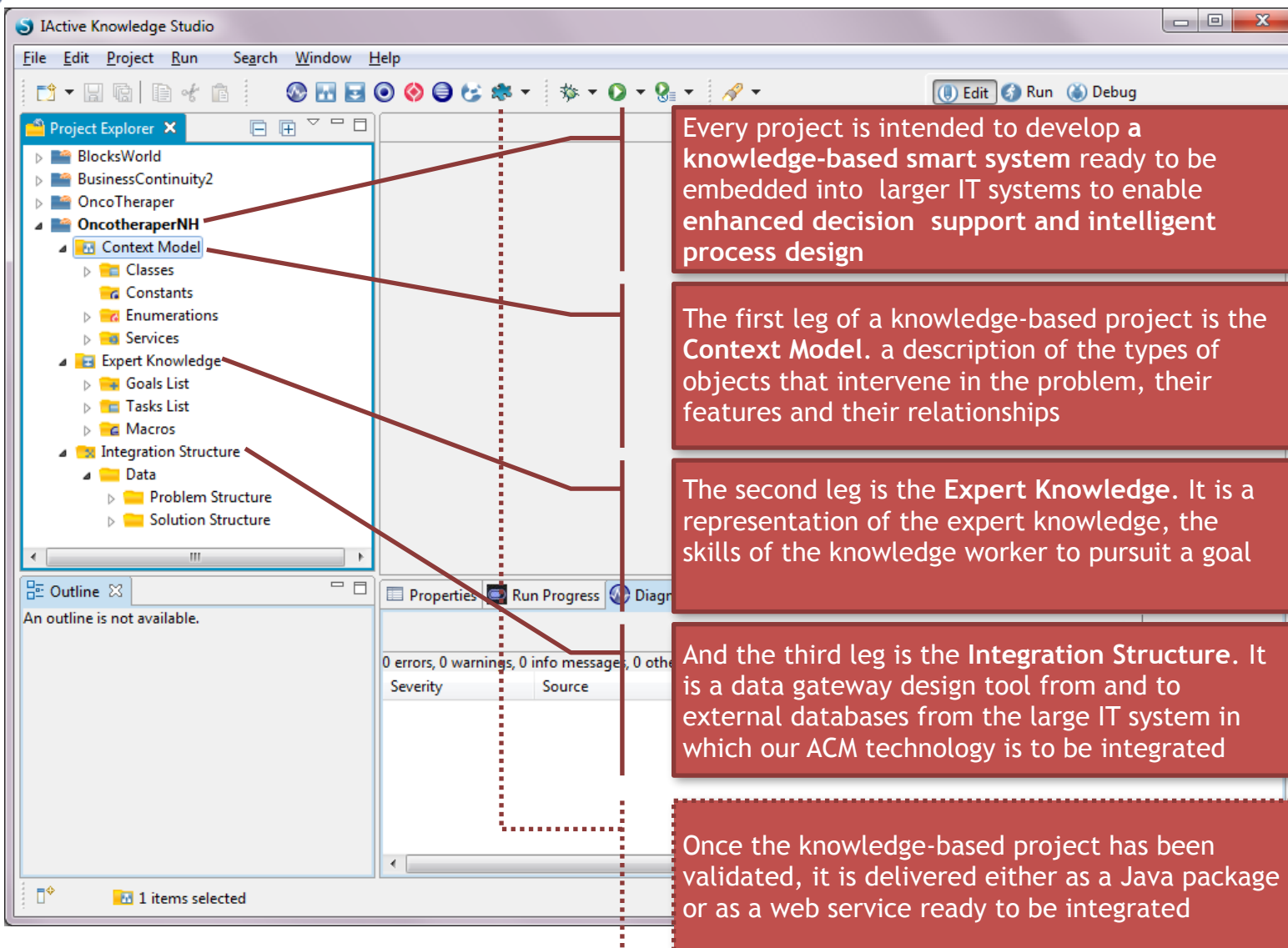




Life cycle of ACM solutions





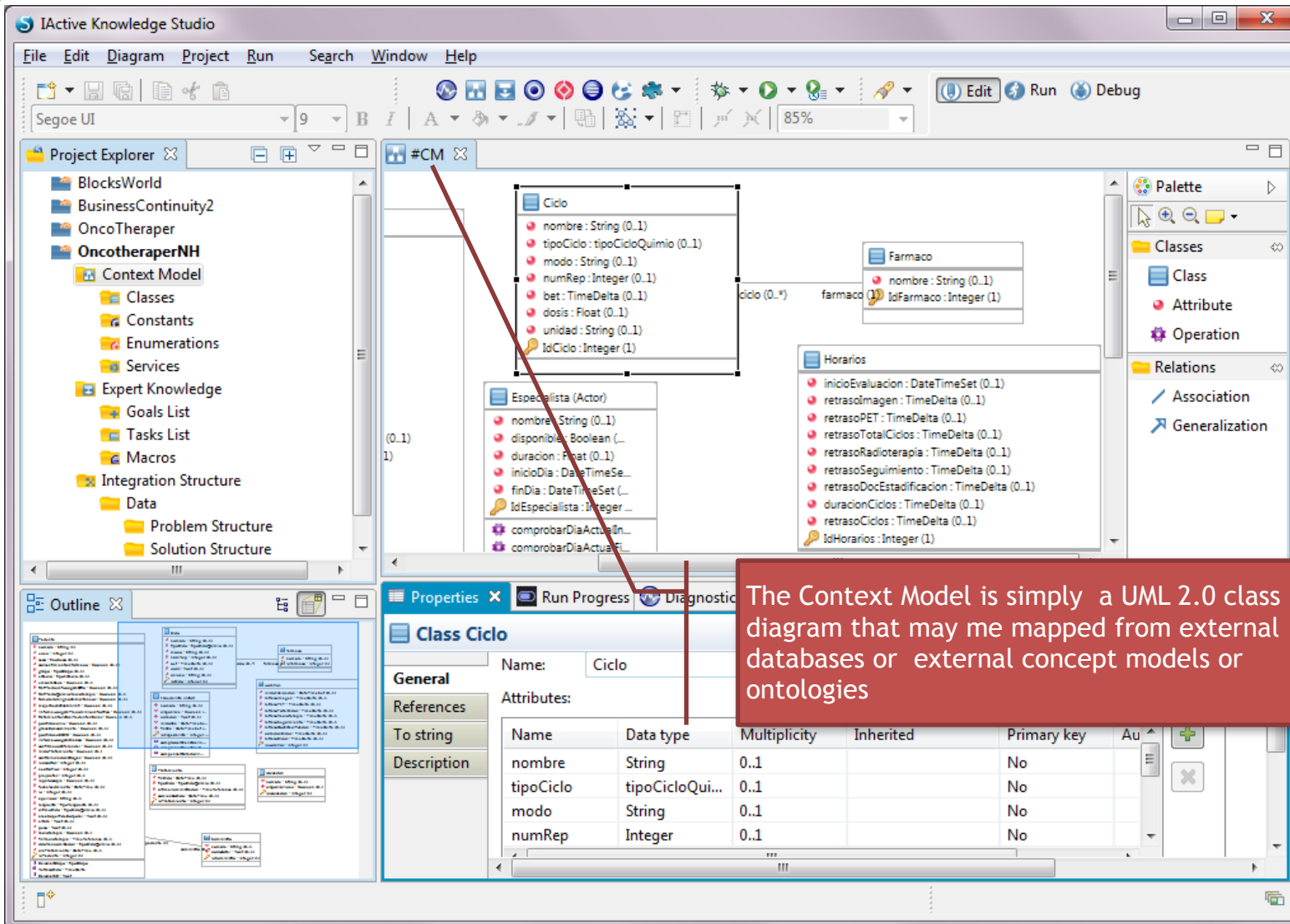


The screenshot shows the IActive Knowledge Studio interface. The Project Explorer on the left lists several projects, with 'OncotheraperNH' expanded to show its internal structure. Red lines connect specific components in the Project Explorer to explanatory text boxes on the right:

- Context Model:** A description of the types of objects that intervene in the problem, their features and their relationships.
- Expert Knowledge:** A representation of the expert knowledge, the skills of the knowledge worker to pursuit a goal.
- Integration Structure:** A data gateway design tool from and to external databases from the large IT system in which our ACM technology is to be integrated.

Additional text boxes provide general information about the project and its delivery:

- Every project is intended to develop a knowledge-based smart system ready to be embedded into larger IT systems to enable enhanced decision support and intelligent process design.
- Once the knowledge-based project has been validated, it is delivered either as a Java package or as a web service ready to be integrated.



The screenshot shows the IActive Knowledge Studio interface. The main workspace displays a UML class diagram for a Context Model. The diagram includes the following classes and their attributes:

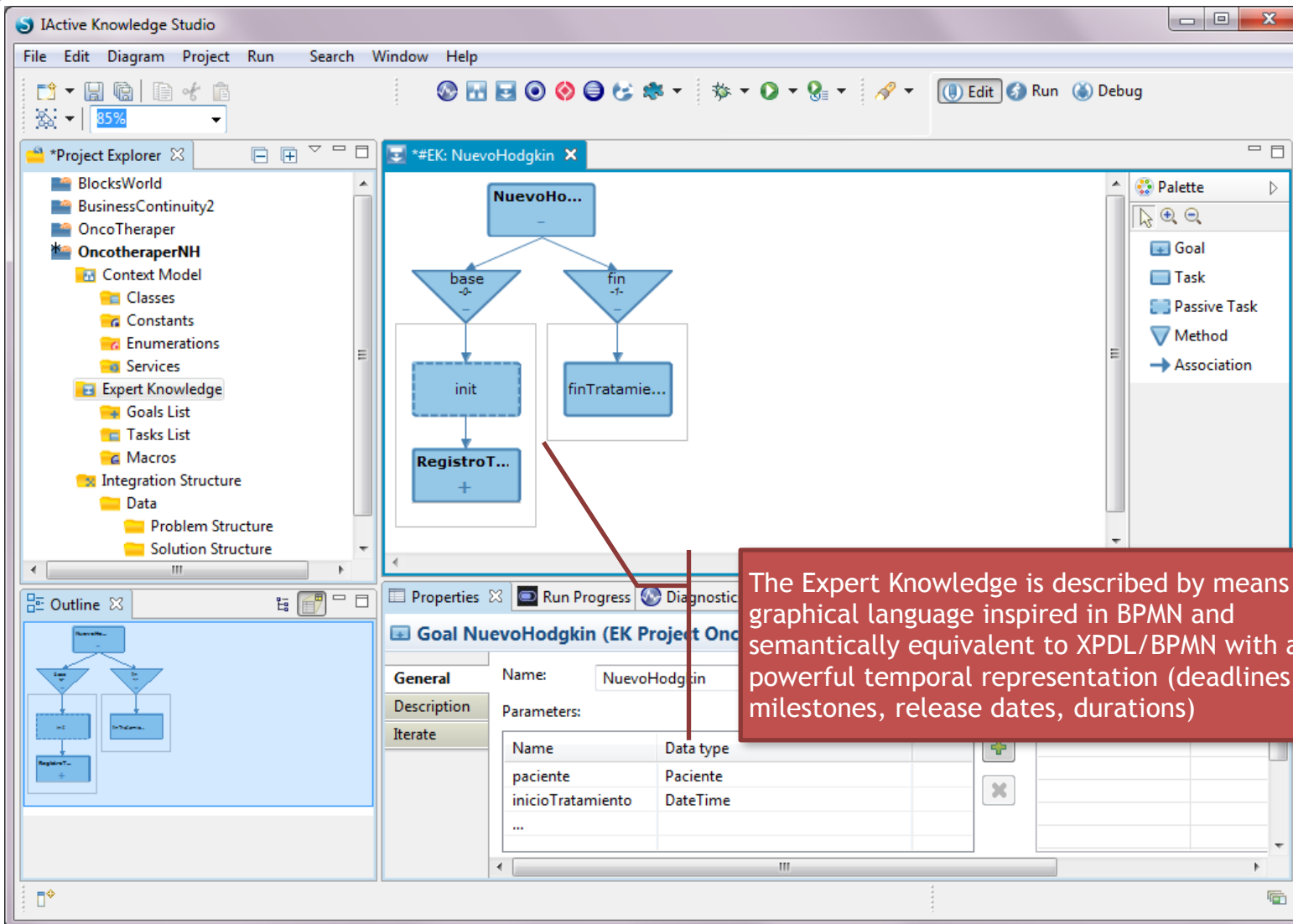
- Ciclo** (Class):
 - nombre: String (0..1)
 - tipoCiclo: tipoCicloQuimio (0..1)
 - modo: String (0..1)
 - numRep: Integer (0..1)
 - bet: TimeDelta (0..1)
 - dosis: Float (0..1)
 - unidad: String (0..1)
 - IdCiclo: Integer (1)
- Farmaco** (Class):
 - nombre: String (0..1)
 - IdFarmaco: Integer (1)
- Horarios** (Class):
 - inicioEvaluacion: DateTimeSet (0..1)
 - retrasoImagen: TimeDelta (0..1)
 - retrasoPET: TimeDelta (0..1)
 - retrasoTotalCiclos: TimeDelta (0..1)
 - retrasoRadioterapia: TimeDelta (0..1)
 - retrasoSeguimiento: TimeDelta (0..1)
 - retrasoDocEstadificacion: TimeDelta (0..1)
 - duracionCiclos: TimeDelta (0..1)
 - retrasoCiclos: TimeDelta (0..1)
 - IdHorarios: Integer (1)
- Especialista (Actor)** (Class):
 - nombre: String (0..1)
 - disponible: Boolean (0..1)
 - duracion: Float (0..1)
 - inicioDia: DateTimeSet (0..1)
 - finDia: DateTimeSet (0..1)
 - IdEspecialista: Integer (1)
 - comprobarDiaActualEn: Operation
 - comprobarDiaActualEn: Operation

The diagram also shows associations between these classes. A red arrow points from the text box to the 'Ciclo' class in the diagram.

Properties Panel - Class Ciclo

Name	Data type	Multiplicity	Inherited	Primary key	Auto
nombre	String	0..1		No	
tipoCiclo	tipoCicloQui...	0..1		No	
modo	String	0..1		No	
numRep	Integer	0..1		No	

The Context Model is simply a UML 2.0 class diagram that may be mapped from external databases or external concept models or ontologies

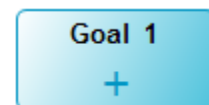


The screenshot displays the IActive Knowledge Studio interface. On the left, the **Project Explorer** shows a tree structure with folders like **BlocksWorld**, **BusinessContinuity2**, **OncoTheraper**, and **OncotheraperNH**. Under **OncotheraperNH**, there are sub-folders for **Context Model**, **Expert Knowledge**, **Goals List**, **Tasks List**, **Macros**, **Integration Structure**, **Data**, **Problem Structure**, and **Solution Structure**. The **Outline** window at the bottom left shows a smaller version of the diagram. The main diagram area, titled ***EK: NuevoHodgkin**, shows a process flow starting with a goal **NuevoHo...**, branching into **base** and **fin** tasks, which then lead to **init** and **finTratamie...** tasks, and finally to a **RegistroT...** task. The **Properties** window at the bottom right shows the details for the **Goal NuevoHodgkin (EK Project Onco**, including its name, parameters, and a table of data types.

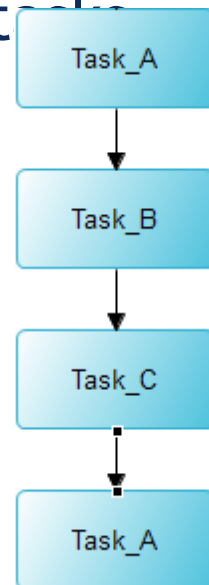
Name	Data type
paciente	Paciente
inicioTratamiento	DateTime
...	

The Expert Knowledge is described by means of a graphical language inspired in BPMN and semantically equivalent to XPD/LBPMN with a powerful temporal representation (deadlines, milestones, release dates, durations)

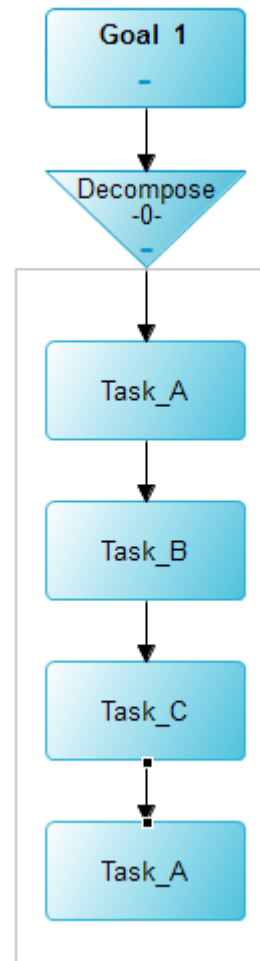
- Knowledge Studio formalism for representing processes is all about how to achieve a given goal



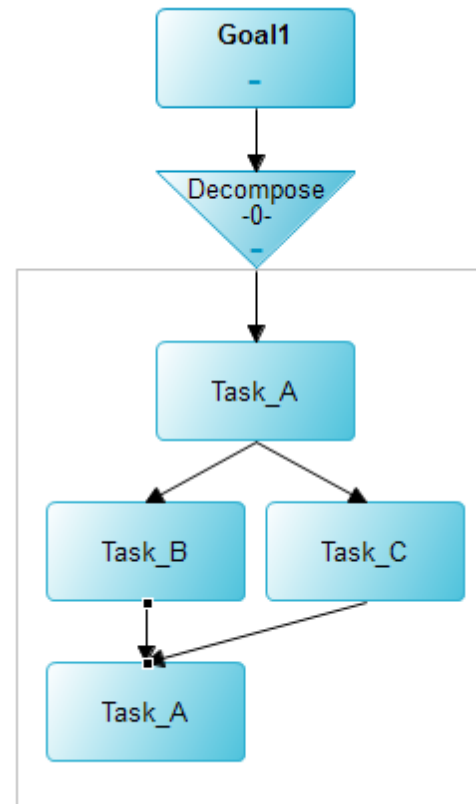
- Goals can be achieved by means of processes, i.e., sequences of tasks



- A goal can be decomposed into a subprocess that solves that goal

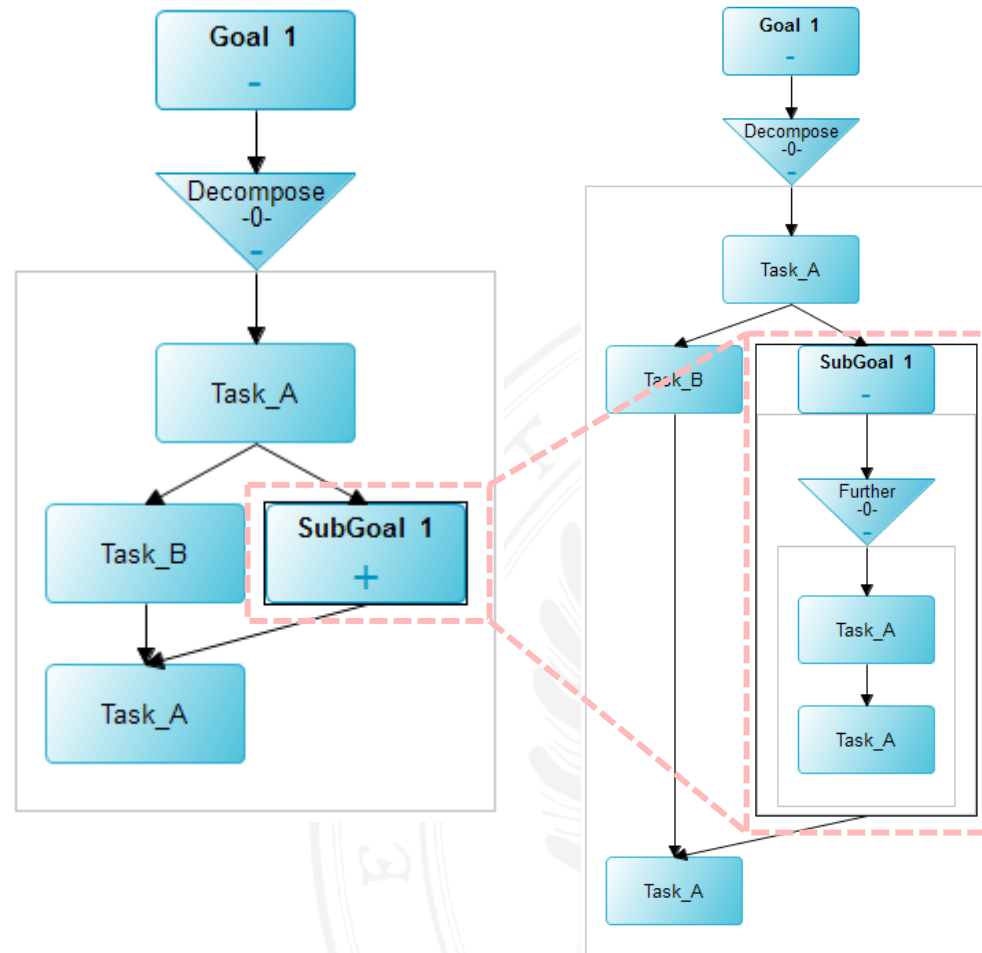


- Subprocesses may have parallel branches as well

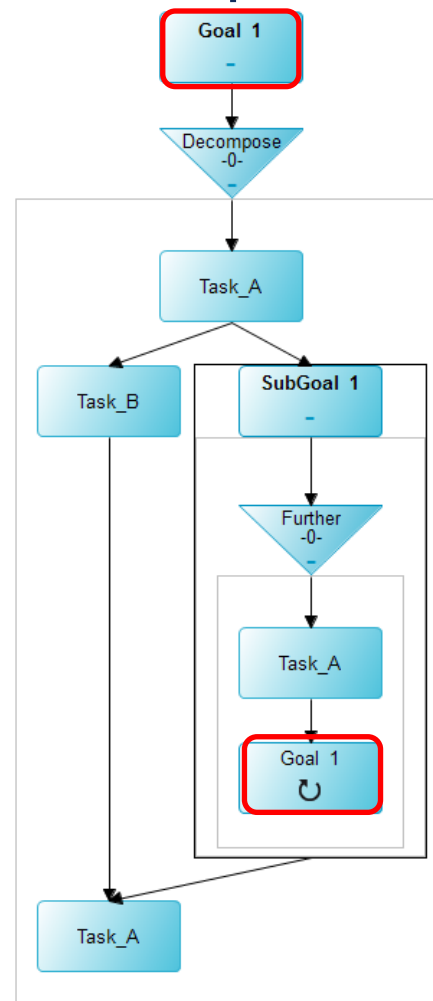


- With complex timing among them: “10 min before”, “at the same time”, “no later than”, “exactly at”

- Subprocesses might be further decomposed

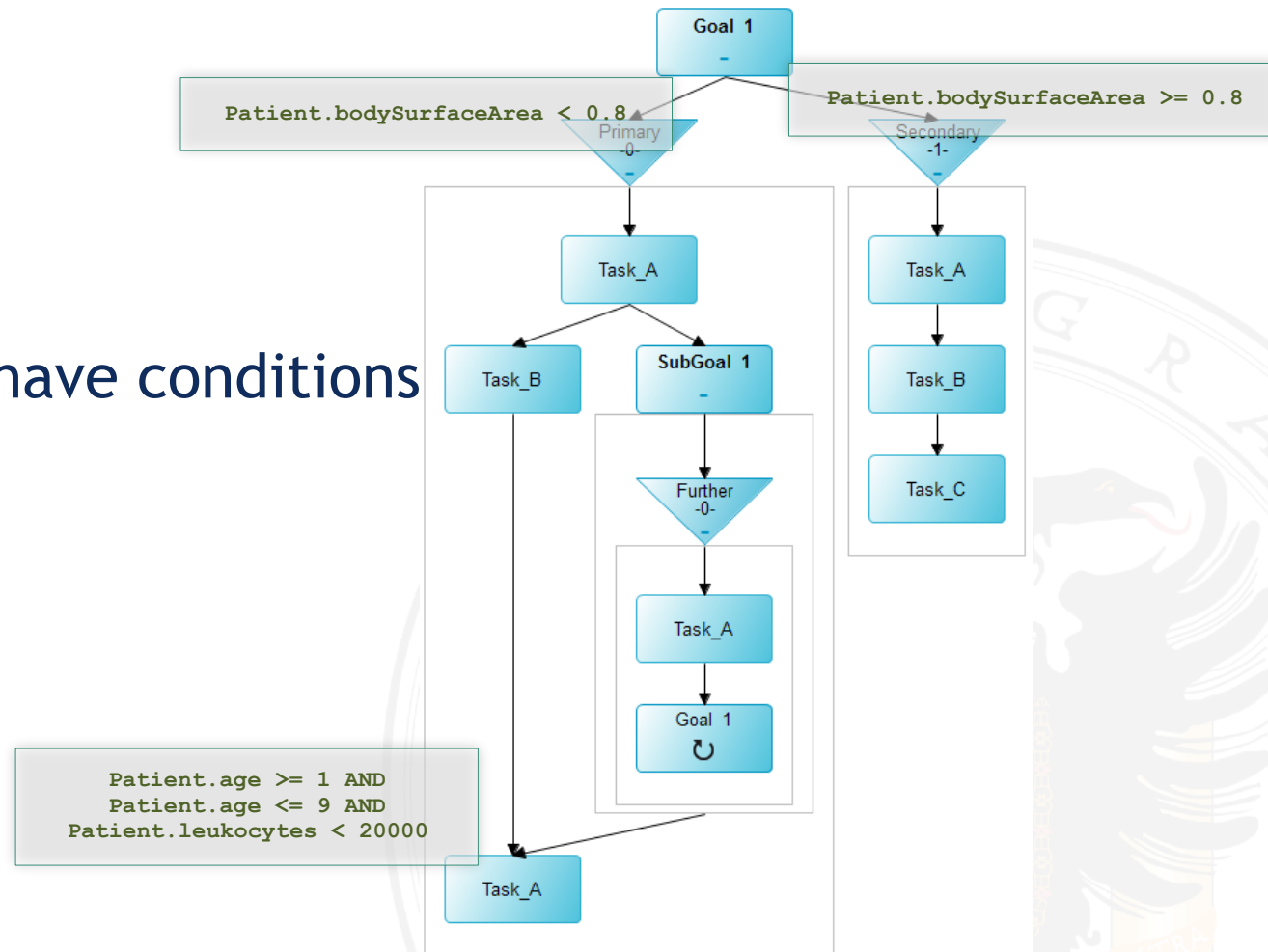


- Subprocesses might also be repeated



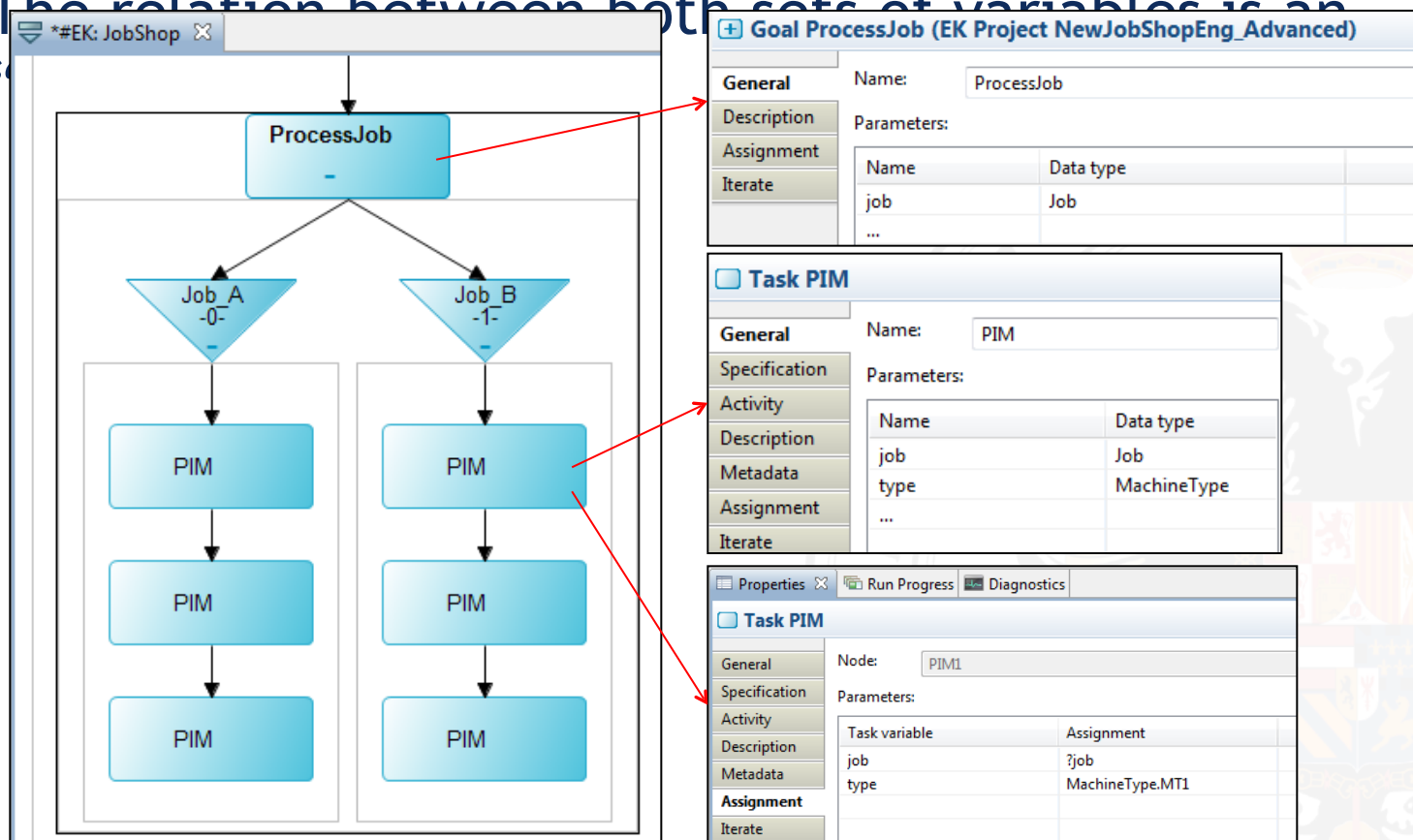
- A goal may have alternative decompositions guided by conditions

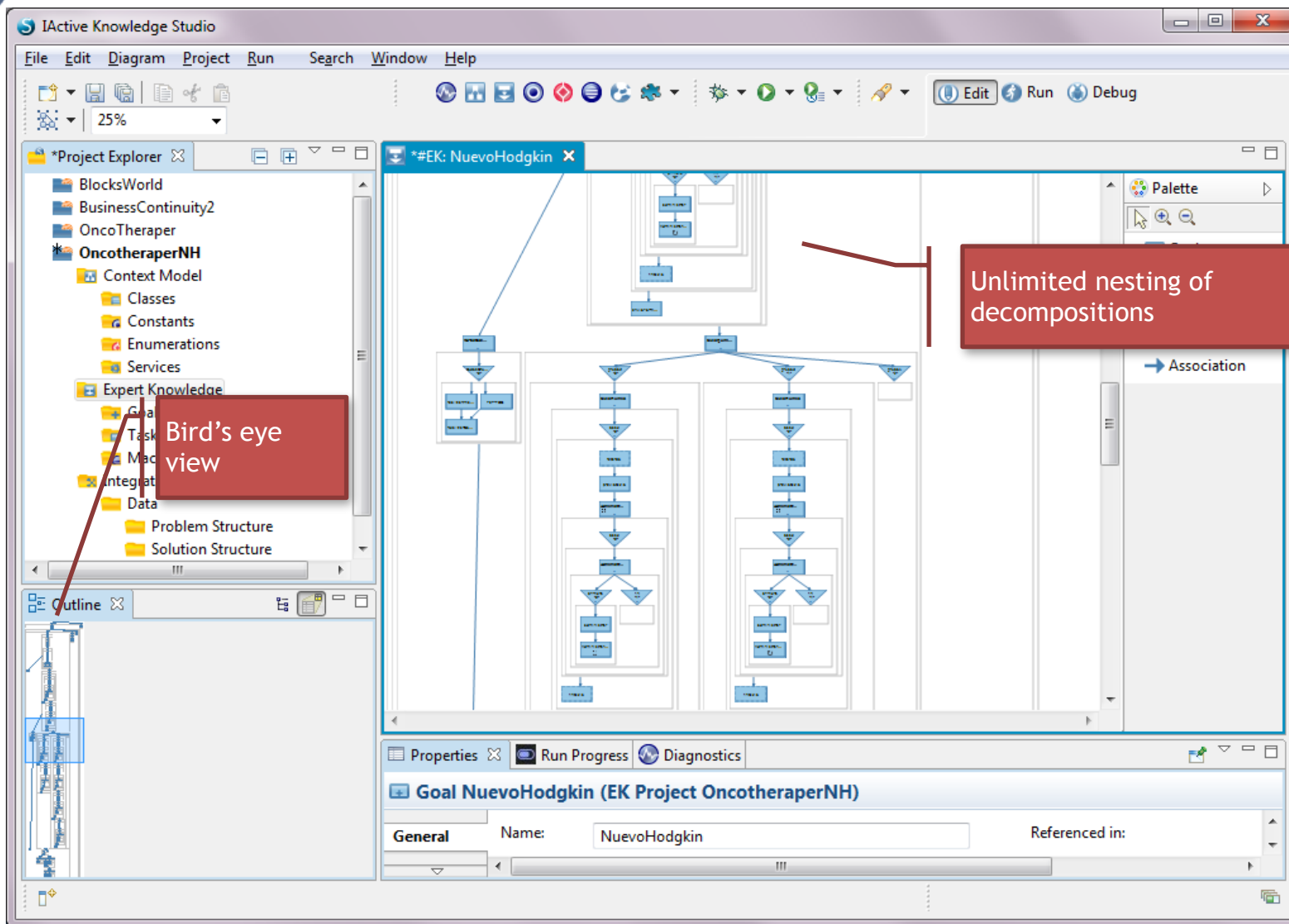
- Tasks may also have conditions



- Goals may have input parameters which could influence their achievement. And so do its constituent Tasks and SubGoals

- The relation between both sets of variables is an

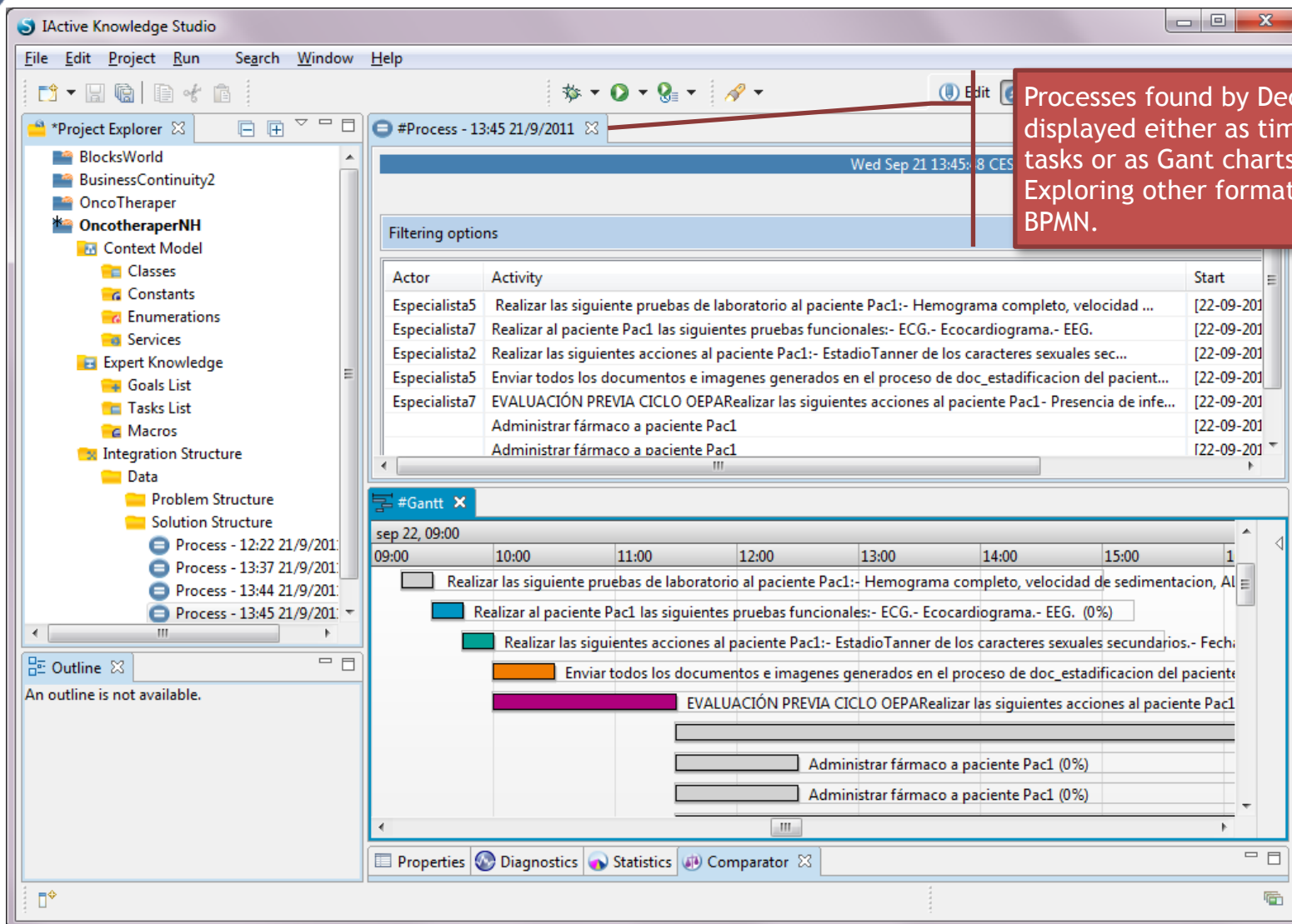




The screenshot displays the IActive Knowledge Studio interface. The main workspace shows a hierarchical goal decomposition diagram for the goal 'NuevoHodgkin'. The diagram consists of multiple nested boxes, each representing a goal, which are further decomposed into smaller goals and tasks. A red callout box with the text 'Unlimited nesting of decompositions' points to the nested structure of the diagram.

On the left side, the 'Project Explorer' pane shows a tree structure of the project 'OncoteraperNH', including folders for 'Context Model', 'Classes', 'Constants', 'Enumerations', 'Services', 'Expert Knowledge', 'Goal', 'Task', 'Mac', 'Integrat', 'Data', 'Problem Structure', and 'Solution Structure'. A red callout box with the text 'Bird's eye view' points to the 'Goal' folder.

At the bottom, the 'Properties' pane shows the details for the selected goal 'NuevoHodgkin' (EK Project OncoteraperNH), including its name and referenced information.

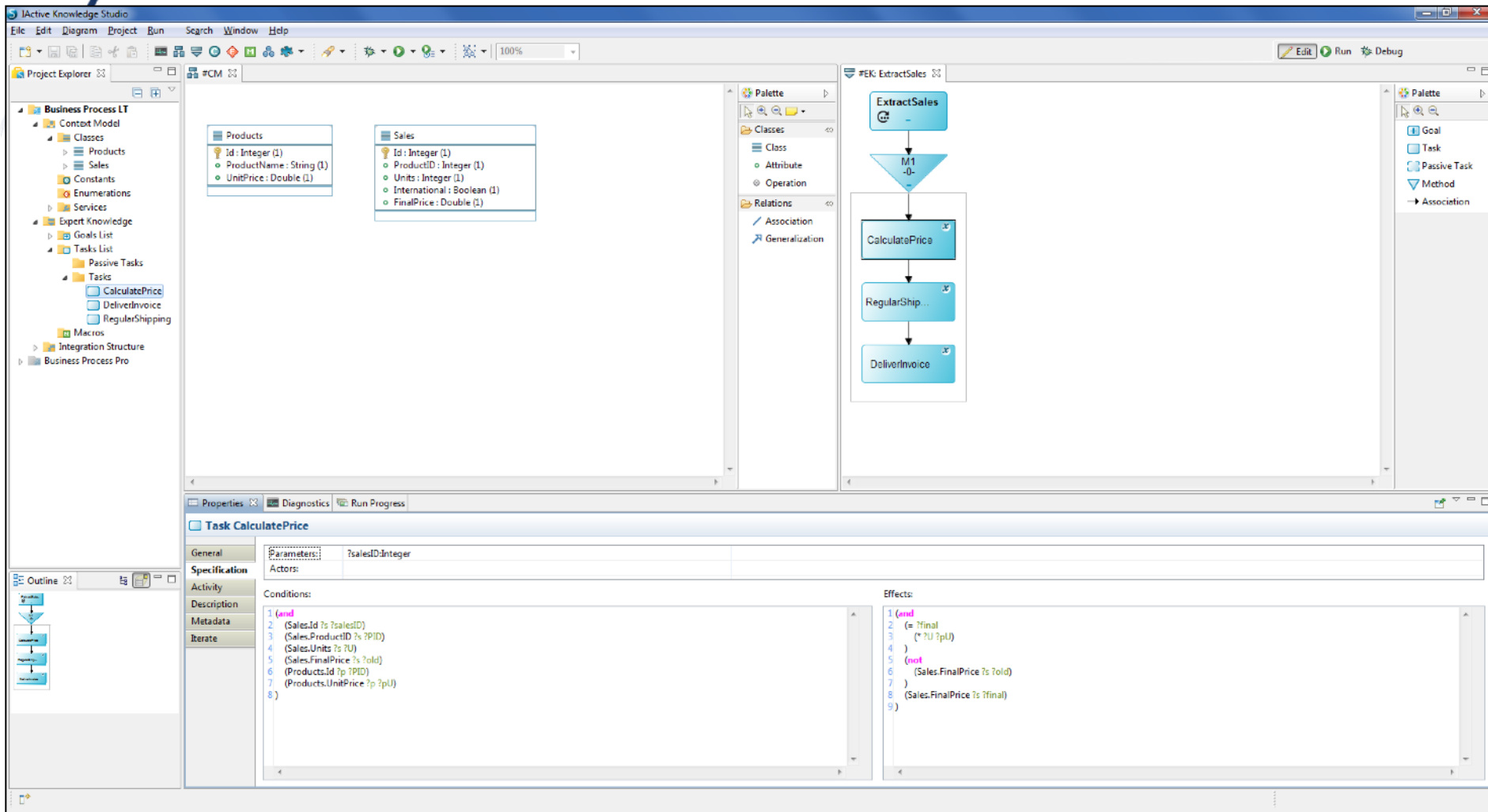


The screenshot displays the IActive Knowledge Studio interface. On the left, the Project Explorer shows a tree structure with folders like BlocksWorld, BusinessContinuity2, OncoTheraper, and OncotheraperNH. The main workspace is divided into two panes. The top pane, titled '#Process - 13:45 21/9/2011', shows a table of process data. The bottom pane, titled '#Gantt', shows a Gantt chart for the same process. A red box highlights the top pane, and a red arrow points from the text box to it.

Processes found by Decisor may be displayed either as timed sequences of tasks or as Gantt charts for validation. Exploring other formats like XPD/ BPMN.

Actor	Activity	Start
Especialista5	Realizar las siguiente pruebas de laboratorio al paciente Pac1:- Hemograma completo, velocidad ...	[22-09-2011]
Especialista7	Realizar al paciente Pac1 las siguientes pruebas funcionales:- ECG.- Ecocardiograma.- EEG.	[22-09-2011]
Especialista2	Realizar las siguientes acciones al paciente Pac1:- EstadíoTanner de los caracteres sexuales sec...	[22-09-2011]
Especialista5	Enviar todos los documentos e imagenes generados en el proceso de doc_estadificacion del pacient...	[22-09-2011]
Especialista7	EVALUACIÓN PREVIA CICLO OEPARealizar las siguientes acciones al paciente Pac1- Presencia de infe...	[22-09-2011]
	Administrar fármaco a paciente Pac1	[22-09-2011]
	Administrar fármaco a paciente Pac1	[22-09-2011]

The Gantt chart shows a timeline from 09:00 to 15:00 on September 22, 2011. It displays the duration of each activity as a horizontal bar, with some bars indicating completion percentage (e.g., 0%).



The screenshot displays the Active Knowledge Studio interface. The main workspace shows a process diagram for 'ExtractSales' with the following steps:

```

graph TD
    ExtractSales[ExtractSales] --> M1{M1  
-0-}
    M1 --> CalculatePrice[CalculatePrice]
    CalculatePrice --> RegularShip[RegularShip...]
    RegularShip --> DeliverInvoice[DeliverInvoice]
  
```

The left sidebar shows the Project Explorer with the following structure:

- Business Process LT
 - Context Model
 - Classes
 - Products
 - Id: Integer (1)
 - ProductName: String (1)
 - UnitPrice: Double (1)
 - Sales
 - Id: Integer (1)
 - ProductID: Integer (1)
 - Units: Integer (1)
 - International: Boolean (1)
 - FinalPrice: Double (1)
 - Constants
 - Enumerations
 - Services
 - Expert Knowledge
 - Goals List
 - Tasks List
 - Passive Tasks
 - CalculatePrice
 - DeliverInvoice
 - RegularShipping
 - Macros
 - Integration Structure
 - Business Process Pro

The bottom panel shows the configuration for the 'Task CalculatePrice' task:

Task CalculatePrice

General: Parameters: ?salesID:Integer

Specification:

Actors:

Conditions:

```

1 (and
2   (Sales.Id ?s salesID)
3   (Sales.ProductID ?s ?PID)
4   (Sales.Units ?s ?U)
5   (Sales.FinalPrice ?s ?old)
6   (Products.Id ?p ?PID)
7   (Products.UnitPrice ?p ?pU)
8 )

```

Effects:

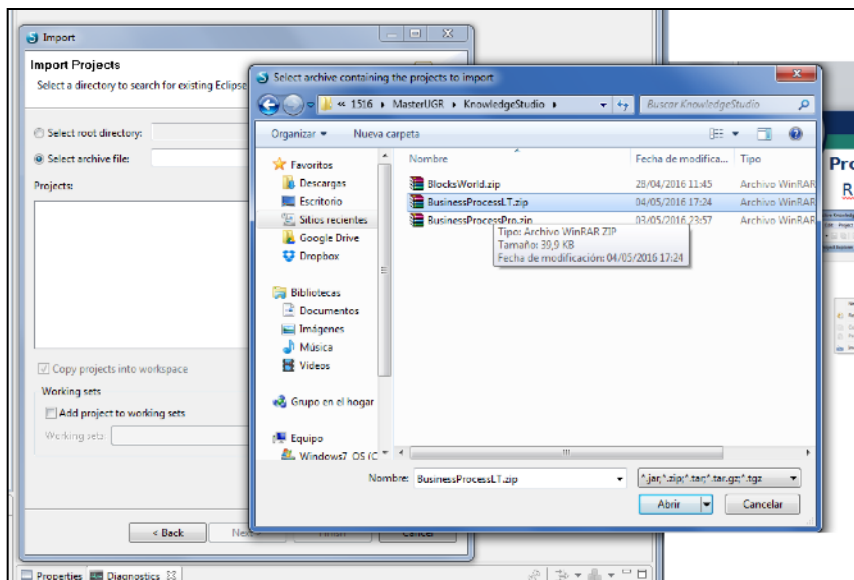
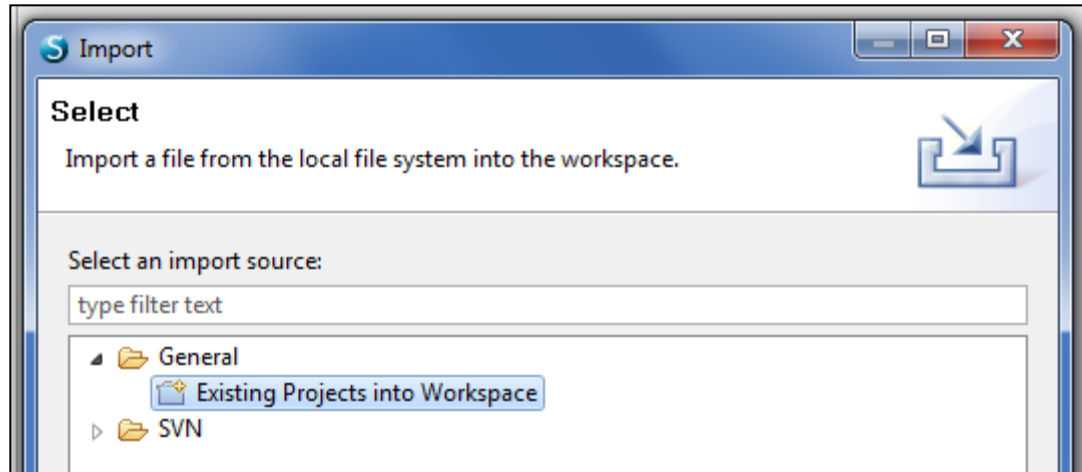
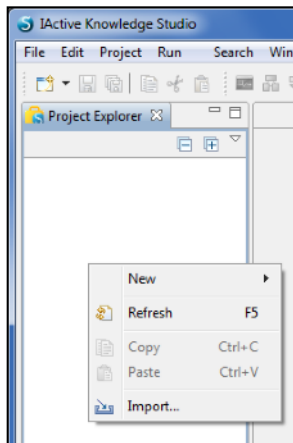
```

1 (and
2   (= ?final
3     (* ?U ?pU)
4   )
5   (not
6     (Sales.FinalPrice ?s ?old)
7   )
8   (Sales.FinalPrice ?s ?final)
9 )

```

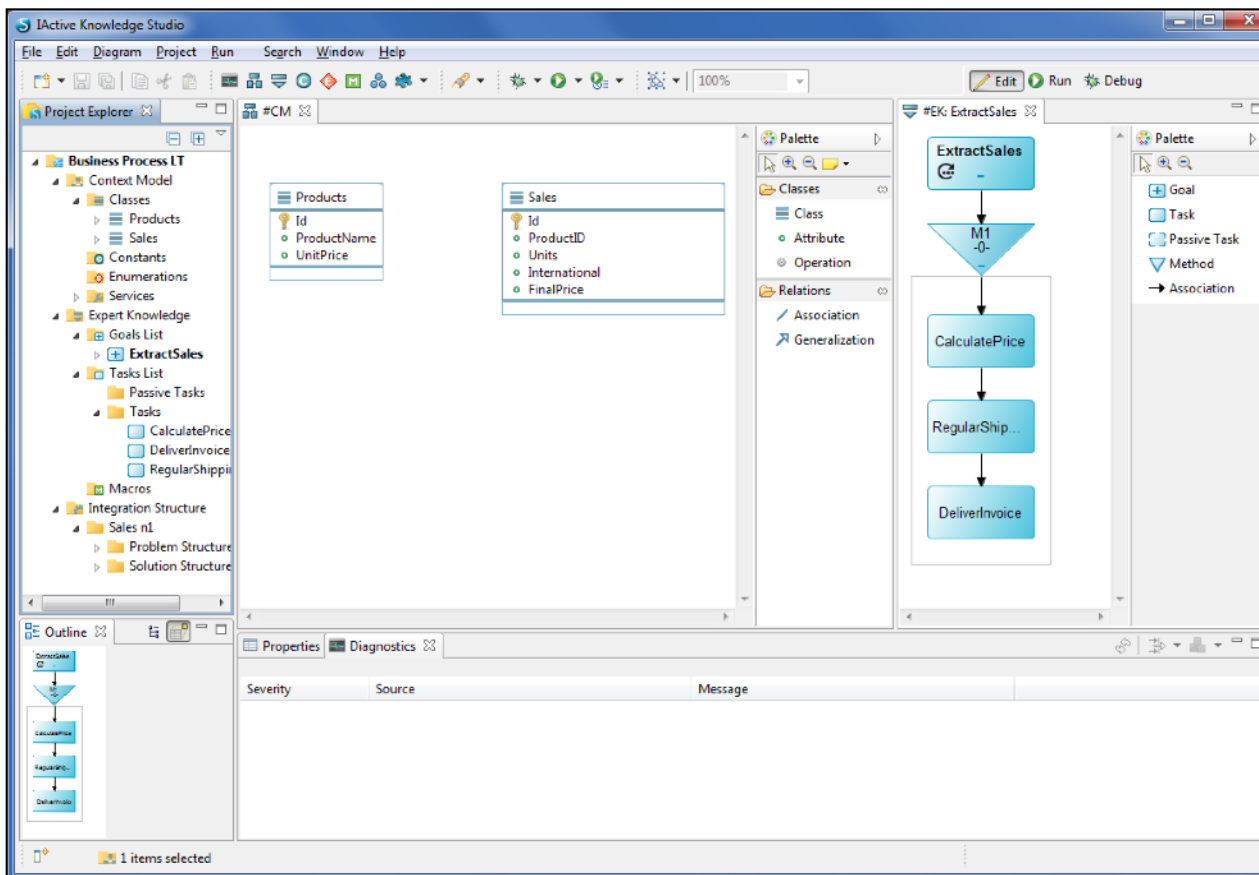
Project 1: Tiny Business Process

- Right click on “Project Explorer” > “Import” > “Existing Projects into Workspace” > “Select archive files” > “BusinessProcessLT.zip”



Project 1: Tiny Business Process

- Expand components “Context Model”
- Expand “Integration Structure” > “Sales n1” > “Problem Structure” > “Local Data”
- Expand “Expert knowledge”

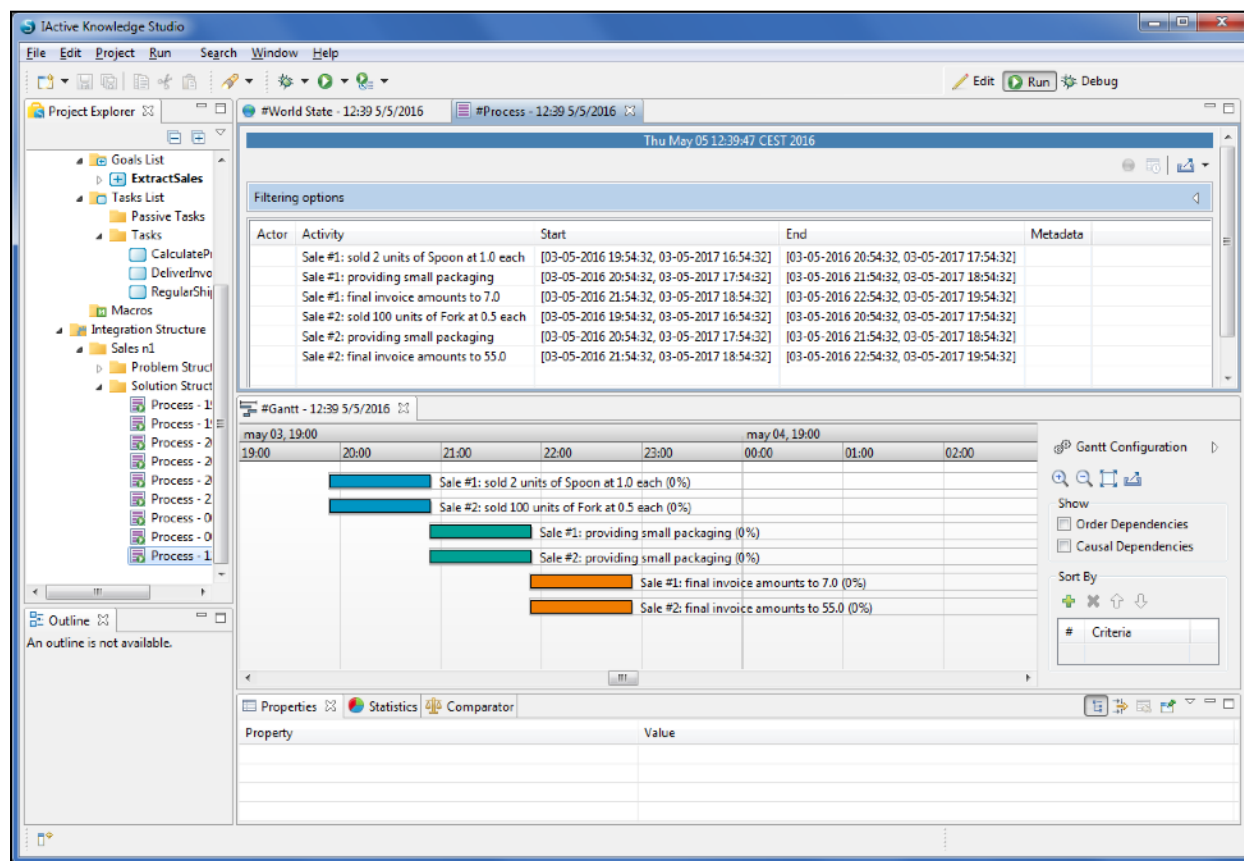
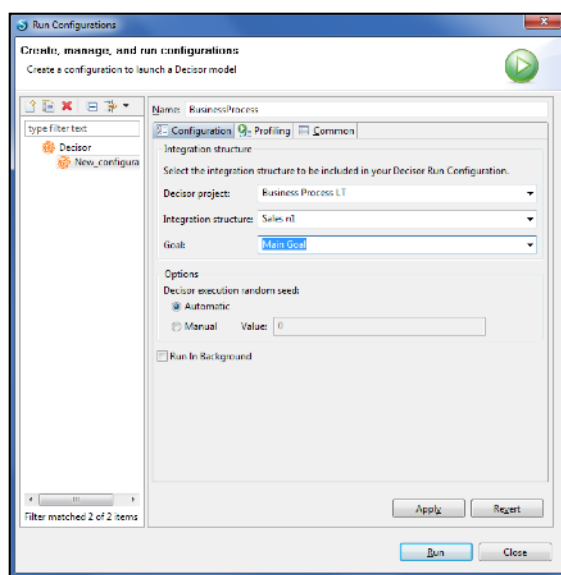
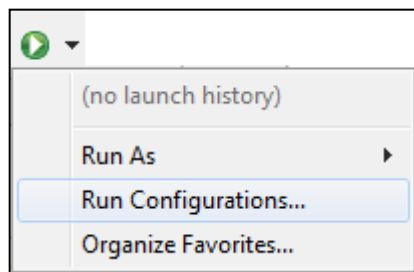


#Local Data			
Tables			
Products(2)			
Sales(2)			
	Id(*)	ProductName(*)	UnitPrice(*)
	1	"Spoon"	1.0
	2	"Fork"	0.5
	...		

#Local Data					
Tables					
Products(2)					
Sales(2)					
	Id(*)	ProductID(*)	Units(*)	International(*)	FinalPrice(*)
	1	1	2	false	0.0
	2	2	100	true	0.0
	...				

Project 1: Tiny Business Process

- Create a run environment > “Apply” > “Run”
- Both sales follow the same process and schedule



Project 2: Adaptive Business Process

- “Project Explorer” > “Import”
> “BusinessProcessPRO(incomp).zip”

1. Create new Tasks

1. LargeShipping
2. RequestCustomsClearance

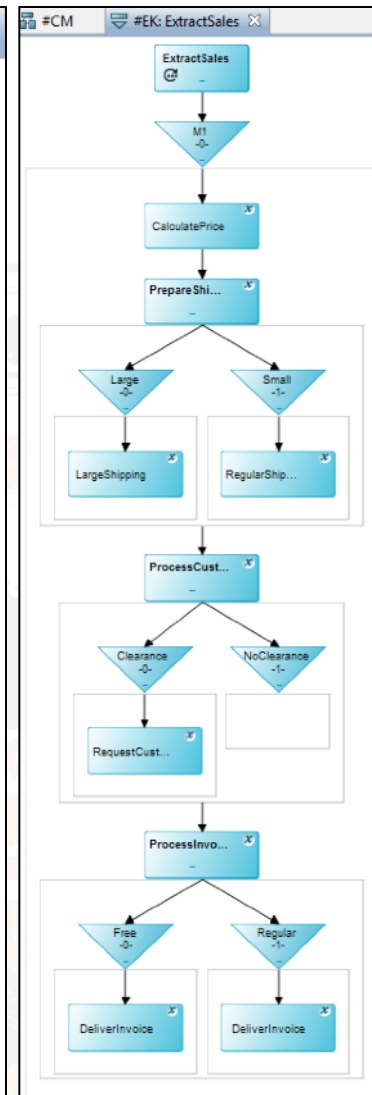
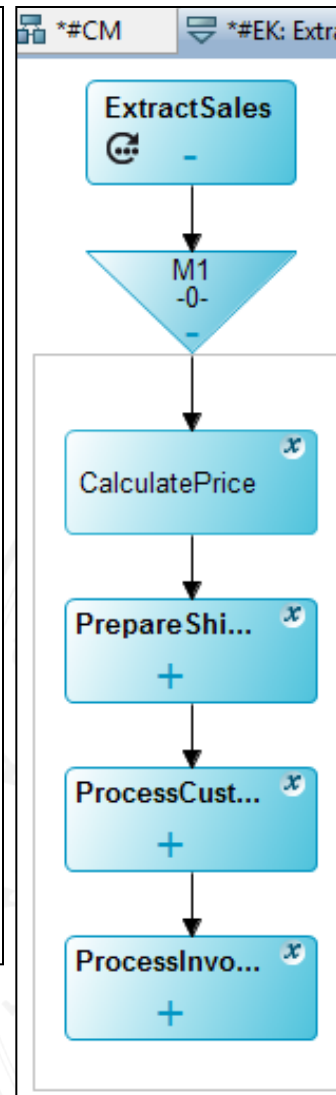
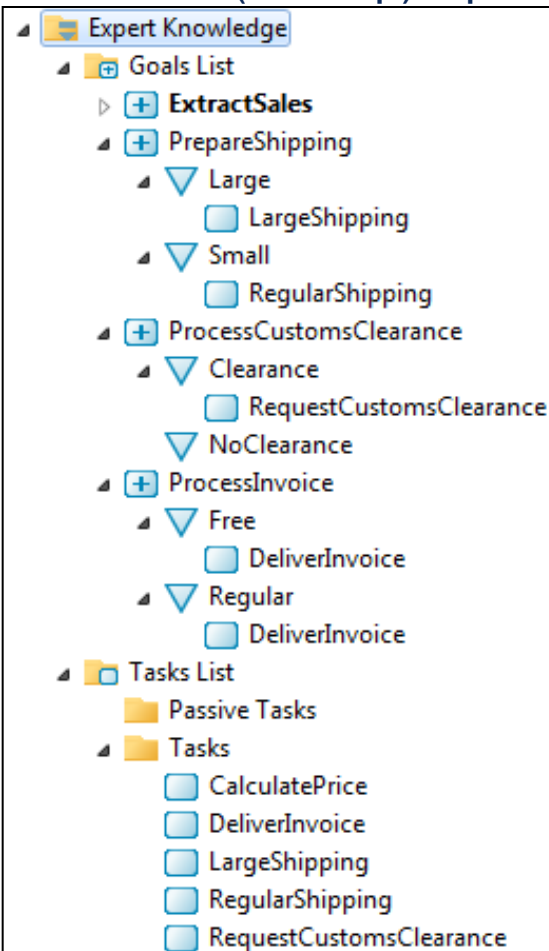
2. Create new Goals

1. PrepareShipping
2. ProcessCustomsClearance
3. ProcessInvoice

3. Create two methods for each new Goal

1. PrepareShipping
 1. Large (>2 u.)
 2. Small (<= 2u.)
2. ProcessCustomsClearance
 1. Clearance (when intl)
 2. NoClearance
3. ProcessInvoice
 1. Free (> 20€)
 2. Regular (<= 20€ +5€ delivery)

**ADAPT PROCESSES
TO THE CONTEXT**



Project 2: Adaptive Business Process

- Redesign subprocesses to fit the goals of adaptive generation
- Create a new run environment
- Each sale follows a different process, schedule and final price

#Local Data

Tables				
Products(2)				
Sales(2)				

	Id(*)	ProductName(*)	UnitPrice(*)
	1	"Spoon"	1.0
	2	"Fork"	0.5
	...		

#Local Data

Tables						
Products(2)						
Sales(2)						

	Id(*)	ProductID(*)	Units(*)	International(*)	FinalPrice(*)
	1	1	2	false	0.0
	2	2	100	true	0.0
	...				

#Process - 13:21 5/5/2016

Thu May 05 13:21:45 CEST 2016

Filtering options

Actor	Activity	Start	End	Metadata
	Sale #1: sold 2 units of Spoon at 1.0 each	[03-05-2016 19:54:32, 03-05-2017 16:54:32]	[03-05-2016 20:54:32, 03-05-2017 17:54:32]	
	Sale #1: providing small packaging	[03-05-2016 20:54:32, 03-05-2017 17:54:32]	[03-05-2016 21:54:32, 03-05-2017 18:54:32]	
	Sale #1: final invoice amounts to 7.0	[03-05-2016 21:54:32, 03-05-2017 18:54:32]	[03-05-2016 22:54:32, 03-05-2017 19:54:32]	
	Sale #2: sold 100 units of Fork at 0.5 each	[03-05-2016 19:54:32, 03-05-2017 15:54:32]	[03-05-2016 20:54:32, 03-05-2017 16:54:32]	
	Sale #2: providing large packaging	[03-05-2016 20:54:32, 03-05-2017 16:54:32]	[03-05-2016 21:54:32, 03-05-2017 17:54:32]	
	Sale #2: requesting Customs clearance	[03-05-2016 21:54:32, 03-05-2017 17:54:32]	[03-05-2016 22:54:32, 03-05-2017 18:54:32]	
	Sale #2: final invoice amounts to 50.0	[03-05-2016 22:54:32, 03-05-2017 18:54:32]	[03-05-2016 23:54:32, 03-05-2017 19:54:32]	

#Gantt - 13:21 5/5/2016

may 03, 19:00					may 04, 19:00			
19:00	20:00	21:00	22:00	23:00	00:00	01:00	02:00	03:00
Sale #1: sold 2 units of Spoon at 1.0 each (0%)								
Sale #2: sold 100 units of Fork at 0.5 each (0%)								
Sale #1: providing small packaging (0%)								
Sale #2: providing large packaging (0%)								
Sale #1: final invoice amounts to 7.0 (0%)								
Sale #2: requesting Customs clearance (0%)								
Sale #2: final invoice amounts to 50.0 (0%)								

Project 2: Adaptive Business Process

- Export project as “BusinessProcessPro.zip”
- Take screenshots of both process and Gantt chart
- Send them to the teacher < **15/JUN/2018**

