LAAS-CNRS

Université de Toulouse

A UML-based method for risk analysis of human-robot interactions

Damien Martin-Guillerez, **Jérémie Guiochet**,
David Powell and Christophe Zanon

2nd International Workshop on Software Engineering for Resilient Systems,
April 13-16, 2010, Birkbeck College (London, United Kingdom)

Introduction

- Increasing safety concerns: computer controlled safety critical systems emerge in many areas (automotive, shipping, medical applications, industrial processes, etc.)
- Increasing complexity of systems: necessity for system modelling, based on languages with high level of expressiveness
- Correlation is needed between system modelling and safety analysis

Motivations (1)

- Related work on model based/driven safety analysis methods and tools:
  - **Based on design models** with different description languages (ex. Statemate, SCADE, Altarica, etc.)
  - Perform **automatic analysis** (sequence generation, fault tree and FMEA synthesis, model checking, etc.)
  - Many associated **tools** (Cecilia OCAS ©Dassault, HIP-HOPS © Univ. of Hull., Statemate STSA © IBM, COMPARE © FBK, etc.)

Motivations (2)

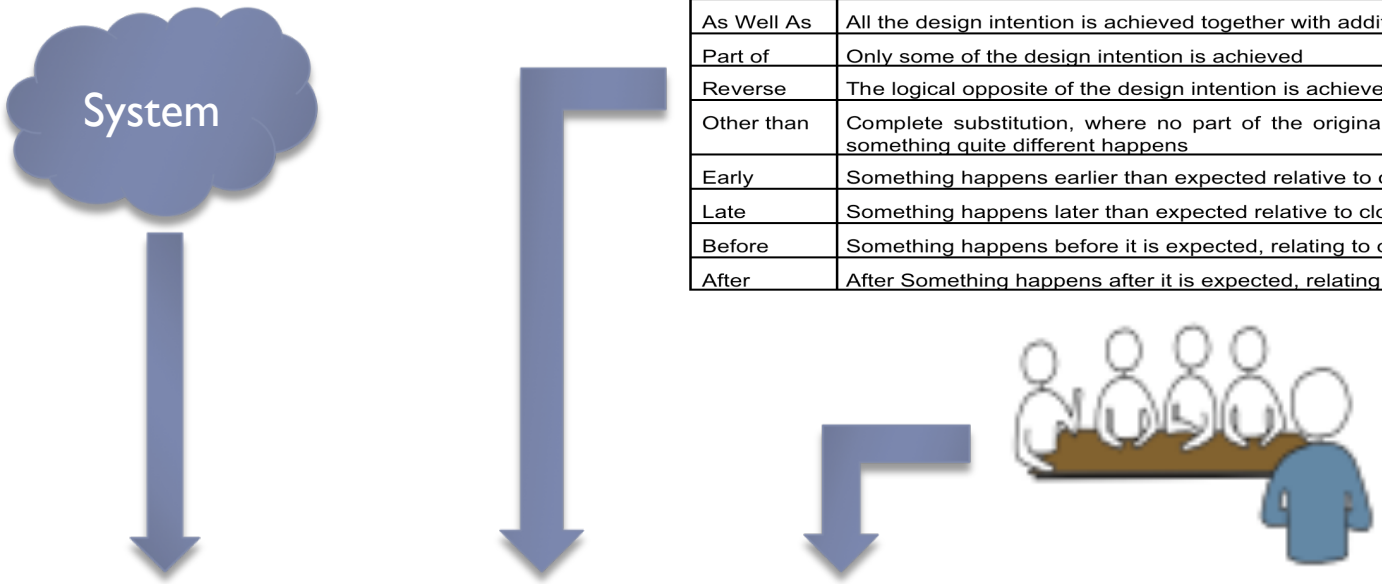
- **Few works on specification or requirement** modelling and safety analysis
  - Mainly research papers with **no associated tools**
  - Languages and techniques difficult to understand for non specialists
- Applicability of existing model-based methods to **safety critical autonomous systems** is limited due to:
  - Multifunction/task
  - Unstructured environment
  - Decisional layer
  - Human factors
- Proposal of a generic, usable and systematic method for the analysis of deviations at the first step of the development process -> based on HAZOP and UML

Why HAZOP and UML ?

- HAZOP (HAZard OPerability)
  - Developed at the beginning of the 70's and is a well known technique
  - Identify hazards and propose recommendations with low level of details of design
  - Based on brainstorming done by a group of experts
  - Guidewords can be adapted according to domain and the case study
- UML (Unified Modeling Language)
  - De facto standard
  - Usage diagrams (Use case and sequence diagrams) are easily understandable by non-experts
  - Diagrams can also be used for development process

HAZOP principle

No/None	Complete negation of the design intention No part of the intention is achieved and nothing else happens
More	Quantitative increase
Less	Quantitative decrease
As Well As	All the design intention is achieved together with additions
Part of	Only some of the design intention is achieved
Reverse	The logical opposite of the design intention is achieved
Other than	Complete substitution, where no part of the original intention is achieved but something quite different happens
Early	Something happens earlier than expected relative to clock time
Late	Something happens later than expected relative to clock time
Before	Something happens before it is expected, relating to order or sequence
After	After Something happens after it is expected, relating to order or sequence



- Element X guideword = deviation
- *Pressure* X *More* = "too much pressure"

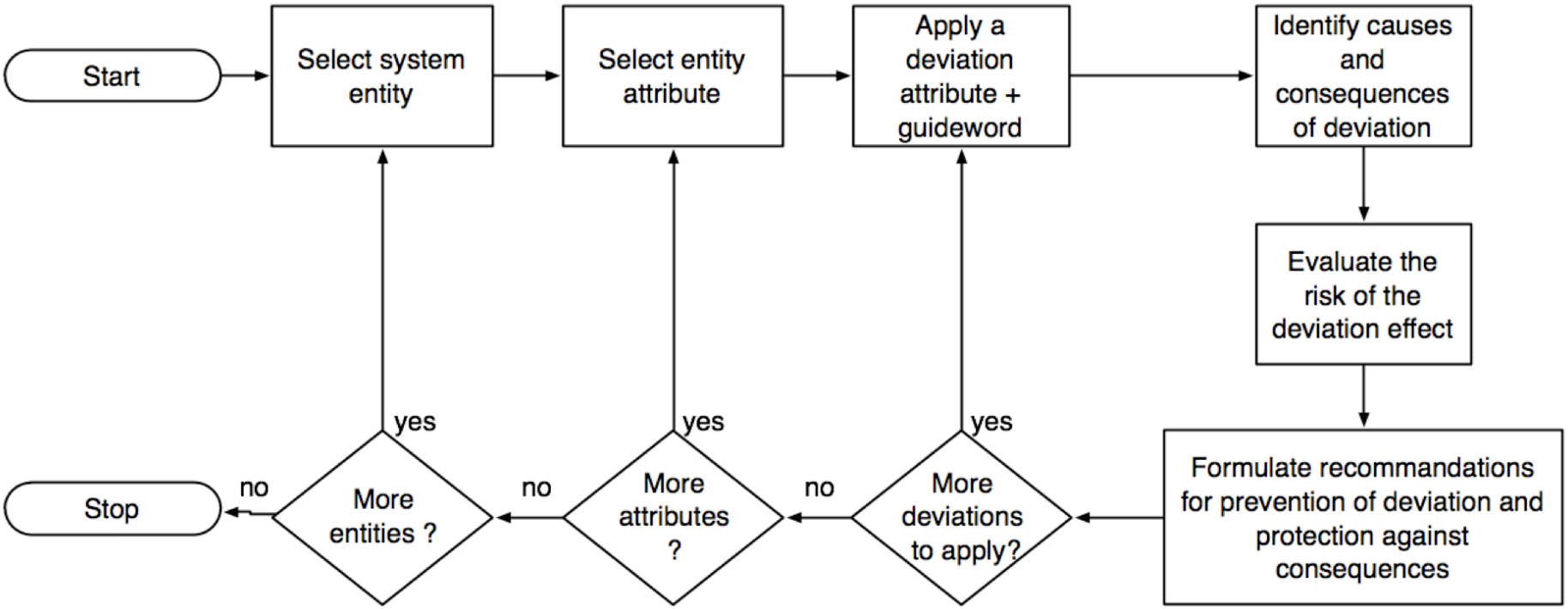
HAZOP tables

No/None	Complete negation of the design intention No part of the intention is achieved and nothing else happens
More	Quantitative increase
Less	Quantitative decrease
As Well As	All the design intention is achieved together with additions
Part of	Only some of the design intention is achieved
Reverse	The logical opposite of the design intention is achieved
Other than	Complete substitution, where no part of the original intention is achieved but something quite different happens
Early	Something happens earlier than expected relative to clock time
Late	Something happens later than expected relative to clock time
Before	Something happens before it is expected, relating to order or sequence
After	After Something happens after it is expected, relating to order or sequence

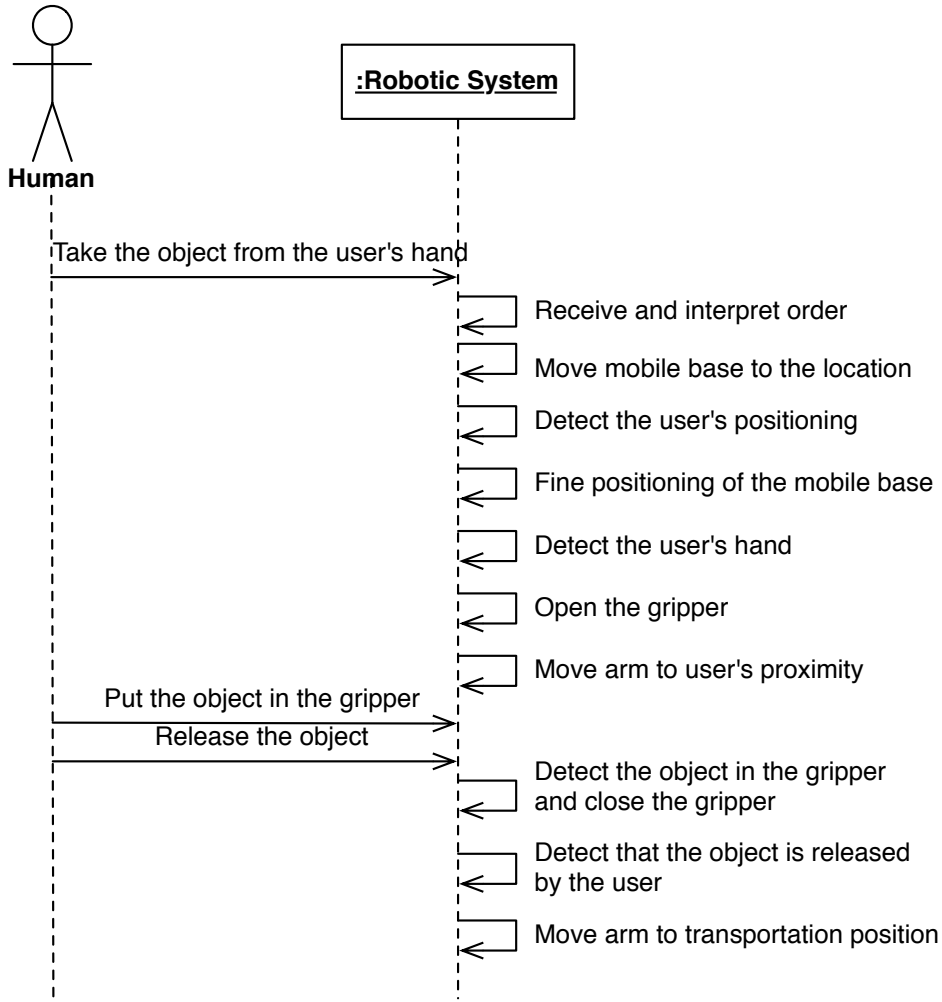
Study title:							Page: of		
Drawing no.:		Rev no.:					Date:		
HAZOP team:							Meeting date:		
Part considered:									
Design intent:		Material: Source:		Activity: Destination:					
No.	Guide-word	Element	Deviation	Possible causes	Conse-quences	Safeguards	Comments	Actions required	Action allocated to

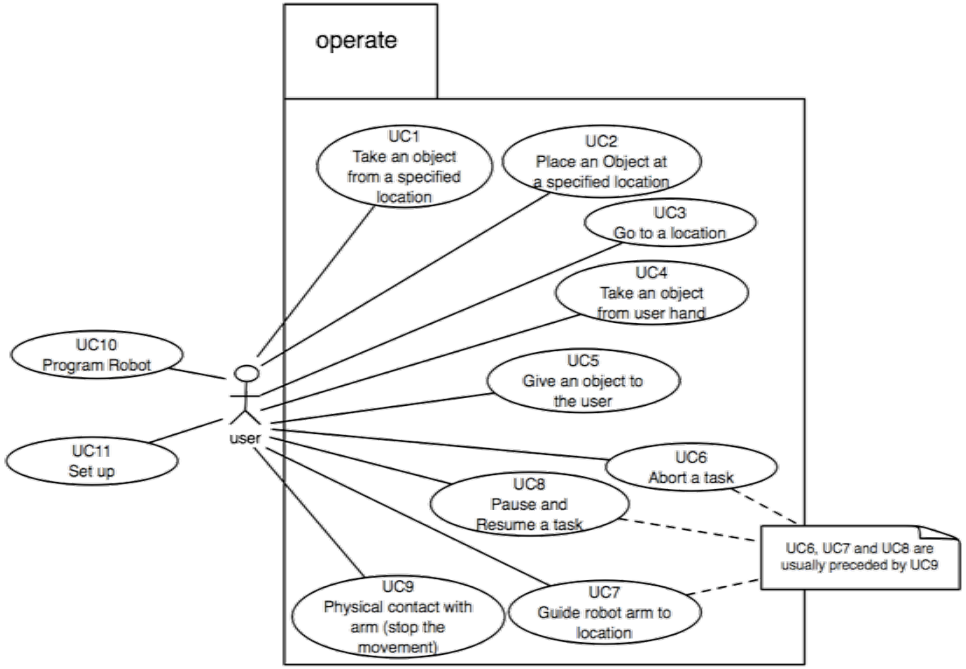
– Source: IEC 61882

HAZOP process adaptation



UML entities and attributes for HAZOP

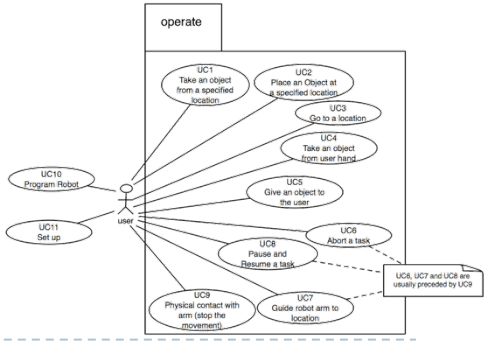




- UML Use case and sequence diagrams

UML **use cases** attributes for HAZOP

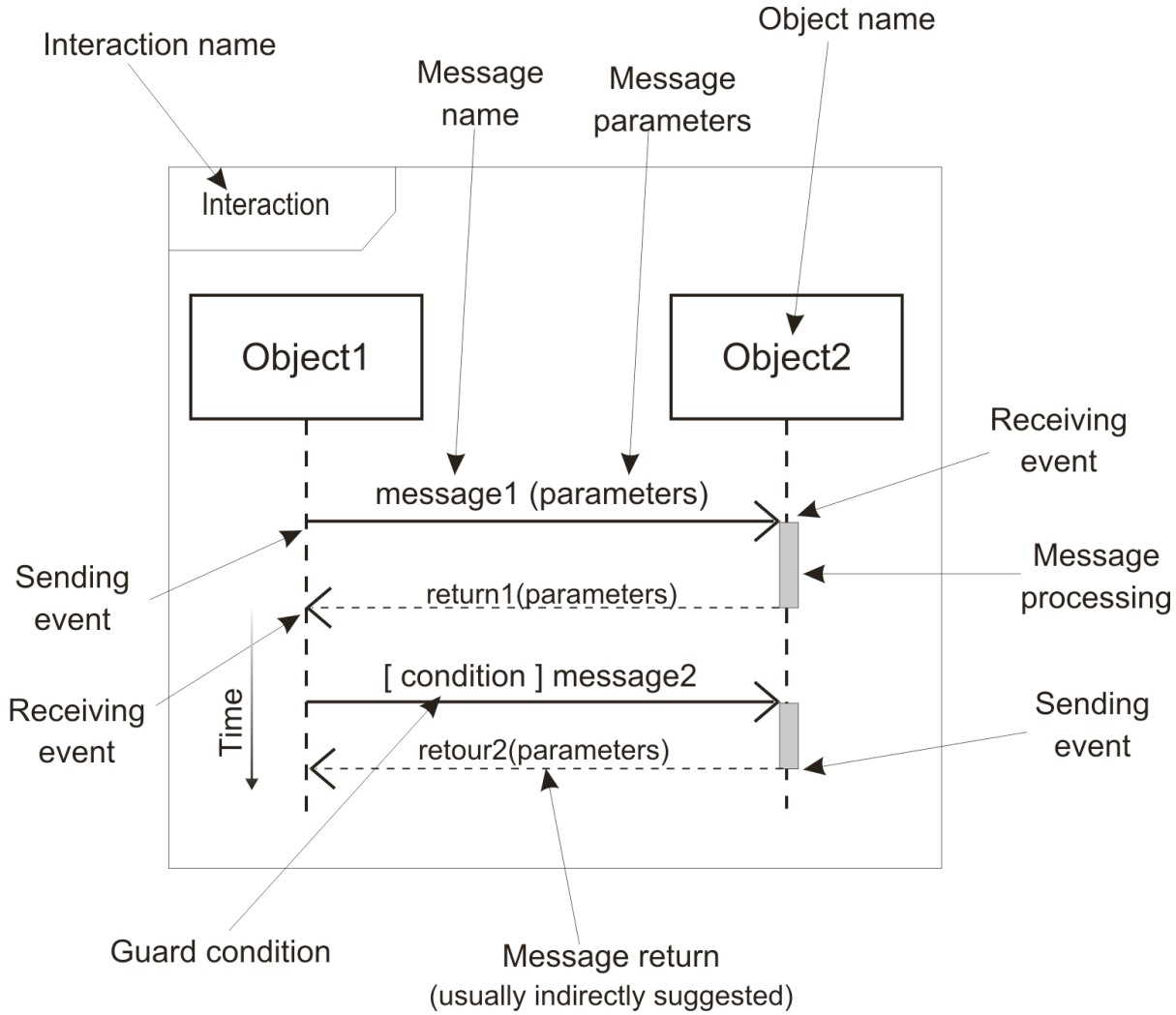
Use case specification	
Use case name	The name of the use case provides a unique identifier
Abstract	Describes the interaction that occurs in the main scenario of the use case
Preconditions	Conditions that must be satisfied before the use case can be executed — they are part of the contract between the use case and the outside world
Postconditions	Conditions that must be satisfied after the use case has been completed successfully
Invariants	Conditions that must be fulfilled throughout the use case execution



HAZOP guidewords adaptation for UML use case

Entity = Use Case		
Attribute	**Guideword**	**Interpretation**
Preconditions / Postconditions / Invariants	No/non e	The condition is not evaluated and can have any valu e
	Other than	The condition is evaluated true whereas it is false The condition is evaluated false whereas it is true
	As well as	The condition is correctly evaluated but other unexpected conditions are true
	Part of	The condition is partially evaluated Some conditions are missing
	Early	The condition is evaluated earlier than required (other condition(s) should be tested before) The condition is evaluated earlier than required for correct synchronization with the environment
	Late	The condition is evaluated later than required (condition(s) depending on this one should have already been tested) The condition is evaluated later than required for correct synchronization with the environment

UML **sequence diagram** attributes



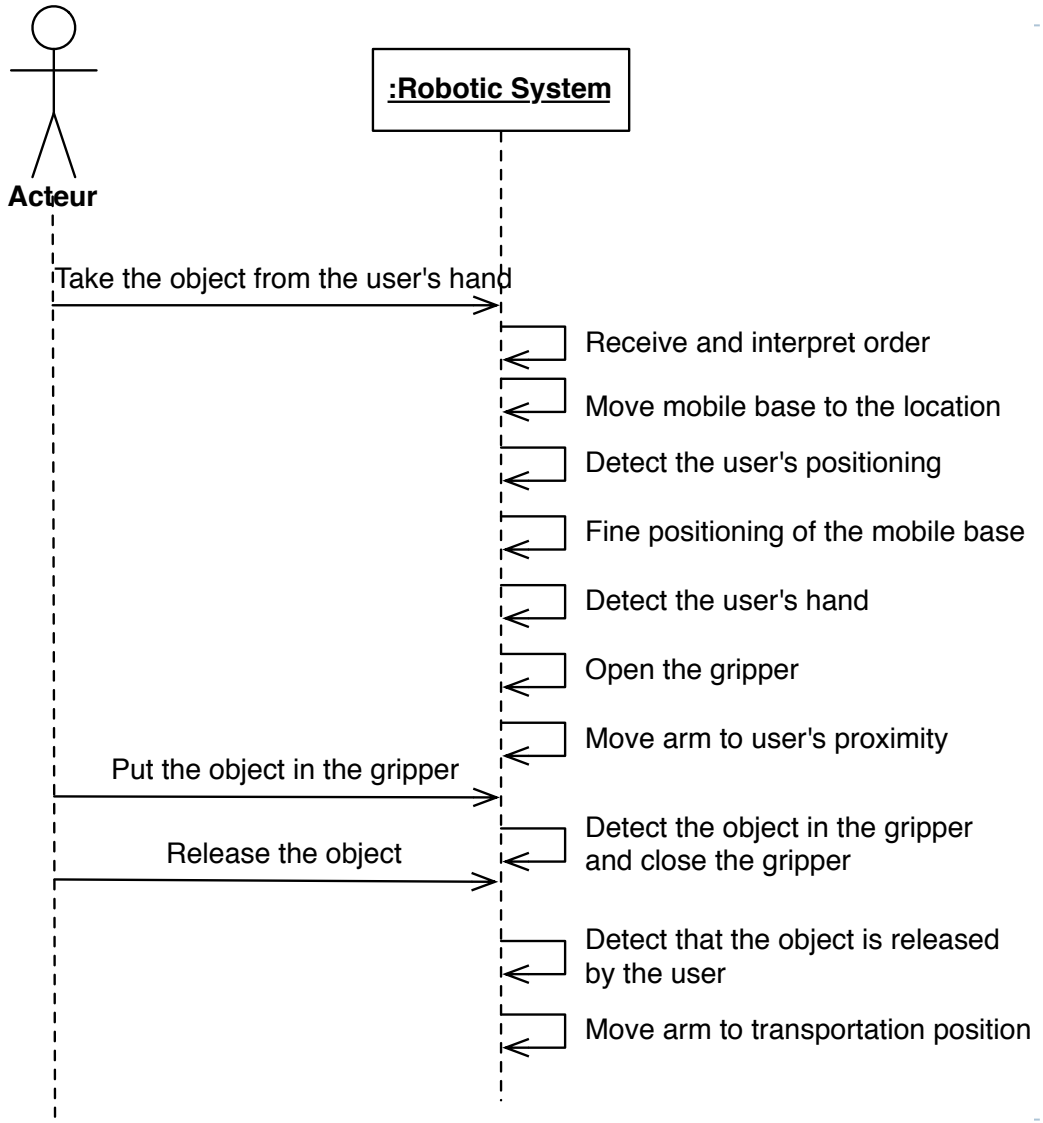
HAZOP guidewords adaptation for UML sequence diagram

Entity = Sequence Diagram		
Attribute	**Guideword**	**Interpretation**
Predecessors / successors during interaction	No	Message is not sent
	Other than	Unexpected message is sent
	As well as	Message is sent as well as another message
	More than	Message sent more often than intended
	Less than	Message sent less often than intended
	Before	Message sent before intended
	After	Message sent after intended
	Part of	Only a part of a set of messages is sent
	Reverse	Reverse order of expected messages
Message timing	As well a s	Message sent at correct time and also at incorrect tim e
	Early	Message sent earlier than intended time
	Later	Message sent later than intended time
Sender / receiver objects	No	Message sent to but never received by intended objec t
	Other than	Message sent to wrong object
	As well as	Message sent to correct object and also an incorrect object
	Reverse	Source and destination objects are reversed
	More	Message sent to more objects than intended
	Less	Message sent to fewer objects than intended

HAZOP guidewords adaptation for UML sequence diagram (2)

Message condition	No/non e	The condition is not evaluated and can have any value (omission)
	Other than	The condition is evaluated true whereas it is false, or vice versa (commission)
	As well as	The condition is well evaluated but other unexpected conditions are true
	Part of	Only a part of condition is correctly evaluated
	Late	The condition is evaluated later than required (other dependent condition(s) have been tested before) The condition is evaluated later than correct synchronization with the environment
Message parameters / return parameters	No/Non e	Expected parameters are never set / returned
	More	Parameters values are higher than intended
	Less	Parameters values are lower than intended
	As Well As	Parameters are also transmitted with unexpected ones
	Part of	Only some parameters are transmitted Some parameters are missing
	Other than	Parameter type / number are different from those expected by the receiver

Example of UML-HAZOP application



Example of UML-HAZOP application (2)

Project : PHRIENDS **Date: June-01-2008**
HAZOP number : UC4/SD4 **Prepared by: Ofaina Taofifenua**
Entity : Sequence Diagram 4 (sd4) "Take an object from the user's hand" **Revised by: Jérémie Guiochet**
Approved by:

Element (attribute)	Guide word	Deviation	a. Use Case Effect b. Real World Effect	Severity	Possible Causes	Integrity level Requirements	New Safety Requirements	Remarks	Hazard Number
Receive and interpret order (pred/succ)	More than / as well as	The robot receives several different orders	a. Wrong order taken into account b. Wrong task, bad synchro-nization between robot and user, could result in collision	**Moderate**	Failure of H/W for order reception Human error	H/W for order reception should be SIL1	User education and training Define a protocol for communication between user and robot (e.g. acknowledgment messages, user can check interpretation of the order)	Means for communication between robot and user needs to be defined for the PHRIENDS use case (speech, graphical HMI, vision, etc.)	
Put the object in the gripper (pred/succ)	Before	Since the gripper is open the user can give the object to the robot before the latter is ready	a. Bad synchronization between user and robot can cause collision b. The object can fall / The arm and human can collide	**Severe**	Human error	None	The robot should keep the gripper closed until the arm movement is finished	The procedure in the seq. diag. is as follows: the robot opens its gripper then the robot arm moves towards the user hand. Only then the user can place the object in the robot gripper. A safer procedure is: the robot should keep the gripper closed until arm movement is finished -> modify sequence diagram	2, 19, 20

Two case studies

Mobile manipulator (PHRIENDS - FP6 project)

Strolling assistant (MIRAS - ANR Project)

Results

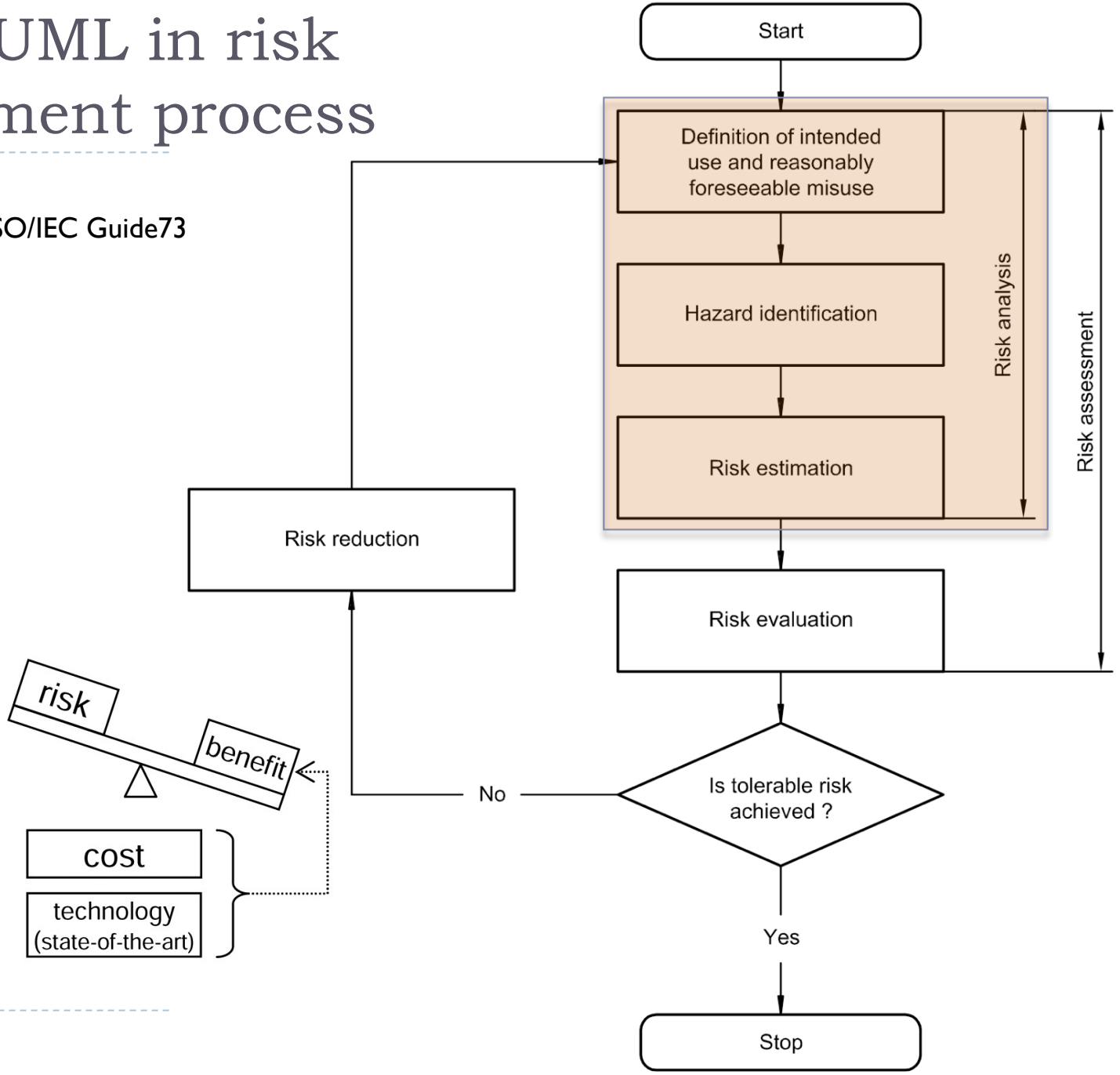
- PHRIENDS project:
  - 1694 deviations considered but only 768 interpreted
  - 21 main hazards (and hazardous situations) identified
  - 18 recommendations for safety
  - Paper study
- MIRAS project:
  - 993 deviations considered but only 297 interpreted
  - 13 main hazards
  - 17 recommendations for safety
  - Prototype#2 is now under construction integrating recommendations

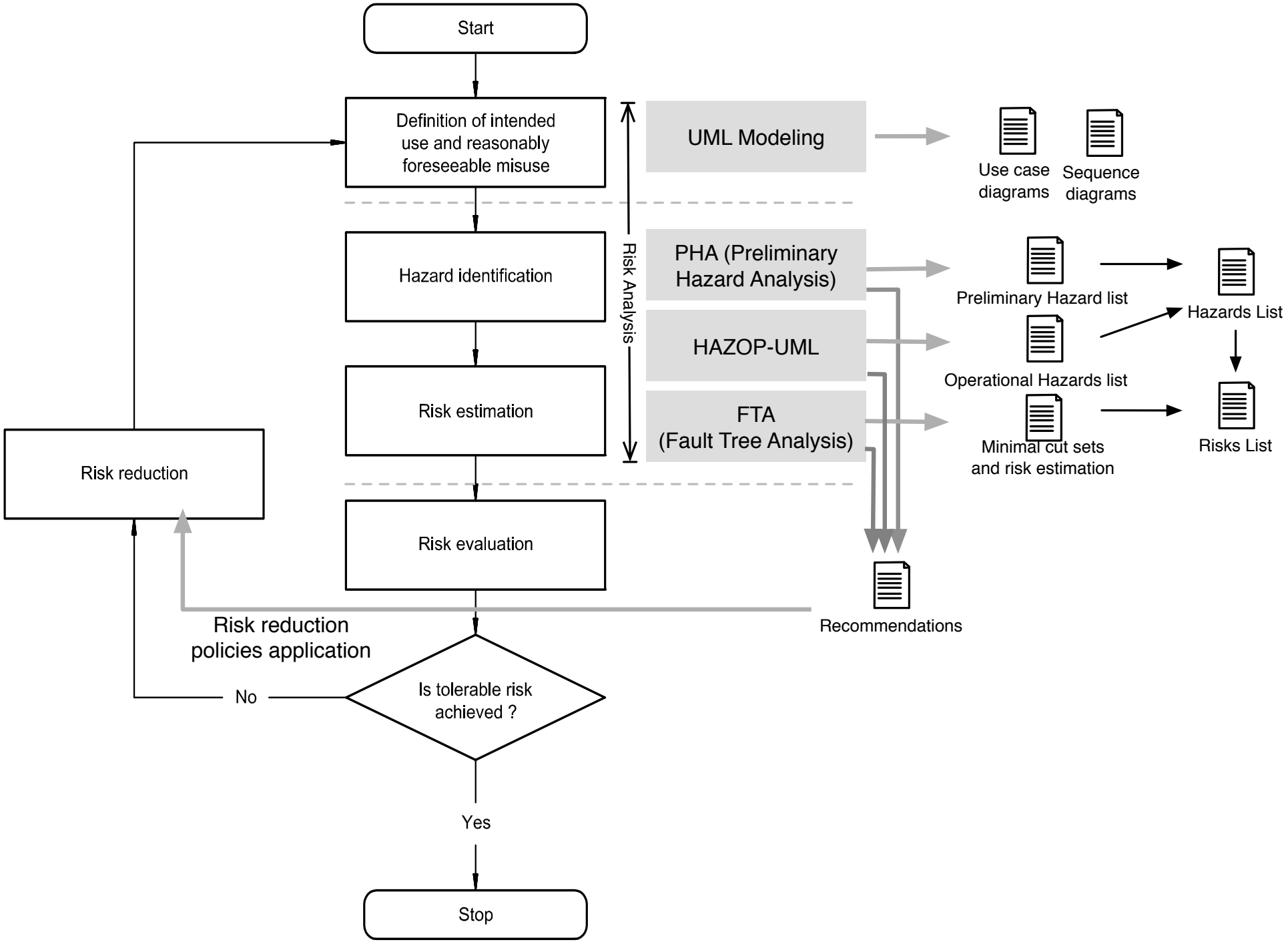
Lessons learnt

- Pros
  - **Integrability** with development process : sharing of the UML model with the development team
  - **Usability**: modelling is limited to 2 diagrams, and flexibility should be improved with consistency checks between modelling and HAZOP tables
  - **Validity**: guidewords selection and interpretation lead to the identification of all operational hazards (compared to a Preliminary Hazard Analysis)
  - **Applicability**: hazard and recommendation lists have been validated by robotics experts and integrated in the design of MIRAS
- Cons
  - Missing hazards: mainly those linked to the use of machinery like electrocution or to the environment like water on floor…
  - Without a tool :
    - Consistency difficult to maintain
    - Difficulties to present the results to experts
    - Repetitive task -> decrease analyst motivation

HAZOP-UML in risk management process

- ISO/CEI Guide 51& SO/IEC Guide73

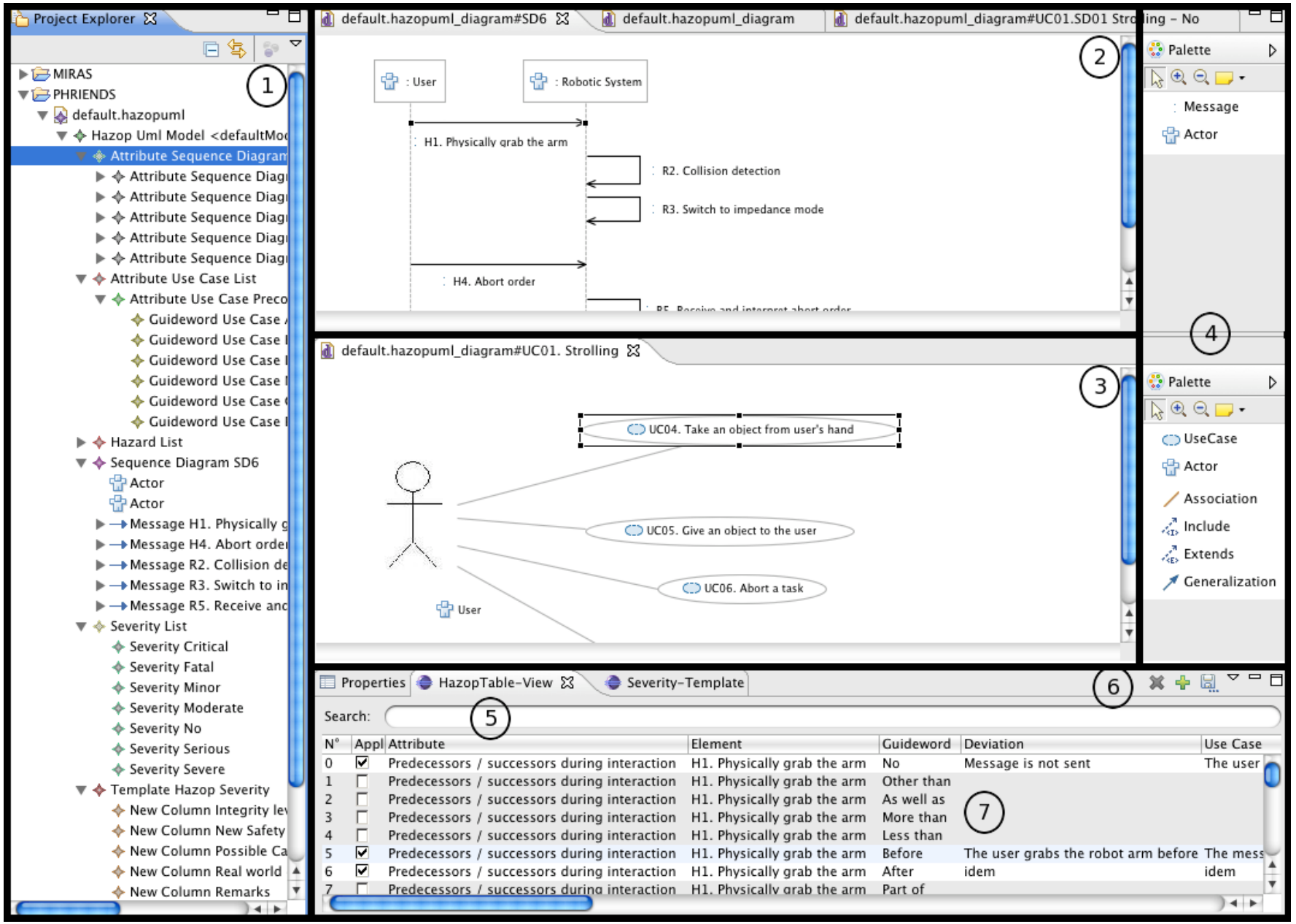



Start
Definition of intended use and reasonably foreseeable misuse
Hazard identification
Risk estimation
Risk evaluation
Risk reduction
Risk Analysis
UML Modeling
PHA (Preliminary Hazard Analysis)
HAZOP-UML
FTA (Fault Tree Analysis)
Use case diagrams
Sequence diagrams
Preliminary Hazard list
Operational Hazards list
Minimal cut sets and risk estimation
Hazards List
Risks List
Recommendations
Risk reduction policies application
Is tolerable risk achieved ?
No
Yes
Stop

Tool development

- Open source
- Developed with as an eclipse plugin (or RCP) using GMF (Graphical Modelling Framework)
- Based on UML2 metamodel
- V0.2 is current version

UMLHAZOP tool v0.2



Next steps

- Integrate same approach with UML *statecharts* including a modelling of *user states/robot operation modes/safety relevant environment states*, and generating deviations with the same guidewords-like approach (under study)
- Complete the development of the tool and application to another robotic system (under study)
- Development of a method for the automatic generation of deviations of scenarios, may be based on *statecharts* modelling (not started)
- Inclusion in the overall safety process dedicated to safety critical autonomous system (under study)

Thank you for your attention