Hazard analysis with HAZOP-UML

Jérémie GUIOCHET
Université Toulouse 3
LAAS-CNRS  TSF
Course planning

• Preliminary Hazard Analysis
  – Risk management & PHA concepts (15min)
  – Application (20min)
• Hazop-UML overview (15min)
• UML
  – Concepts introduction (15min)
  – Application (20 min)
• HAZOP-UML in action
  – Concepts introduction (15min)
  – Application (60min)
RISK MANAGEMENT
Unwanted effects: harm

• **Harm**: physical injury or damage to the health of people, or damage to property or the environment.

• Three attributes of a harm are usually defined: the **nature** of the harm, its **severity** and its **probability** of occurrence.
  – E.g., nature=cut, severity=recoverable with care, probability=frequent
# Nature of harm

<table>
<thead>
<tr>
<th>Human part</th>
<th>Harm nature</th>
</tr>
</thead>
<tbody>
<tr>
<td>Skeletal/Muscular System</td>
<td>Break large bones in primary skeletal structure</td>
</tr>
<tr>
<td></td>
<td>Break small bones in extremities</td>
</tr>
<tr>
<td></td>
<td>Break protective bone structure (e.g., skull, ribs)</td>
</tr>
<tr>
<td></td>
<td>Cut muscle tissue</td>
</tr>
<tr>
<td></td>
<td>Pierce muscle tissue</td>
</tr>
<tr>
<td></td>
<td>Crush muscle tissue</td>
</tr>
<tr>
<td>Organs</td>
<td>Brain concussion</td>
</tr>
<tr>
<td></td>
<td>Pierce lungs, heart, stomach, kidneys, etc.</td>
</tr>
<tr>
<td></td>
<td>Cardiac arrest from electric shock</td>
</tr>
<tr>
<td></td>
<td>Groin trauma</td>
</tr>
<tr>
<td>Circulatory System/Breathing</td>
<td>Arterial/vascular puncture</td>
</tr>
<tr>
<td></td>
<td>Prevent diaphragm action</td>
</tr>
<tr>
<td></td>
<td>Block airways</td>
</tr>
<tr>
<td>Mouth/Nose/Eyes/Ears</td>
<td>Pierce eyes</td>
</tr>
<tr>
<td></td>
<td>Crushing trauma to eyes</td>
</tr>
<tr>
<td></td>
<td>Break teeth</td>
</tr>
<tr>
<td></td>
<td>Cut/tear facial/head tissue</td>
</tr>
<tr>
<td></td>
<td>Break nose</td>
</tr>
<tr>
<td>Life Support/Ambient Conditions (non robot specific)</td>
<td>Asphyxiate</td>
</tr>
<tr>
<td></td>
<td>Toxic substances</td>
</tr>
<tr>
<td></td>
<td>High/low ambient temperatures</td>
</tr>
<tr>
<td></td>
<td>High sonic energy</td>
</tr>
</tbody>
</table>

*Table 1 - Nature of harm (from [KAR95])*
Severity – Table Example

<table>
<thead>
<tr>
<th>AIS*</th>
<th>Severity</th>
<th>Type of Injury</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>None</td>
<td>None</td>
</tr>
<tr>
<td>1</td>
<td>Minor</td>
<td>Superficial injury</td>
</tr>
<tr>
<td>2</td>
<td>Moderate</td>
<td>Recoverable</td>
</tr>
<tr>
<td>3</td>
<td>Serious</td>
<td>Possibly recoverable</td>
</tr>
<tr>
<td>4</td>
<td>Severe</td>
<td>Not fully recoverable without care</td>
</tr>
<tr>
<td>5</td>
<td>Critical</td>
<td>Not fully recoverable with care</td>
</tr>
<tr>
<td>6</td>
<td>Fatal</td>
<td>Not survivable</td>
</tr>
</tbody>
</table>

*Table 1 - The Abbreviated Injury Scale from [AAAM 98]*
# Probability of occurrence or likelihood

<table>
<thead>
<tr>
<th>Likelihood</th>
<th>Indicative probability (per year)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Frequent</td>
<td>&gt;1</td>
</tr>
<tr>
<td>Probable</td>
<td>$1 - 10^{-1}$</td>
</tr>
<tr>
<td>Occasional</td>
<td>$10^{-1} - 10^{-2}$</td>
</tr>
<tr>
<td>Rare</td>
<td>$10^{-2} - 10^{-6}$</td>
</tr>
<tr>
<td>Almost Impossible</td>
<td>&lt;$10^{-6}$</td>
</tr>
</tbody>
</table>

*Table 1 – Indicative probability values for likelihood estimation*
Risk: combination of the probability of occurrence of harm and the severity of that harm.

Tolerable risk: risk which is accepted in a given context based on the current values of society.

<table>
<thead>
<tr>
<th>Likelihood</th>
<th>6</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Fatal</td>
<td>Critical</td>
<td>Severe</td>
<td>Serious</td>
<td>Moderate</td>
<td>Minor</td>
<td>None</td>
</tr>
<tr>
<td>Frequent</td>
<td>H</td>
<td>H</td>
<td>H</td>
<td>H</td>
<td>H</td>
<td>I</td>
<td>N</td>
</tr>
<tr>
<td>Probable</td>
<td>H</td>
<td>H</td>
<td>H</td>
<td>H</td>
<td>I</td>
<td>L</td>
<td>N</td>
</tr>
<tr>
<td>Occasional</td>
<td>H</td>
<td>H</td>
<td>H</td>
<td>I</td>
<td>L</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>Rare</td>
<td>H</td>
<td>H</td>
<td>I</td>
<td>L</td>
<td>N</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>Improbable</td>
<td>I</td>
<td>I</td>
<td>L</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>Almost Impossible</td>
<td>L</td>
<td>L</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
</tr>
</tbody>
</table>

H - High, I - Intermediate, L - Low, N - Negligible
Other Risk Estimation

\[ R = N \times C \times F \times Q \]

- \( R \): risk related to the considered hazard
- \( Q \): probability of occurrence of harm
- \( F \): frequency and duration of exposure
- \( C \): severity of possible harm that can result
- \( N \): number of exposed people
**Safety**

- **Safety**: freedom from unacceptable risk.

<table>
<thead>
<tr>
<th>Likelihood</th>
<th>Severity</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>6 Fatal</td>
</tr>
<tr>
<td>Frequent</td>
<td>H</td>
</tr>
<tr>
<td>Probable</td>
<td>H</td>
</tr>
<tr>
<td>Occasional</td>
<td>H</td>
</tr>
<tr>
<td>Rare</td>
<td>H</td>
</tr>
<tr>
<td>Improbable</td>
<td>I</td>
</tr>
<tr>
<td>Almost Impossible</td>
<td>L</td>
</tr>
</tbody>
</table>

H - High, I - Intermediate, L - Low, N - Negligible
Causes of harm: hazards

• Hazard: potential source of harm
  – Hazardous inherent characteristics (e.g., a cutting edge, a toxic substance, etc.)
  – Hazardous controllable states of the system (e.g., hazardous motion, suspended mass)
  – Failure of hardware or software components
  – Human errors
  – Unspecified external events
  – The term hazardous motion is defined in the standard [ISO 10218:2006] to be “any motion that is likely to cause personal physical injury or damage to health”
Causes of harm: hazards (cont’d)

• **Hazardous situation**: circumstance in which people, property or the environment are exposed to one or more hazards

• **Harmful event or accident**: occurrence in which a hazardous situation results in harm

• **Incident**: event that does not lead to harm, but which has the potential to create harm in other circumstances
Example of use of terminology

- **Scenario of use**: Installation robot procedure + Patient → Cutting edge
- **Hazardous situation**: Hazard
- **Harmful event**: Cut
- **Accident**: Open wound
Risk management process overview (ISO)
Risk management activities

- Risk management: coordinated activities to direct and control an organization with regard to risk
  - **Risk analysis**: systematic use of available information to identify hazards and to estimate the risk
  - **Risk Evaluation**: process of comparing the estimated risk against given risk criteria to determine the significance of the risk
  - **Risk treatment**: process of selection and implementation of measures to modify risk
    - Risk treatment measures can include reducing, avoiding, optimizing, transferring or retaining risk.
    - Risk reduction: actions taken to lessen the probability, negative consequences, or both, associated with a risk
  - (Risk communication, transfer, etc.)
Relationship between terms, based on their definitions regarding “Risk” (ISO Guide 73)

<table>
<thead>
<tr>
<th>RISK MANAGEMENT</th>
</tr>
</thead>
<tbody>
<tr>
<td>RISK ASSESSMENT</td>
</tr>
<tr>
<td>RISK ANALYSIS</td>
</tr>
<tr>
<td>SOURCE IDENTIFICATION</td>
</tr>
<tr>
<td>RISK ESTIMATION</td>
</tr>
<tr>
<td>RISK EVALUATION</td>
</tr>
<tr>
<td>RISK TREATMENT</td>
</tr>
<tr>
<td>RISK AVOIDANCE</td>
</tr>
<tr>
<td>RISK REDUCTION</td>
</tr>
<tr>
<td>RISK TRANSFER</td>
</tr>
<tr>
<td>RISK RETENTION</td>
</tr>
<tr>
<td>RISK ACCEPTANCE</td>
</tr>
<tr>
<td>RISK COMMUNICATION</td>
</tr>
</tbody>
</table>
Risk analysis techniques

- Quantitative state-based
  - Markov chain
  - Stochastic petri nets
- Tree representation based
  - Fault Tree Analysis (FTA)
  - Event Tree Analysis (ETA)
- Table based
  - Preliminary Hazard Analysis (PHA)
  - HAZard OPerability (HAZOP)
  - Failure Modes Effects and Criticality Analysis (FMECA)
The results of the PHA are usually reported by using a PHA worksheet (or, a computer program). A typical PHA worksheet is shown below. Some analyses may require other columns, but these are the most common.

Ref. | Hazard | Accidental event (what, where, when) | Probable causes | Contingencies/Preventive actions
--- | --- | --- | --- | ---

System: Operating mode:
## Failure Modes and Effects Criticality Analysis (FMECA)

| Failure mode number | Identification of Item or Function | a. Failure mode | b. Failure cause | Failure effects | a. Local or Subsystem  
| b. Next Higher Level  
| c. Real word effect (Mission) | Risk assessment | Remarks | a. Failure detection method  
| b. Compensating Features/actions  
| c. Other |

|   |   |   |   |   | O | S | R | Remarks | a. Failure detection method  
| b. Compensating Features/actions  
| c. Other |
Hazard Operability (HAZOP)

<table>
<thead>
<tr>
<th>As Well As</th>
<th>All the design intention is achieved together with additions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Part of</td>
<td>Only some of the design intention is achieved</td>
</tr>
<tr>
<td>Reverse</td>
<td>The logical opposite of the design intention is achieved</td>
</tr>
<tr>
<td>Other than</td>
<td>Complete substitution, where no part of the original intention is achieved but something quite different happens</td>
</tr>
<tr>
<td>Early</td>
<td>Something happens earlier than expected relative to clock time</td>
</tr>
<tr>
<td>Late</td>
<td>Something happens later than expected relative to clock time</td>
</tr>
<tr>
<td>Before</td>
<td>Something happens before it is expected, relating to order or sequence</td>
</tr>
<tr>
<td>After</td>
<td>After Something happens after it is expected, relating to order or sequence</td>
</tr>
</tbody>
</table>

---

- Source: IEC 61882
Fault Tree Analysis (FTA)
### Event Tree Analysis

<table>
<thead>
<tr>
<th></th>
<th>( B_1 )</th>
<th>( B_2 )</th>
<th>( B_3 )</th>
<th>( B_4 )</th>
<th>Outcome / consequence</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accidental event</td>
<td>Additional event I occurs</td>
<td>Barrier I does not function</td>
<td>Barrier II does not function</td>
<td>Additional event II occurs</td>
<td></td>
</tr>
</tbody>
</table>

- **Accidental event**
- **Additional event I occurs**
- **Barrier I does not function**
- **Barrier II does not function**
- **Additional event II occurs**

- **Outcome / consequence**
  - Outcome 1
  - Outcome 2
  - Outcome 3
  - Outcome 4
  - Outcome 5
  - Outcome 6
  - Outcome 7
  - Outcome 8
  - Outcome 9
PRELIMINARY HAZARD ANALYSIS
Preliminary Hazard Analysis

• Identify and list potential hazards
• Performed during conceptual or preliminary design
• uncomplicated and easily learned
### Preliminary Hazard List Analysis

<table>
<thead>
<tr>
<th>No.</th>
<th>System Item</th>
<th>Hazard</th>
<th>Hazard Effects</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
</tr>
</tbody>
</table>
The results of the PHA are usually reported by using a PHA worksheet (or, a computer program). A typical PHA worksheet is shown below. Some analyses may require other columns, but these are the most common.

## Advanced worksheet: include causes / actions

<table>
<thead>
<tr>
<th>Ref.</th>
<th>Hazard</th>
<th>Accidental event (what, where, when)</th>
<th>Probable causes</th>
<th>Contingencies/Preventive actions</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

System:  
Operating mode:  
Analyst:  
Date:
4. Mishap risk assessment (before and after design safety features are implemented)

5. SCFs and TLMs

6. Recommendations for eliminating or mitigating the hazards

Figure 5.3 shows the columnar format PHA worksheet recommended for SSP usage.

This particular worksheet format has proven to be useful and effective in many applications and it provides all of the information necessary from a PHA.

The following instructions describe the information required under each column entry of the PHA worksheet:

1. **System**: This entry identifies the system under analysis.

2. **Subsystem/Function**: This entry identifies the subsystem or function under analysis.

3. **Analyst**: This entry identifies the name of the PHA analyst.

4. **Date**: This entry identifies the date of the analysis.

5. **Hazard Number**: This column identifies the number assigned to the identified hazard in the PHA (e.g., PHA-1, PHA-2, etc.). This is for future reference to the particular hazard source and may be used, for example, in the hazard action record (HAR) and the hazard tracking system (HTS).

6. **Hazard**: This column identifies the specific hazard being postulated and evaluated. (Remember: Document all hazard considerations, even if they are later proven to be nonhazardous.)

7. **Causes**: This column identifies conditions, events, or faults that could cause the hazard to exist and the events that can trigger the hazardous elements to become a mishap or accident.

8. **Effects**: This column identifies the effects and consequences of the hazard, should it occur. Generally, the worst-case result is the stated effect. The effect ultimately identifies and describes the potential mishap involved.

9. **Recommended Action**: This column identifies the recommended actions to eliminate or mitigate the identified hazards.

10. **FMRI**: This column identifies the functional mishap ranking index (FMRI) for each hazard.

11. **Comments**: This column provides space for any additional comments or notes.

12. **Status**: This column indicates the status of the recommendation (e.g., implemented, under review, etc.).

---

<table>
<thead>
<tr>
<th>No.</th>
<th>Hazard</th>
<th>Causes</th>
<th>Effects</th>
<th>Mode</th>
<th>IMRI</th>
<th>Recommended Action</th>
<th>FMRI</th>
<th>Comments</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>11</td>
<td>12</td>
<td>13</td>
<td>14</td>
</tr>
</tbody>
</table>
Application of the basic PHA

a. Construct list of hardware components and system functions.
b. Evaluate conceptual system hardware; compare with hazard checklists.
c. Evaluate system operational functions; compare with hazard checklists.
d. Identify and evaluate system energy sources to be used; compare with energy hazard checklists.
e. Evaluate system software functions; compare with hazard checklists.
f. Evaluate possible failure states.
### Subset of list of significant hazards (Extracted from ISO 10218 Annex A, Table A.1 – List of significant hazards which is itself based on Annex A of ISO 14121:1999).

<table>
<thead>
<tr>
<th>No.</th>
<th>Description</th>
<th>Example(s) of related hazardous situations</th>
<th>Related danger zone</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Mechanical hazards</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.1</td>
<td>Crushing</td>
<td>Movement (normal or singularity) of any part of the robot arm or additional axes</td>
<td>Restricted space</td>
</tr>
<tr>
<td>1.2</td>
<td>Shearing</td>
<td>Movement of additional axes</td>
<td>Around accessory equipment</td>
</tr>
<tr>
<td>1.3</td>
<td>Cutting or severing</td>
<td>Movement or rotation creating scissors action</td>
<td>Restricted space</td>
</tr>
<tr>
<td>1.4</td>
<td>Entanglement</td>
<td>Rotation of wrist or additional axes</td>
<td>Restricted space</td>
</tr>
<tr>
<td>1.5</td>
<td>Drawing-in or trapping</td>
<td>Between robot arm and any fixed object</td>
<td>Around fixed objects close to restricted space</td>
</tr>
<tr>
<td>1.6</td>
<td>Impact</td>
<td>Movement (normal or singularity) of any part of the robot arm</td>
<td>Restricted space</td>
</tr>
<tr>
<td>2</td>
<td>Electrical hazards</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.1</td>
<td>Contact of persons with live parts (direct contact)</td>
<td>Contact with live parts or connections</td>
<td>Electrical cabinet, terminal boxes, control panels at machine</td>
</tr>
<tr>
<td></td>
<td>Hazards generated by neglecting ergonomic principles in the design process</td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>-------------------------------------------------------------------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8.1</td>
<td>Unhealthy postures or excessive effort (repetitive strain)</td>
<td>Poorly designed teach pendant</td>
<td>Teach pendant</td>
</tr>
<tr>
<td>8.2</td>
<td>Inadequate consideration of hand-arm or foot-leg anatomy</td>
<td>Inappropriate location of controls</td>
<td>At load/unload work piece and tool mounting or setting positions</td>
</tr>
<tr>
<td>8.7</td>
<td>Inadequate design, location or identification of manual controls</td>
<td>Inadvertent operation of controls</td>
<td>At or near robot cell</td>
</tr>
<tr>
<td>8.8</td>
<td>Inadequate design or location of visual display units</td>
<td>Misinterpretation of displayed information</td>
<td>At or near robot cell</td>
</tr>
<tr>
<td>10</td>
<td>Unexpected start-up, unexpected overrun/over speed</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10.1</td>
<td>Failure/disorder of the energy source</td>
<td>Mechanical hazards associated with robot additional axes</td>
<td>At or near robot cell</td>
</tr>
<tr>
<td>10.2</td>
<td>Restoration of energy supply after an interruption</td>
<td>Unexpected movements of robot or additional axes</td>
<td>At or near robot cell</td>
</tr>
<tr>
<td>10.3</td>
<td>External influences on the electrical equipment</td>
<td>Unpredictable behavior of electronic controls due to electromagnetic interference</td>
<td>At or near robot cell</td>
</tr>
<tr>
<td>13</td>
<td>Failure of the power supply (external power sources)</td>
<td>Malfunctions of the control with consequent release of robot arm brake; release of brake causes robot elements to move under residual forces (inertia, gravity, spring/energy storage means) unexpectedly</td>
<td>At or near robot cell where robot elements retained in a safe condition by the application of power or fluid pressure</td>
</tr>
<tr>
<td>14</td>
<td>Failure of the control circuit (hardware or software)</td>
<td>Unexpected movements of robot or additional axes</td>
<td>At or near robot cell</td>
</tr>
<tr>
<td>18</td>
<td>Loss of stability, overturning of machinery</td>
<td>Unrestrained robot or additional axes (maintained in position by gravity), falls or overturns</td>
<td>At or near robot cell</td>
</tr>
</tbody>
</table>
Application

- Apply basic PHA to a case study
- Discuss benefits and limits
HAZOP-UML OVERVIEW
HAZard OPerability (HAZOP)

• Main principle
  – System parameter (ex. temperature)
  – Guide-word from a generic list
    (ex: more, less, etc…)
  – Deviation identification
  – Consequences analysis and recommendations

• Exemple
  – Temperature x More = temperature too high

How to identify these parameters and associated deviations?
HAZOP-UML

UML diagrams

Use case diagrams

HAZOP guidewords

Deviation analysis

Guide word | Meaning
---|---
NO OR NOT | Complete negation of the design intent
MORE | Quantitative increase
LESS | Quantitative decrease
AS WELL AS | Qualitative modification/increase
PART OF | Qualitative modification/decrease
REVERSE | Logical opposite of the design intent
OTHER THAN | Complete substitution

Table HAZOP-UML

Hazards | Recommendations
**Deviation analysis**

**Guidewords for UML models**

**Entity = Use Case**

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Guideword</th>
<th>Interpretation</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>No/none</td>
<td>The condition is not evaluated and can have any value</td>
</tr>
<tr>
<td></td>
<td>Other than</td>
<td>The condition is evaluated true whereas it is false, or vice versa</td>
</tr>
<tr>
<td></td>
<td>As well as</td>
<td>The condition is correctly evaluated but other unexpected conditions are true</td>
</tr>
<tr>
<td></td>
<td>Part of</td>
<td>The condition is partially evaluated</td>
</tr>
<tr>
<td></td>
<td>Early</td>
<td>The condition is evaluated earlier than required for correct synchronization with the environment</td>
</tr>
<tr>
<td></td>
<td>Late</td>
<td>The condition is evaluated later than required for correct synchronization with the environment</td>
</tr>
</tbody>
</table>

**HAZOP Table**

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Attribute</th>
<th>Deviation</th>
<th>Use Case Effect</th>
<th>Real World Effect</th>
<th>Severity</th>
<th>Possible Causes</th>
<th>Safety Recommendation</th>
<th>Remarks</th>
<th>Hazard Num.</th>
</tr>
</thead>
<tbody>
<tr>
<td>UC02.15</td>
<td>Battery charge is sufficient to do this task and to help the patient to sit down (precond)</td>
<td>No/none</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>HN6</td>
</tr>
<tr>
<td></td>
<td>Battery charge is too low but the robot starts the standing up operation</td>
<td>The robot interrupts its movement (standing up or walking)</td>
<td></td>
<td></td>
<td>Serious</td>
<td>HW/SS Failure Specification error</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Loss of balance or fall of the patient</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Robot starts standing operation, the most probable scenario is that the patient will fall back on the seat.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Worst-case electrical consumption must be evaluated beforehand. Take the lower bound of the battery charge estimation</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>UC02.16</td>
<td>Other than Battery charge is high enough but the robot thinks Robot refuses to start stand up operation</td>
<td>Patient is confused</td>
<td></td>
<td></td>
<td>None</td>
<td>HW/SS Failure Specification error</td>
<td></td>
<td></td>
<td>None</td>
</tr>
</tbody>
</table>

**Project:** MIRAS

**HAZOP table number:** UC02

**Entity:** UC02. Standing up operation

**Prepared by:** DMG

**Revised by:** JG

**Date:** 04/08/2009

<table>
<thead>
<tr>
<th>Precond</th>
<th>Postcondition</th>
<th>Invariant</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Figure 4:** is analyzed using the guide words No and Other than. It leads to the following scenario: the patient tries to standup while the robot is not otherwise.

The robot starts the standing up operation, the most probable scenario is that the patient will fall back on the seat.
Previous applications

ANR-MIRAS (2009-2013) Multimodal Interactive Robot of Assistance in Strolling


HAZOP–UML Complexity measures

<table>
<thead>
<tr>
<th></th>
<th>PHRIENDS</th>
<th>MIRAS</th>
<th>SAPHARI</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Use cases</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Conditions</td>
<td>39</td>
<td>45</td>
<td>54</td>
</tr>
<tr>
<td>Analyzed deviations</td>
<td>297</td>
<td>317</td>
<td>324</td>
</tr>
<tr>
<td>Interpreted deviations</td>
<td>179</td>
<td>134</td>
<td>65</td>
</tr>
<tr>
<td>Interpreted deviations with recommendation</td>
<td>120</td>
<td>72</td>
<td>50</td>
</tr>
<tr>
<td><strong>Sequence diagrams</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Messages</td>
<td>91</td>
<td>52</td>
<td>122</td>
</tr>
<tr>
<td>Analyzed deviations</td>
<td>1397</td>
<td>676</td>
<td>2196</td>
</tr>
<tr>
<td>Interpreted deviations</td>
<td>589</td>
<td>163</td>
<td>87</td>
</tr>
<tr>
<td>Interpreted deviations with recommendation</td>
<td>274</td>
<td>85</td>
<td>36</td>
</tr>
<tr>
<td>Number of hazards</td>
<td>21</td>
<td>16</td>
<td>28</td>
</tr>
</tbody>
</table>
**HAZOP-UML -ity**

- **Applicability:** model based systematic analysis / few ressources / controlled complexity

- **Validity:** has been compared to classic Preliminary Hazard Analysis

- **Usability:** Simple / first step of the dev. Process / share models with system developers

- **Transferability:** no specific tool / already transferred to several indus & research partners
UNIFIED MODELING LANGUAGE
Why use a model?

- A model is used when:
  - Reality is too complex (simplification)
  - A concept is required (abstraction)
  - Direct modification of the design is too hazardous (representation)
  - Communicate between developers
  - Prevent and eliminate errors of specification/design
  - Guarantee tracability from requirements to implementation
  - Perform non functional analysis (performance, verification, etc.)
Programming Complex Motions by Demonstration

To program new motions, the robotic system is equipped with two programming environments. In keyboard mode, the robot is programmed in the traditional way, available in almost all robots. In this mode, with the keyboard point-to-point positions on a trajectory are generated that are traced and stored in a database. In programming by demonstration mode (RPD), the programmer demonstrates the task to be executed with his own hand. The motions are measured, recorded, and processed so that the robot can reproduce them. Many approaches described in the literature [3-5] share a common feature: they are designed mainly for simple pick-and-place applications like those found in industry, such as loading pallets and sorting and feeding parts. Neither the demonstrated motion trajectory nor the dynamics of the motion, such as the speed or general time response, are considered. But in the field of rehabilitation robots like FRIEND, where the tasks are much more complicated, this information is of great importance.

Fig. 2. Architecture of the system FRIEND.
Functional decomposition

- Traditional approach
- Each module is a step of the global process
- Functional division from specification to subprograms
Functional decomposition

- Main function
  - Subfunction 1
    - Subfunction 1.1
    - Subfunction 1.2
  - Subfunction 2
    - Subfunction 2.1
    - Subfunction 2.2
Object decomposition

• More recent approach (computer systems)
• Each module is an object of the application
• Objects are autonomous entities that collaborate to reach a goal
• Function is carried with collaborative objects

1. go to ground floor

2: blink

3: open

Door

Lift

Light

Button
Functional approach

• More intuitive
• Focus on “DO”
• Suits when all is known in advance
• BUT
  – Stiff Architecture
  – Evolvability is limited
  – Not suitable to discovery
Object approach

- Focus on “BE”
- Simple (small number of concepts)
- Reasonning on abstraction (object of the domain)
- Suitable for discovery and evolvability
- BUT
  - Hard to understand for people used to functional approach.
Object Oriented advantages

• Lead to more stable model
  – Based on real world
• Independancy from functions
  – Evolvability
• Encapsulate complexity
  – Suitable for reuse
What do we need?

- A modelling language
  - Clear notation
  - Usability
    - Not too complex
    - Exchange data between developers, and stakeholders
  - Completeness and consistency semantics
- A development process

Method = Language + Process
The unified notation UML

- Comes from BOOCH, OMT and OOSE
- And take good ideas from other methods
- Convergence of notations
- A unique example of standard notation which is a de facto standard (in computer science)
UML development

- UML 1.0
- UML 1.5
- UML 2.0 (free on www.omg.org)

- Standardization by OMG September 1997
- 1995
- 1996
- September 2001
- August 2005

- Méthode unifiée 0.8
- Booch'93
- OMT-2
- OOSE
- Partenaires
- Autres méthodes
- Booch'91
- OMT-1
Summary

• UML is a notation not a method
• UML is an object modelling language
• UML is suitable for all object development
• UML is free

UML is a de facto standard for the notation of object oriented development
UML diagrams
Two types: structural and dynamic

- Structural representation for an element
  - Internal structure (composition) and external (relationships and dependencies with other elements)

- Dynamic representation
  - Behavior considering time: interaction with other elements, modification of its internal state…
• **Represents objects and their relationships**

Object diagram

```
<table>
<thead>
<tr>
<th>awindow : Window</th>
</tr>
</thead>
<tbody>
<tr>
<td>horizontalBar:ScrollBar</td>
</tr>
<tr>
<td>verticalBar:ScrollBar</td>
</tr>
<tr>
<td>surface:Pane</td>
</tr>
<tr>
<td>title:TitleBar</td>
</tr>
</tbody>
</table>
```

```
moves
```

```
moves
```

```
moves
```
Class diagram

- Represents static structure with classes and their relationships
• Represents **physical components of a system**
Deployement diagram

• Represents the deployement of the components on hardware devices
Use case diagram

- Represents objectives of the use of the system according to actors' viewpoint
Sequence diagram

- Represents interactions between objects according to time.
• Equivalent to sequence diagram but with a *spacial* representation
Timing Diagram

- **Lifelines**
- **State or condition**
- **Duration Constraints**

- **sd UserAccepted**
  - WaitAccess
  - Idle

- **User**

- **ACSystem**
  - NoCard
  - HasCard

- **Time Observation**
- **Time Constraint**
- **Message**

- **t=now**
- **d**

- **Unlock**
• Represents life cycle of an object
Activity diagram

• Represents an activity flow in an operation, a use case or a business process
Classes and objects
The objects

- Real world objects born, live and dead
- Computer system objects are a simple representation of real world elements
- Objects represent concrete entities (a sensor, an actuator) or abstract (PID regulator, Neural….)
Graphical notation of object

One object

Another object

And another one
Objects are abstractions

- An abstraction is a summary of essential characteristics
- Hide the details
- An abstraction depends on a viewpoint (e.g. mathematical, automatic, architectural)
Abstraction examples

- A television
- A complex number
- A financial operation
- A logical gate
- A battery
- An actuator
- A sensor
- A PID regulator
- A joint
Object chaos

- Many objects
- Humans are always classing/categorizing in order to understand: animals, plants, mushrooms, atoms...
Object chaos cont’d
Classes

• A class is an abstraction of several objects
• Can be interpreted as a factorization
3.40 Composite Object

A composite object represents a high-level object made of tightly-bound parts. This is an instance of a composite class, which implies the composition aggregation between the class and its parts. A composite object is similar to (but simpler and more restricted than) a collaboration; however, it is defined completely by composition in a static model. See Section 3.48, "Composition," on page 3-81.

3.40.2 Notation

A composite object is shown as a larger object symbol. The names of the composite object are placed in a compartment near the top of the rectangle (as with any object). The lower compartment holds the parts of the composite object instead of a list of attribute values. (However, even a list of attribute values may be regarded as the parts of a composite object, so there is not a great difference.) It is possible for some of the parts to be composite objects with further nesting.

Figure 3-39 Composite Objects
### 3.5 Example

Figure 3-47

**Styles of Displaying Generalizations**

- **Shape**
- **Spline**
- **Ellipse**
- **Polygon**

**Shared Target Style**

**Separate Target Style**

### 3.4 Style Guidelines

If there are multiple adornments on a single association end, they are presented in the following order, reading from the end of the path attached to the classifier toward the bulk of the path:

- qualifier
- aggregation symbol
- navigation arrow

Rolenames and multiplicity should be placed near the end of the path so that they are not confused with a different association. They may be placed on either side of the line. It is tempting to specify that they will always be placed on a given side of the line (clockwise or counterclockwise), but this is sometimes overridden by the need for clarity in a crowded layout. A rolename and a multiplicity may be placed on opposite sides of the same association end, or they may be placed together (for example, "*employee").

### 3.4.6 Mapping

The adornments on the end of an association path map into properties of the corresponding role of the Association. In general, implications cannot be drawn from the absence of an adornment (it may simply be suppressed) but see the preceding descriptions for details. The interface specifier maps into the "specification" rolename in the AssociationEnd-Classifier association.
Object dynamics
Communication between objects

- System = society of collaborative objects
- Object work together to perform the service
- The behavior of a system depends on how the objects collaborate
A message

• Is the communication unit between objects
• Very general concept with various application
• Can represents both control and data flow and also events, or activities
• **A** send a message X to object B, the object B sent Y to C, then etc…
Sequence diagram

- Lifelines are objects
- The tag is `objectName:ClassName`

  is for a message
  is a return

```
objet1:Sender
message1(arg)
message2(arg)
message de retour
message4()
```

```
objet2: Receiver
```

Time
Collaboration and sequence diagram

Diagram with nodes A, B, and C, showing interactions and sequence flows as follows:

- A to B: M1, M4, M10
- B to C: M2, M6, M7
- C to A: M8

Other interactions:
- M3
- M5
- M9
- M7
Exercise : Simple Watch

From class diagram :

1. Perform a sequence diagram of the following scenario: a user wants to set the minutes

   *Pushing twice the button 1, he can set the minutes (hours blinks and then minutes). Then with the button 2 (with releasing it), minutes are incremented. Once minutes are set, the user push the button 1 and the minutes stop blinking.*
Simple watch: Sequence diagram
Simple Watch: Communication diagram
Use cases
Use cases

- Represent functional requirements
Why use case diagrams?

• A graphical modelling of requirements
• Used by final users to express/discuss about their requirements
• Are usefull to communicate at the first steps of the development
• Are a basis for functional testing and other activities
Project main thread

- User
  - express
- Analyst
  - understand
- Developer
  - implement
- Architect
  - realize
- Testing
  - check

Use cases
A**ctors**

- Represent roles that humans, hardware devices, or external systems play while interacting with the given system.
- They are not part of the system and are situated outside of the system boundary.
- Actors may be both at input and output ends of a use case.
Identify actors

- Define system boundary to identify actors correctly
- Identify users and systems that depend on the system’s primary and secondary functionalities
- Identify hardware and software platforms with which the system interacts
- Select entities that play distinctly different roles in the system
- Identify as actors external entities with common goals and direct interaction with the system
- Denote actors as nouns
Identifying Use Cases

• Interactions between actors and the system
• Objectives of the actors
• Services delivered by the system
• A use case encompasses several scenarios (nominal, exceptions, alternatives)

To name the use cases, give it a verb name to show the action that must be performed
Scenarios

• Specify behaviour of use case by description, not modeling
  – Examples include informal structured text, formal structured text with conditions, and pseudocode

• Typically specify:
  – How and when the use case starts and ends
  – Interaction with the actors and the exchange of objects
  – Flow of events: main / typical (success), alternative (success), and exceptional (failure) flows
## Example of UC textual description

### Use-Case:
- **Use-case name**

### Brief Description
- **Brief description of use-case**

### Actor Brief Descriptions
- **Actor 1 Name**

### Preconditions
- **Pre-condition 1**

### Basic Flow of Events
- The use case begins when **actor**, **does something**...
- **Basic flow step 1**
  - ...
  - **Basic flow step n**
  - The use case ends.

### Alternative Flows
- **Alternate flow 1**
- If in step **x** of the basic flow the **actor or system does something**, then
- **Describe flow**
- The use case resumes at step **y**

### Subflows
- **Subflow 1, step 1**
  - ...
  - **Subflow 1, step n**

### Post-Conditions
- **Post-condition 1**

### Special Requirements
- **Special requirement 1**
HAZOP-UML IN ACTION
HAZOP-UML

UML diagrams

HAZOP guidewords

Deviation analysis

<table>
<thead>
<tr>
<th>Guide word</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>NO OR NOT</td>
<td>Complete negation of the design intent</td>
</tr>
<tr>
<td>MORE</td>
<td>Quantitative increase</td>
</tr>
<tr>
<td>LESS</td>
<td>Quantitative decrease</td>
</tr>
<tr>
<td>AS WELL AS</td>
<td>Qualitative modification/increase</td>
</tr>
<tr>
<td>PART OF</td>
<td>Qualitative modification/decrease</td>
</tr>
<tr>
<td>REVERSE</td>
<td>Logical opposite of the design intent</td>
</tr>
<tr>
<td>OTHER THAN</td>
<td>Complete substitution</td>
</tr>
</tbody>
</table>

HAZOP-UML tables

Hazards

Recommendations
HAZOP-UML Process

1. UML models
   – Actors & use cases identification, pre/post conditions and invariant identification for each use case
   – Sequence diagram (nominal & alternative) for each use case

2. Deviation analysis for each element and for each attribute of UML diagrams (UC conditions & messages)
and not the internal objects of the system.

Sequence diagrams are used to represent alternative ways of achieving objectives (e.g., actions, objectives, etc.) but to rather use several diagrams to represent alternative ways of achieving objectives. As the time increases from top to bottom, each message exchanged between actors and the system are represented along the robot and the user.

Fig. 2. Exemple: rehabilitation robot

In UML, a sequence diagram is a representation of an interaction. In UML, a sequence diagram is a representation of an interaction. In our case three types of messages are used: the robot tasks and interactions.

Fig. 3.

Fig. 4.

Fig. 5

Fig. 6

Fig. 7

2. Background

For its execution (e.g., opening a door), the messages are all physical contacts, whereas FTA and FMECA are more dedicated to advanced steps, like robot tasks and interactions.

Fig. 1

They are also widely used in reactive systems as robot controllers. Hence, at the beginning of any project development. State machine diagrams provide background on UML and HAZOP. In Section 4. focuses on operational hazards, i.e., hazards linked with the software interfaces), indirect interaction (software interfaces), and exceptions (activities that must take place, or any conditions that must be accomplished tasks).

Table is given in DefStan00-58 (2000) and IEC61882 (2001) – -

 deren- 

UC01 Strolling

UC02 Standing up operation

UC03 Sitting down operation

UC08 Alarm Handling

UC10 Patient profile learning

Actor

Association

Use case

Studied system boundary

Patient

Medical Staff

MIRAS Robot

UC01 Strolling

UC02 Standing up operation

UC03 Sitting down operation

UC08 Alarm Handling

UC10 Patient profile learning

Exemple: rehabilitation robot
Use case specification

<table>
<thead>
<tr>
<th>Use case name</th>
<th>UC02. Standing up operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Abstract</td>
<td>The patient stands up with the help of the robot</td>
</tr>
<tr>
<td>Precondition</td>
<td>The patient is sitting down</td>
</tr>
<tr>
<td></td>
<td>The robot is waiting for the standing up operation</td>
</tr>
<tr>
<td></td>
<td>Battery charge is sufficient to do this task and to help the patient to sit down</td>
</tr>
<tr>
<td></td>
<td>The robot is in front of the patient</td>
</tr>
<tr>
<td>Postcondition</td>
<td>The patient is standing up</td>
</tr>
<tr>
<td></td>
<td>The robot is in admittance mode</td>
</tr>
<tr>
<td>Invariant</td>
<td>The patient holds both handles of the robot</td>
</tr>
<tr>
<td></td>
<td>The robot is in standing up mode</td>
</tr>
<tr>
<td></td>
<td>Physiological parameters are acceptable</td>
</tr>
</tbody>
</table>
Sequence diagram specification
For each deviation, the procedure is then to investigate causes, through dynamic models of the system. We propose to use UML to partition and describe the interactions between team members. The choice of the considered domain, so adaptation depends on the expertise of the initiators of the method. Additionally, the HAZOP method needs the allocation of human resources and suffers from the combinatorial explosion when too many deviations are considered. Another issue is actually capable of finding all the hazards, i.e., hazards linked to the operational hazards, i.e., hazards linked to the human–robot interactions. The whole process is then introduced in Sections 3.1, and 3.2, and 3.3.

The study relies on HAZOP–UML method proposed in this paper, presents a prototype of a tool for HAZOP–UML. The HAZOP–UML approach, it is important to note that HAZOP–UML does not identify all hazards. First because no single hazard identification technique is actually capable of finding all the hazards. Second, the success of a HAZOP study depends greatly on the ability of the analyst and the analysts' expertise. Even if our objective is to propose a systematic approach, it is important to note that HAZOP–UML does not identify all hazards. First because no single hazard identification technique is actually capable of finding all the hazards. Second, the success of a HAZOP study depends greatly on the ability of the analyst and the analysts' expertise.
• Use case textual specification

<table>
<thead>
<tr>
<th>Use case name</th>
<th>UC02. Standing up operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Abstract</td>
<td>The patient stands up with the help of the robot</td>
</tr>
</tbody>
</table>
| Precondition  | The patient is sitting down
The robot is waiting for the standing up operation
Battery charge is sufficient to do this task and to help the patient to sit down
The robot is in front of the patient |
| Postcondition | The patient is standing up
The robot is in admittance mode |
| Invariant     | The patient holds both handles of the robot
The robot is in standing up mode
Physiological parameters are acceptable |

• For each use case one or more sequence diagram

UML final models
In Figure 12 given example, a precondition of UC02 (previously presented in Figure 4) is analyzed using the guide words No and Other than. It leads to a concatenated list of identified hazards. The resulting combination of the precondition and to help the patient to sit down (precond) and has been formulated in the HAZOP table UC02 line 15 (UC02.15).

As was identified the hazard HN6 (Fall of the patient due to imbalance caused by the robot).

As well as the condition is partially evaluated/Some conditions are missing, recommendation Rec2 from Figure 14, covers hazards HN6 (fall of the patient due to imbalance caused by the robot).

Hazard Numbers: real world examples are identified as hazards and as-

- Battery charge is too low but the robot starts the standing operation
- Battery charge is high enough but the robot thinks it is low
- Robot refuses to start standing operation
- Patient is confused

The condition is evaluated true whereas it is false, or vice versa

The condition is evaluated later than required for correct synchronization with the environment

- The condition is not evaluated and can have any value
- The condition is correctly evaluated but other unexpected conditions are true
- The condition is partially evaluated
- The condition is correctly evaluated but other unexpected conditions are true

The resulting documents are the tables as the raw artefacts, but also: a concatenated list of identified hazards validated by domain experts to confirm the study and the HAZOP-UML tables.

**Guidewords for UML models**

<table>
<thead>
<tr>
<th>Entity = Use Case</th>
</tr>
</thead>
<tbody>
<tr>
<td>Attribute</td>
</tr>
<tr>
<td>------------</td>
</tr>
<tr>
<td>No/none</td>
</tr>
<tr>
<td>Other than</td>
</tr>
<tr>
<td>As well as</td>
</tr>
<tr>
<td>Part of</td>
</tr>
<tr>
<td>Some conditions are missing</td>
</tr>
<tr>
<td>Early</td>
</tr>
<tr>
<td>Late</td>
</tr>
</tbody>
</table>

**HAZOP Table**

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Attribute</th>
<th>Guide word</th>
<th>Deviation</th>
<th>Use Case Effect</th>
<th>Real World Effect</th>
<th>Severity</th>
<th>Possible Causes</th>
<th>Safety Recommendation</th>
<th>Remarks</th>
<th>Hazard Num.</th>
</tr>
</thead>
<tbody>
<tr>
<td>UC02.15</td>
<td>Battery charge is sufficient to do this task and to help the patient to sit down (precond)</td>
<td>No/none</td>
<td>Battery charge is too low but the robot starts the standing up operation</td>
<td>The robot interrupts its movement (standing up or walking)</td>
<td>Loss of balance or fall of the patient</td>
<td>Serious</td>
<td>HW/SW Failure Specification error</td>
<td>Worst-case electrical consumption must be evaluated beforehand. Take the lower bound of the battery charge estimation</td>
<td>If the robot stops during standing operation, the most probable scenario is that the patient will fall back on the seat.</td>
<td>HN6</td>
</tr>
<tr>
<td>UC02.16</td>
<td>Other than</td>
<td>Battery charge is high enough but the robot thinks it is low</td>
<td>Robot refuses to start stand up operation</td>
<td>Patient is confused</td>
<td>None</td>
<td>HW/SW Failure Specification error</td>
<td>None</td>
<td>None</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
# HAZOP-UML outputs

## Hazard List

<table>
<thead>
<tr>
<th>Num.</th>
<th>Hazard</th>
<th>Severity</th>
<th>References</th>
</tr>
</thead>
<tbody>
<tr>
<td>HN4</td>
<td>Fall of the patient without alarm or with a late alarm</td>
<td>Severe</td>
<td>UC13.SD01.29</td>
</tr>
<tr>
<td>HN5</td>
<td>Physiological problem of the patient without alarm or with a late alarm</td>
<td>Severe</td>
<td>UC03.SD02.57</td>
</tr>
<tr>
<td>HN6</td>
<td>Fall of the patient due to imbalance caused by the robot</td>
<td>Severe</td>
<td>UC12.SD01.19,30</td>
</tr>
<tr>
<td>HN7</td>
<td>Failure to switch to safe mode when a problem is detected. The robot keeps moving</td>
<td>Severe</td>
<td>UC12.SD01.62,89</td>
</tr>
<tr>
<td>HN1</td>
<td>Incorrect position of the patient during robot use</td>
<td>Serious</td>
<td>UC13.SD01.1,2,3</td>
</tr>
</tbody>
</table>

## Safety Recommendation

<table>
<thead>
<tr>
<th>Num.</th>
<th>Safety recommendation</th>
<th>Hazard Num.</th>
<th>References</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rec1</td>
<td>The standing-up profile should be validated by a human operator</td>
<td>HN8, HN12</td>
<td>UC03.SD02.91,96</td>
</tr>
<tr>
<td>Rec2</td>
<td>Worst-case electrical consumption must be evaluated beforehand (and display of the mean battery time left by the robot)</td>
<td>HN6</td>
<td>UC02.15</td>
</tr>
<tr>
<td>Rec22</td>
<td>Send regularly a network heartbeat from the robot to the medical staff control panel. Launch alarm on time-out.</td>
<td>HN6</td>
<td>UC01.SD1.15,24</td>
</tr>
<tr>
<td>Rec31</td>
<td>Safety margins should determined for maximum and minimum height of the robot (monitoring is required)</td>
<td>HN8</td>
<td>UC03.SD02.91</td>
</tr>
</tbody>
</table>
HAZOP-UML outputs, tracability

- HAZOP-UML Tables
- Hazards list
- Recommandations list
Application

• Case study: [change name here]
  – Textual specification and/or vidéo
  – UML diagrams
  – HAZOP-UML analysis
  – Conclusions