

Article

Not peer-reviewed version

---

# RTMN 2.0 – an Extension of Robot Task Modelling and Notation (RTMN) Focus on Human Robot Collaboration

---

[Congyu Zhang Sprenger](#) <sup>\*</sup>, [Juan Antonio Corrales Ramón](#) <sup>\*</sup>, Norman Urs Baier

Posted Date: 23 November 2023

doi: 10.20944/preprints202311.1427.v1

Keywords: RTMN; HRC; HRC Mode; HRC Task; KPI; Requirement; Decision making



Preprints.org is a free multidiscipline platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This is an open access article distributed under the Creative Commons Attribution License which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

*Article*

# RTMN 2.0—An Extension of Robot Task Modelling and Notation (RTMN) Focus on Human Robot Collaboration

Congyu Zhang Sprenger <sup>1,\*</sup>, Juan Antonio Corrales Ramón <sup>2,\*</sup> and Norman Urs Baier <sup>3</sup>.

<sup>1</sup> Bern University of Applied Science, School of Engineering and Computer Science, Institute of I3S, Burgdorf, 3400, Switzerland; congyu.thangsprenger@bfh.ch

<sup>2</sup> Universidade de Santiago de Compostela, Centro Singular de Investigación en Tecnoloxías Intelixentes (CiTIUS), Santiago de Compostela, 15782, Spain; juanantonio.corrales@usc.es

<sup>3</sup> Bern University of Applied Science, School of Engineering and Computer Science, Institute of I3S, Burgdorf, 3400, Switzerland; norman.baier@bfh.ch

\* Correspondence: congyu.thangsprenger@bfh.ch (C.Z.S.); juanantonio.corrales@usc.es (J.A.C.R.)

**Abstract:** This paper describes RTMN 2.0 - an extension of the modeling language RTMN. RTMN combines process modeling and robot execution. The reusability and intuitive robot programming allows people without programming expertise to plan and control robots. While RTMN focuses on the ease of use and flexibility of robot programming by providing an intuitive modelling language, RTMN 2.0 concentrates on Human Robot Collaboration (HRC), which represents the current trend of the industry shift from 'mass-production' to 'mass-customization'. While traditional robot systems provide high payload capabilities and repeatability, they suffer from limited flexibility and dexterity. Therefore, human-robot collaboration is a suitable arrangement to leverage the unique capabilities of both humans and robots for increased efficiency and quality in industrial scenarios. HRC has made a great impact on the robotic industry: increased efficiency, reduced costs and improved productivity, which can be adopted to make up for the skill gap of a shortage of workers in the manufacturing industry. The extension in RTMN 2.0 includes the following notations: HRC tasks, requirement, Key Performance Indicator (KPI), condition check and decision making, join/split and data association. With all these additional elements RTMN 2.0 covers the full scale of agile manufacturing – light out and collaborative.

**Keywords:** RTMN; HRC; HRC mode; HRC task; KPI; requirement; decision making

## 1. Introduction

Flexible robot programming usually is costly and requires a lot of expertise. The development cycle often last months and years [1]. This puts a lot of pressure on operators and engineers. On the one hand they are required to control the robot, and on the other hand it is hard to understand the robot tasks due to a lack of high-level robotics expertise. Although frameworks and tools are available to simplify robot programming, these tools are often not flexible (brand specific), too technical and not intuitive enough for operators and engineers [1]

In our former research we have developed the modelling language called RTMN (Robot Task Modeling and Notation) [2] - an ontology-enabled skill-based robot task modeling and notation. It aims to ease the robotic process modeling. The basic elements are based on Business Process Modeling and Notation (BPMN) [3]. The main motivation for developing RTMN was the challenges that non-robotic experts are facing in the fast-growing agile production industry: costly and high requirements for robot programming expertise as well as the complexity and difficulties of controlling robotic systems. In order to bridge this gap, the authors introduced a model-driven framework that allows intuitive modeling and programming of robotic processes using RTMN - an ontology-enabled skill-based robot task model and notation - that enables non-experts to plan and program robot tasks. In this research work the authors conducted literature analysis, adopted both

quantitative (questionnaires) and qualitative (interviews) research methods which balance the limitations of one type of method by the strengths of another and enable more comprehensive understanding of the research area. The validation results indicate that users find RTMN notations simple to understand and intuitive to use.

This paper covers the second generation of the RTMN. We call this new version RTMN 2.0. RTMN 2.0 adds many additional notations: four Human Robot Collaboration (HRC) tasks, requirement and KPIs, condition check and decision making, join/split and data association. HRC is the main focus among these notations.

This research is conducted under the research project ACROBA. ACROBA is an EU H2020 project. The consortium consists of 17 partners from 9 countries. "The ACROBA project aims to develop and demonstrate a novel concept of cognitive robotic platforms based on a modular approach able to be smoothly adapted to virtually any industrial scenario applying agile manufacturing principles" [1].

### *1.1. What is Human-Robot Collaboration (HRC)?*

The ever-growing advancements in the manufacturing industry have made it necessary for companies to optimize their systems by decreasing human workload, fatigue risk, and overall costs. This necessity has led to the introduction of human-robot collaboration in industrial environments [4]. Industry experts have established that the complete removal of humans from manufacturing systems is not viable. A more realistic goal is to have humans and intelligent machines working in harmony [5]. From an anthropological standpoint, "collaboration" refers to multiple parties communicating with each other and coordinating their actions in order to achieve a common goal. To this aim, collaborators observe each other, infer the intent behind an action, and plan their own actions in accordance with this intent. Similarly, in human-robot collaboration, robot systems should be designed with the appropriate tools to coordinate their actions with humans and employ the relevant cognitive and communicative mechanisms such that they can plan actions towards an established goal [6].

For the successful implementation of human-robot collaboration, the machines require advanced cognitive capabilities, to allow the human operators to collaborate comfortably and efficiently and maintain high confidence in these systems. If implemented correctly, industries may achieve a reasonable task reduction for human operators. To this end, robots should be equipped with understanding capabilities which allow them to operate with a human just as two humans would when working together [7]. HRC systems do not rely on an equal divide of workload between humans and robots. The levels of robot automation are based on the application and are decided such that they lead to an overall improvement in the system performance [8]. This improvement in performance can be attributed to the complementary strengths of either party. Robots offer efficient and guaranteed performance at high speeds, whilst humans offer understanding, reasoning and problem solving [4].

### *1.2. The Importance of Human Robot Collaboration (HRC)*

The aim of industrial robotics is to enable efficient performance, repeatedly and accurately [9]. Nowadays, assembly lines have important requirements related to adaptability, mainly due to the rapid rate at which new products are introduced to the market, as well as changing technologies. The current trends in industry reflect a shift from 'mass-production' to 'mass-customization'. Products now come with numerous variants or upgrades, and a much shorter product lifetime. This imposes a challenge of flexibility and adaptability on the manufacturing process. A challenge for which human-robot collaboration is an attractive arrangement [10]. Companies have traditionally relied on robots with in-built capabilities and limited flexibility. However, this level of flexibility is not enough to match the current market demands [11]. While traditional robot systems provide high payload capabilities and repeatability, they then suffer from limited flexibility and dexterity [12]. Thus, human-robot collaboration is a suitable arrangement to leverage the unique capabilities of both humans and robots for increased efficiency and quality in industrial scenarios. The recent trend of

automation and data exchange, known as Industry 4.0, also supports the use of collaborative systems in industry. The aim of Industry 4.0 is to achieve efficiency, cost-reduction, and increased productivity, by means of integrated automation. This aim highlights the need for flexible and interoperable systems, including intelligent decision-making software, and robots which can be quickly, safely and intuitively operated by humans [13].

Industries are increasingly relying on HRC arrangements, both from an engineering perspective, as well as a socio-economic standpoint. While the manufacturing industry is a significant source of employment, it has been reported that most jobs offered by this sector may remain unfilled. This is attributed to a shortage of workers with the relevant technological and technical skills [14]. HRC is therefore a promising alternative, which makes up for the skill gap, still requires human operators, and may be more attractive to the younger generation. Additionally, robotic systems result in higher competitiveness with countries with cheap labor systems and increase trust in the company's technological aptitude. Collaborative systems also alleviate the ergonomic burden on human workers, resulting in an improved work environment and reduction of occupational injuries. This makes environments which include both robots and human laborers more attractive to interested partners, customers and the public [15]. Modern technologies of intuitive systems such as augmented reality, walkthrough programming, and programming by demonstration, are all simple methods to operate collaborative robots, unlike the advanced technical expertise necessary to operate traditional robotic systems [13].

Nowadays, collaborative robotic solutions are attractive even to small and medium-sized companies, since such systems are more affordable, compact and easy-to-use compared with traditional robotic systems. Traditionally, factory floors have had strict divisions of labor, with robots confined to strict safety cages far from humans. Collaborative robots overcome this division of labor, allowing humans and robots to work closely together. In doing so, the advantages of strength and automation of the robot are combined with the flexibility and intuitive nature of the human [12,13]. Evidently, there are numerous advantages to collaborative robotic systems, including economic, social, and ergonomic improvements to traditional systems. However, to harness the full benefits of such systems, companies should adhere to the appropriate safety standards to ensure optimal operation. These will be discussed in the following section.

### 1.3. Safety Standards and HRC Modes

The International Federation of Robotics has reported an all-time high of 517,385 new industrial robots installed in 2021 in factories around the world with a growth rate of 31%, increased 22% in compared to 2018 (pre-pandemic record). Up to now the stock of operational robots around the globe has reached 3.5 million units [16]. With the increasing use of robots in industry, standardization, and guidelines to ensure the safety of human operators are required [10]. Many standards have been proposed to give guidelines for the safe use of collaborative robots. The machinery safety regulated under the Machinery Directive covers the scope of collaborative applications [10]. The following reference standards are reported (see Table 1).

**Table 1.** Safety Standards.

Type	Description	Standard
Type A Standard	Basic safety standards for general requirements	ISO 12100: :2010 "Machine safety, general design principles, risk assessment, and risk reduction" IEC 61508: terminology and methodology
Type B Standard	Generic safety standards	– B1 standards (ISO 13849-1, IEC 62061): specific safety aspects – B2 standards (ISO 13850, ISO 13851): safeguard
Type C Standard	Safety countermeasures for specific machinery	– ISO 10218: safety of industrial robots – ISO 10218-1:2012 "Robots and equipment for robots, Safety requirements for industrial



Prioritized over Type A and Type B standards	robots, Part 1: Robots”, safety requirements for robot manufacturers (robot and controller) – ISO 10218-2: 2011 “Robots and equipment for robots, Safety requirements for industrial robots, Part 2: Systems and integration of robots”, safety requirements for system integrators (robot and ancillary devices), – ISO TS 15066: 2016 “Robots and robotic devices, Collaborative Robots”, guidance on collaborative robot Operations
--	--

Four categories of safety requirements are defined for collaborative robots in the type C international standards ISO 10218-1, ISO 10218-2, and ISO TS 15066 [10,12,13]:

- Safety-rated monitored stop (SMS)

SMS [10,13] is a collaboration arrangement in which robot motion is stopped before a human operator enters the collaborative workspace to interact and carry out a task with the robotic system. This is the most basic form of collaboration, which takes place within a collaborative area, that is, an area of operation shared by the robot and the human. Both parties can work in this area, but not simultaneously, since the robot cannot move if the operator is in the shared space. Therefore, it is ideal for tasks in which the robot primarily works alone and is occasionally interrupted by a human operator. Examples of such tasks include visual inspection, or the positioning of heavy components by the robot for the human.

- Hand-guiding (HG)

Another mode of collaboration is known as hand-guiding [10,13], or ‘direct teach’. In this mode, the operator simply moves the robot to teach it significant positions, without the use of intermediate interfaces such as teach pendants. These positions are communicated as commands to the robot system. Throughout this process, the robot arm weight is compensated such that its position is held. A guiding, hand-operated device is used by the operator to guide the robot’s motion. For this advanced form of collaboration, the robot must be equipped with safety-rated monitored stop and speed functionalities. Once the robot has learned the motion and the human operator has left the collaborative area, the robot may execute the program in automatic mode. However, if the operator enters the area, the program is interrupted. When the operator is using the hand-guiding device, the robot operates in a state of safety-rated monitored speed functionality, until the operator releases the arm and leaves the collaborative area, allowing the robot to resume automatic operation once again.

- Speed and separation monitoring (SSM)

In SSM [10,13], the robot operates even when a human is present, by means of safety-rated monitoring sensors. Thus, human and robot operation takes place simultaneously. To reduce risks, a stipulated protective distance must always be kept between the two parties. If this distance is not kept, the robot operation stops, and resumes once the operator has moved away from the system. If the robot system operates at a reduced speed, the protective distance is reduced accordingly. The workspace may be divided into ‘zones’, whereby if the human is in the green zone the robot may operate at full speed, if in the yellow zone the robot operates with reduced speed, and if the human enters the red zone, robot operation is stopped. Vision systems are used to monitor these zones.

- Power and force limiting (PFL)

PFL [10,13] is a collaborative approach in which limits are set for motor power and force, such that the human operator and robot may work side-by-side. These limits are set as a risk reduction method, defined by a risk assessment. To implement this approach, specific equipment and control modes are required to handle collisions between the robot and human and prevent any injuries to the human.

These four collaborative modes can be applied to both traditional industrial robots and collaborative robots. For traditional industrial robots, additional safety devices such as laser sensors or light curtains are required. On the other hand, for collaborative robot additional features like force and torque sensors, force limits, vision systems, laser systems, and anti-collision systems are required [12].

#### 1.4. HRC Task Types

It is important to analyze the different types of collaboration tasks [10,12,13]. Matheson et al. used the classification that Müller et al. [17] proposed for human robot collaboration in their paper [12], which distinguish HRC task type to four groups: coexistence (same environment, no interaction), synchronized (same workspace, different times), cooperation (same workspace, at the same time, separate tasks), collaboration (same workspace, same task, same time). Wang et al. [18] presented the following types in their paper: coexistence (not sharing workspace, no direct contact), interaction (sharing workspace, communicating with each other, perform task sequentially), cooperation (sharing workspace, have own goals, sharing resources, work simultaneously), collaboration (joint activity, sharing workspace, have the same goal, physical contact allowed). Thiernemann [19] has differentiated four operating modes: manual mode (human), automation(robot), parallelization (same product, direct contact, suitable for pre-assembly), and collaboration (same product, work together). There are other classifications in literature [20,21,22]. To summarize, there are four basic task types for HRC based on literature: coexistence, sequential cooperation, parallel cooperation, and collaboration.

## 2. Materials and Methods

### 2.1. Literature Review and Analysis on HRC Modelling Methods

Choosing the appropriate modelling language is imperative for the development of a robust human-robot collaboration system. This choice depends on various factors, including the particular task requirements, the target platform or framework, and the specific preferences and technical skills of the developers. Commonly used modelling languages in this field include Business Process Modelling Notation (BPMN), Systems Modelling Language (SysML), Behavior Trees (BT), Unified Modelling Language (UML), and Petri Nets. Literature related to each of these languages was reviewed and discussed in this section.

#### 2.1.1. Business Process Modeling Notation (BPMN)

BPMN was originally developed as a standard notation for business users, to bridge the gap between business process design and process implementation. A Business Process Model contains a network of graphical objects representing activities or work, and the flow controls representing the order of performance [23]. While BPMN is typically used for business processes, its capabilities can be extended to the field of robotics. More specifically, it can be used to model the flow, tasks, decisions and interactions that occur in HRC processes.

One approach found that there was a lack of consideration for the modelling of collaborative tasks, and therefore attempted to completely model processes such as assembly workflows, interactions, and decisions taken by machines and humans. It supports variability modelling, making it attractive for customized manufacturing scenarios. A BPMN-based workflow designer was implemented to allow the user to model processes intuitively [24].

BPMN was also used for the development of a risk analysis software tool to support changes in adaptive collaborative robotic scenarios [25]. The work presents a tool which can automatically identify any changes which were made to a particular application's components or processes, providing safety experts with a tool to monitor manufacturing processes. Results showed that this technology provided better usability and decreased errors when compared with conventional methods. BPMN was used as a basis for the development, with certain modifications to make it appropriate for collaborative scenarios.

Another paper tackles the task of warehouse material handling using an automated guide vehicle system [26]. The Manufacturing Process Management System (MPMS) is used to control the process. The aim is to achieve a system which performs all tasks automatically without requiring major changes for various processes. The MPMS was designed using BPMN and Camunda as a platform, allowing for the BPMN model to be executed and controlled in real time.

Concerning the lack of efficiency arising from static task allocation, one paper focused on the use of adaptive task sharing in manufacturing and assembly [27]. A model was developed for experimental purposes, which included a task sharing worker assistance software based on BPMN. The results indicated that adaptive task sharing reduces the productivity gap in automated assembly.

### 2.1.2. Unified Modeling Language (UML)

UML is a popular modelling language in the fields of software engineering and system design. Through its standardized notation, clear visualization and specification of structures, behaviors, and interactions of systems is achieved. UML is therefore thought to be an ideal approach to model architecture, interfaces and interactions in HRC systems. The use of UML in the field of Human-Robot Collaboration seems to mostly focus on safety and risk analysis. This can be seen in [28,29,30], and [31]

F. D. Von Borstel et al [32] use a combination of UML and Colored Petri Nets (CPNs) to model and simulate mobile robots based on wireless robotic components (WRCs). UML diagrams are customized to describe the specific WRC architecture based on the robotic software, particular task and operational environment. The hierarchical CPNs are developed using the customized UML diagrams as a guide. From there, an executable model of the robotic system can be developed. Further work aims to improve the translation from UML to CPNs.

L. Carroll et al [33] attempt using UML in development for the design of real-time robot controllers, based on the necessity for better adaptability from robots when working in cooperation with humans. The main focus was dependability, fault avoidance, fault removal and fault tolerance. UML was found to be appropriate for the development and maintainability of this controller.

As a major software engineering modeling tool, UML can model requirements and high-level system architecture very efficiently. However, UML is not well established for robotics, including HRC. While UML can be useful for modeling certain aspects of human-robot collaboration, limitations arise when applied to the full scope collaborative scenarios. Robots often operate autonomously and need to make decisions, but UML does not offer specialized modeling constructs for representing autonomy and decision-making processes.

### 2.1.3. Systems Modeling Language (SysML)

SysML is a standardized formal language, of which the main purpose is the modelling of systems engineering processes. It is based on UML but is more well-adapted for complex systems in engineering and automation.

Researchers proposed SysML as a modeling language as it was thought to simplify robot programming through abstraction [34]. Since robots are increasingly being integrated into environments where humans are present, they should be made safer, with straightforward programming and interaction capabilities. To achieve this, SysML was used to represent manufacturing processes by graphical diagrams capturing the system structure from various perspectives. While this work did not focus directly on modelling human-robot collaboration, it was one of the motives for the approach, and the capabilities of the system show promise in being adapted for HRC scenarios.

While literature specifically focusing on using SysML for HRC tasks is limited, SysML has been widely used for modelling general robotic workflows. K. Ohara et al [35] propose robot software design based on SysML. The language was chosen due to its benefits for reusability and flexibility. R. Candell et al [36] aimed at achieving real-time observation and control using SysML modelling for architecture, components and information flows. This was based on Industry 4.0 and the resultant requirements for adaptability, flexibility, and responsive communication. SysML is used to develop a graphical model for a wireless factory work cell.

In summary, although SysML can be a useful tool for modelling certain aspects of HRC such as requirements, system architecture and interaction with external systems, SysML is not the most suitable tool for modeling human, real-time interaction, and complex interaction scenarios.

#### 2.1.4. Behavior Trees

Behavior Trees have been widely used in the field of robotics to define the behavior of autonomous agents. The graphical modelling language involves a hierarchical structure of tasks and dependencies, making them an ideal approach to model complex decisions and coordination in HRC scenarios.

The literature related to Behavior Trees was largely concerned with the limitations of automation to capture human dexterity and judgement, especially in unstructured environments where unforeseen events commonly occur. A framework was thus presented which captured various behaviors through which robot autonomy is enhanced. This was done using behavior trees to model the robot's intelligence, including social behaviors, human intention, and various tasks, including collaborative ones [37].

While behavior trees can be a valuable tool for modeling the autonomous behavior of a robotic system, they are less suited for modeling the complexities of human-robot collaboration that involve, for example communication, shared decision-making, and social interaction. Behavior trees tend to focus on predefined actions and decision sequences, making them less flexible in accommodating the varied and sometimes unpredictable nature of human intention. Behavior trees prioritize system-centric decision-making and are not good at handling human-centric decision processes. The design of behavior trees is not human-centered, nor user friendly, this is another drawback of behavior trees modelling.

#### 2.1.5. Petri Nets

Petri Nets were developed primarily for the modelling of systems in which events may occur concurrently, with particular constraints on the concurrence. This is done using an abstract, formal model of the flow of information. This makes Petri Nets an attractive approach for modelling HRC processes, since these generally involve tasks which must be carried out concurrently [38].

Literature indicates an extensive use of Petri Nets for Human-Robot Collaboration. By using time Petri Nets, A. Casalino et al [39] present a scheduling method for collaborative tasks in assembly, which aims at achieving optimal planning and adaptability of assembly process based on runtime knowledge. Time Petri Nets are used for this scheduler. Similar work is carried out in [40] and [41]

Colored Petri Nets were used to model the concurrent scenario in which a human works collaboratively with a wearable robot. The model developed assigns tasks to the robot which are considered too laborious for the human, in a panel installation task. A hybrid control system was developed from the CPN framework, achieving promising results for real-time scheduling and trajectory planning [42].

The main aim of this work carried out by R. E. Yagoda et al [43] was to develop a technique to model Human-Robot Interaction (HRI). The method used Petri nets as the modeling tool for HRI, built upon traditional approaches using job and cognitive work analysis. Petri nets were found to be advantageous as they are generally simple, yet are built upon underlying mathematics, making them powerful for such modelling applications. This work focused on a particular use case, that of UAVS, the control of which is complex and typically involves two or three operators, various stages of flight and varying environments. Petri nets were found to be useful as they are capable of expressing import time dependencies, task concurrencies and event-driven behavior, unlike traditional tools. Additionally, their graphical nature makes them an excellent communication medium, whilst they allow for explicit state modelling since they are state-based, rather than event-based. The research was deemed unique since the behavioral events were modelled using a combination of job analysis, cognitive work analysis and Petri nets.

A scheduling strategy using Time Petri Nets was proposed by A. Casalino et al [44]. The approach was demonstrated in a realistic scenario involving a human operator and dual-arm robot performing an assembly task. While the actions of the human operator were not scheduled, a plan for the activities of the robot was made based on predicted future human intentions. The control policy was designed specifically for this use case but can easily be extended to more general cases.



This is considered future work. An additional further improvement would be to lessen the assumptions on which the scheduling algorithm was based.

Petri Nets have limitations on expressiveness for complex decision making, task allocation, and communication. Their design is mainly used for discrete event modeling other than real-time and continuous motion and perception. Petri Nets also do not naturally integrate with AI and learning algorithms. It is not considered the most suitable choice for modeling the complexity and dynamics of human-robot collaboration, which involves a combination of continuous and discrete interactions, uncertainty, and complex decision-making processes.

Based on the user requirements of the ACROBA project, we need an intuitive and user-friendly user interface for modelling and executing the robotic processes. This basic requirement rules out behavior tree, UML, SysML, and Petri Nets. UML and SysML are more well-adapted for high-level system modeling, requirement, and system integration, behavior trees mainly focus on automation, predefined actions and decision sequences, and Petri Nets lack expressiveness for decision making and communication. BPMN notations are found to be well accepted by ACROBA users in terms of usability. However, standard BPMN does not have robotic elements or notations. Although BPMN can be applied to robotics, the process model gets too complicated for non-experts of BPMN. In general, these modelling tools are all well-established tools in their application area. However, they are not developed specifically for robotics in the HRC domain. In the current versions, it is not possible to separate the HRC task types, specify the HRC modes, verify safety standards, nor monitor human factors.

Therefore, we have created a new modelling language – RTMN for robotic processes that is based on BPMN. It uses the well-known basic notation shapes and builds robotic notations upon them. After our first RTMN version, we continued our research with the aim of improving RTMN, especially in the human robot collaboration area. RTMN 2.0 is the extension of RTMN.

## 2.2. RTMN 2.0 – Extension of RTMN

In this section we will explain the RTMN 2.0 in detail.

### 2.2.1. The RTMN Elements

RTMN originally contained eleven basic notations that are differentiated by shapes, colors, and icons [2]. These notations are developed based on the well-known business process modeling notation (BPMN) [45]. The goal is to develop a modelling language that covers the robotics domain with familiar notations, rather than completely new ones. These notations adopt color, icon, and shapes in their design, aiming to make the modelling more intuitive. Four BPMN standard notations - “sequence flow”, “event” (start, end) and “gate way” (exclusive) are reused in RTMN (see BPMN specification [45]).

There are five new robotics specific notations: robot task, robot skill, robot primitive, HRC (human robot collaboration) task, and one slightly changed notation - human task, similar to a user task in BPMN, however, here it has a human icon and a color assigned to it. A “task” in BPMN is at the lowest level of the process and it cannot be broken down into finer details. It is considered an atomic activity within a process [46]. However, the term “task” in the robotics area is rather a subprocess in relation to BPMN. With the different usage, a picking task for example - it is not an atomic activity, it consists of detailed steps such as to move to the right place, locate the gripper and grasp etc. Therefore, we differentiate a robot task from a typical BPMN task. We define it as an activity consisting of a set of activities which corresponds to a special type of BPMN sub-process that deals with robot activities. The activities within the robot task are defined as skills and primitives. Primitives are atomic activities that compose robotic skills, and skills form robot tasks.

### 2.2.2. The RTMN 2.0 Elements

RTMN 2.0 in comparison to the first version of RTMN has the following extensions:

• Adding HRC modeling elements, including safety in combination with collaboration modes and task types of humans and robots. This has significantly enriched the former RTMN model and enabled it to be applied to HRC application areas.

- Adding requirements, KPI as basic elements
- Adding the link from requirements/KPI to robot control
- Adding decision making elements

RTMN 2.0 is designed especially for modeling human robot collaboration processes. Together with version 1.0 it is capable of modeling light-out automated robotic processes as well as human robot collaboration robotic processes. The model contains two sets of elements: the RTMN elements and the extended elements. The complete overview is presented in Figures 1-3. It consists of all the elements of RTMN 2.0 with the newest improvements and extensions.

Notation	Notation Name	Description	Attribute	Condition	Cardinality	Relation
	Process	Process is a complex activity or set of activities that accomplish a specific organizational goal. It is a orange (#ED7D31) double line rectangle with round corners	has name has KPI has requirement has goal has task has sequenceElement has inputs (goals) has outputs (results) has feedback has precondition has postcondition	N/A	has at least one task	a process can be connected to another process
	Human task	It is an activity that is performed by a human It is a green (#009688) rounded corner rectangle with a single thin line and includes a human figure marker in the top middle of the rectangle	has name has human performer has inputs has output (button/signal)	no robot involvement	has at least 1 human performer	output is used to continue the process
	Robot Task	Robot task is a compound activity that done by robots to perform a complex robot action. It can be broken down into finer level of details. It is a red (#FF5252) rounded corner rectangle with a single thin line and includes a robot figure marker in the top middle of the rectangle, a "plus" sign in the lower-center of the shape indicating that the activity has a lower level of details.	has name has skill has sequenceElement has inputs (goals) has outputs (results) has feedback has precondition has postcondition	no human involvement	no human operator involved.	has skills &/robot task. Used to compose other robot task and processes
	Skill	Robot skills are used to integrate and synchrConize robot actions and sensor data in a consistent way. A robot skill is an elementary functionality or action provided by a robot in order to perform an autonomous mission. It is a blue (#448AFF) rounded corner rectangle with a single thin line and includes a robot figure marker in the top middle of the rectangle, a "plus" sign in the lower-center of the shape indicating that the activity has a lower level of details.	has name has skill has primitive has sequenceElement has inputs has outputs has feedback has precondition has postcondition	no human involvement	has at least 1 primitive	has primitives &/skills Used to compose robot tasks
	Primitive	Primitives are simple, atomic robot activities that can be combined to form more complex robot behaviors. They are the lowest level model elements and provide direct access to hardware elements. It is a green (#4CAF50) rounded corner rectangle with a single thin line and includes a robot figure marker in the top middle of the rectangle	has name has inputs has outputs has feedback has precondition has postcondition	no human involvement	atomic, cannot be broken down to a finer level of detail	receives information from other primitives and skills. Used to compose robot skills

Figure 1. RTMN Modeling Elements – Part A: Basic Notations.

	HRC Task - Coexistence Fence (CF)	Human robot task CF is a compound activity that is performed by both human and robot. It can be broken down into finer level of details. This HRC task type is a combination of coexistence task type and a minimum HRC mode Fence. It is a round corner rectangle with a single thin line. It has two colours the red (#FF5252) represents robot and the green (#009688) represents human. It has a white space in between indicates the robot and human do not work at the same workplace. The thick black line indicates robot in fence. It has a "plus" sign in the lower-center of the shape indicating the activity has a lowerlevel of detail.	has name has sequenceElement has inputs (goals) has outputs (results) has feedback has precondition has postcondition has loopCharacteristics has HRCtaskType has HRCMode has Safety Level			
	HRC Task - Sequential Cooperation SMS(SS)	Human robot task SS is a compound activity that is performed by both human and robot to achieve a common goal. It can be broken down into finer level of details. This HRC task type is a combination of sequential cooperation task type and HRC mode SMS. It is a round corner rectangle with a single thin line. It has two colours the red (#FF5252) represents robot and the green (#009688) represents human. It has an icon indicating the sequential cooperation task type and HRC mode SMS. It has a "plus" sign in the lower-center of the shape indicating the activity has a lowerlevel of detail.	has name has sequenceElement has inputs (goals) has outputs (results) has feedback has precondition has postcondition has loopCharacteristics has HRCtaskType has HRCMode has Safety Level	human and robot are both involved in completing the activity	has at least 1 human task and 1 robot task	has human task and robot task has HRC task type: sequential has HRC mode: SMS has Safety level: >=1
	HRC Task - Teaching HG(TH)	Human robot task TH is a compound activity that performed by both human and robot to achieve a common goal. It can be broken down into finer level of details. This HRC task type is a combination of teaching task type and HRC mode HG. The middle brown area with the joined color of red and green represents the shared workspace by human and robot. It is a round corner rectangle with a single thin line. It has two colours the red (#FF5252) represents robot and the green (#009688) represents human. It has an icon indicating the collaboration task type and HRC mode. It has a "plus" sign in the lower-center of the shape indicating that the activity has a lowerlevel of detail.	has name has sequenceElement has inputs (goals) has outputs (results) has feedback has precondition has postcondition has loopCharacteristics has HRCtaskType has HRCMode has Safety Level	human and robot are both involved in completing the activity	has at least 1 human task and 1 robot task	has human task and robot task has HRC task type: teaching has HRC mode: HG has Safety level: >=2
	HRC Task - Parallel Cooperation SSM(PS)	Human robot task PS is a compound activity that performed by both human and robot to achieve a common goal. It can be broken down into finer level of details. This HRC task type is a combination of parallel cooperation task type and HRC mode SSM. The middle brown area with the joined color of red and green represents the shared workspace by human and robot. It is a round corner rectangle with a single thin line. It has two colours the red (#FF5252) represents robot and the green (#009688) represents human. It has an icon indicating the collaboration task type and HRC mode. It has a "plus" sign in the lower-center of the shape indicating that the activity has a lowerlevel of detail.	has name has sequenceElement has inputs (goals) has outputs (results) has feedback has precondition has postcondition has loopCharacteristics has HRCtaskType has HRCMode has Safety Level	human and robot are both involved in completing the activity	has at least 1 human task and 1 robot task	has human task and robot task has HRC task type: parallel has HRC mode: SSM has Safety level: >=3
	HRC Task - Collaboration PFL(CP)	Human robot task CP is a compound activity that performed by both human and robot to achieve a common goal. It can be broken down into finer level of details. This HRC task type is a combination of collaboration task type and HRC mode PFL. It is a round corner rectangle with a single thin line. It has two colours the red (#FF5252) represents robot and the green (#009688) represents human. It has an icon indicating the sequential collaboration task type and HRC mode SMS. It has a "plus" sign in the lower-center of the shape indicating that the activity has a lowerlevel of detail.	has name has sequenceElement has inputs (goals) has outputs (results) has feedback has precondition has postcondition has loopCharacteristics has HRCtaskType has HRCMode has Safety Level	human and robot are both involved in completing the activity	has at least 1 human task and 1 robot task	has human task and robot task has HRC task type: joint has HRC mode: PFL has Safety level: >=4

Figure 2. RTMN Modeling Elements – Part B: HRC Tasks.

	Requirement	Requirements are KPI targets to achieve certain process goals.	has name has requirementDes has goal has KPI has formula has input has output has precondition has postcondition	N/A	at least one	connecting to process or subprocess
	KPI	KPI stands for key performance indicator, a quantifiable measure of performance over time for a specific objective.	has name has KPIDes has formular	N/A	quantitative KPI, at least one	connecting to requirement
	Sequence Flow	connect flow objects in a process, the arrow direction indicates the workflow direction	has name has sourceRef has targetRef	N/A	can connect only one element on the left and one on the right	connecting process elements
	Data Association	connect data input or output to a flow object in a process.	has name has sourceRef has targetRef	N/A	can connect only one element on the left and one on the right	connecting process elements
	Condition	Condition is rule-based decision making. It is enabled by a rule table where users can specify their decision logic. It is a diamond shape with a rule table icon.	has name has ruleTable has alternativePath	Rule table	system can execute automatically	is connected with multiple sequence flows
	Decision	Decision refers to human decision making. Human decision making allows operators/users to specify a question that will be answered at run time. It is a diamond shape with a human icon.	has name has condition has alternativePath has humanInput	Decision table	Human needs to decide here by giving input	is connected with multiple sequence flows
	Workspace	Workspace is an area in the robotic cell/workplace. It is used to separate the different physical spaces. It is a black rectangle with a smaller rectangle in it on the left.	has name has inputs has outputs has feedback has precondition has postcondition			
	Start	indicating where a process start	has name	N/A	can only be used to start a process	can be connected to a sequence flow
	End	indicating where a process will end.	has name	N/A	can only be used to end a process	a sequence flow can be connected to it

Figure 3. RTMN Modeling Elements – Part C: Other Notations.

### 2.2.3. The RTMN 2.0 Sequence Flow Connection Rules

Figure 4 displays the rules regarding how to connect the elements in a sequence flow. The basic rules are: Start can only start a process, and cannot have any element flow into it; End is the opposite, it ends the process and cannot have any elements flow out of it; Sequence Flow connects the other elements and points out the direction of the process flow; Requirements are normally defined on the process level, but it can also be assigned to tasks, skills and primitives if needed; KPIs are usually used to measure achievement of requirements for the process, but can also be assigned to tasks, skills and primitives if needed; Tasks can be connected to any other task, condition, decision or sequence flow; Skills and primitives can be connected to skills and primitives as well as conditions, decisions and sequence flows.






























From\To																
	Y	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N
	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	N	Y	
	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	N	Y	
	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	N	Y	
	N	N	N	Y	Y	N	N	N	N	N	N	Y	Y	N	Y	
	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	N	Y	
	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	N	Y	
	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	N	Y	
	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	N	Y	
	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	N	Y	
	N	Y	Y	Y	Y	Y	Y	Y	Y	N	N	N	N	N	N	
	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	N	N	N	N	
	N	Y	Y	Y	Y	Y	Y	Y	Y	N	N	Y	Y	N	Y	
	N	Y	Y	Y	Y	Y	Y	Y	Y	N	N	Y	Y	N	Y	
	N	Y	Y	Y	Y	Y	Y	Y	Y	N	N	Y	Y	N	N	
	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	

Figure 4. RTMN 2.0 Sequence Flow Connection Rules.

2.2.4. The HRC Model

We have developed an HRC model that covers the following aspects of HRC: HRC task types, HRC modes, HRC tasks, workspace and decision making. We address every aspect one by one in the next section.

2.2.4.1. Combining Collaboration Task Types and HRC Modes

Based on the literature [10–22] we consolidated the different classifications of the HRC task types and HRC modes. In literature, five HRC modes (fence, SMS, HG, SSM, PFL) and five HRC task types (cell, coexistence, synchronized, cooperation and collaboration) are commonly defined. The HRC modes can ensure the safety of people even when unexpected human intrusions occur. However, the HRC task types do not by definition ensure the safety of people. This classification focuses on how closely humans and robots work together and only the people planned for the task are considered. In reality, no matter how we define the task type, humans can act differently than expected, and



human intrusions can occur. Therefore, to ensure safety each task type must have a minimum safety mode enabled. For this reason, we present a new classification that integrates the HRC task types, HRC modes, and safety levels. Safety level 0 indicates no risk for the human from the robot. It has the lowest safety requirements, as the robot is in a physical fence. Safety levels go up to level 4, which requires the highest safety requirements not only from hardware, but also from software. We also distinguish the collaboration levels (0-5): the higher, the more collaboration between human and robot.

For each task we have defined general safety requirements. Solution design on the hardware and software level are proposed for each HRC task type. This makes it easier for process engineers to ensure safety when implementing HRC processes. The overall approach is presented in Figure 5.

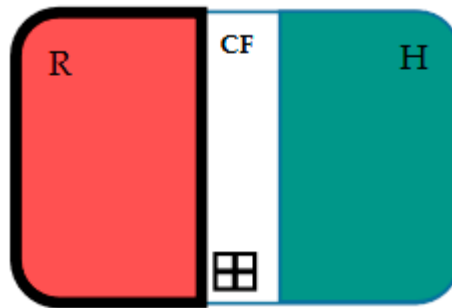
In our new classification we have five categories: Coexistence Fence (CF), Sequential Cooperation SMS (SS), Teaching HG (TH), Parallel Cooperation SSM (PS), and Collaboration PFL (CP). The human robot task type “cell” refers to robot working autonomously surrounded by physical fences without human involvement, so we only need to model the robot tasks and no human tasks need to be modeled. In this case we do not need to create a separate HR task type for it as it has only robot tasks. We have added a new task type “Teaching”, this is when a human takes control of the robot and demonstrates what the robot should do, then the robot learns and repeats it. The reason for adding it is to cover this missing collaboration task type and then to be able to map it to the collaboration mode HG.

Collaboration Level							
Safety Level	HRC Task						General safety(including human intrusion)
	HRC Task Type in Literature	Cell	Coexistence	Sequential Cooperation	Teaching	Parallel Cooperation	
	New HRC Type		Coexistence Fence (CF)	Sequential Cooperation SMS (SS)	Teaching HG (TH)	Parallel Cooperation SSM (PS)	
	Collaboration Level		Level 0	Level 1	Level 2	Level 3	
Safety Level & HRC Modes based on ISO 10218-1/2	Level 0 - Fence						Safe. Robot in cage
	Level 1 - Safety-rated monitored stop						Safe. Robot pauses (safe standstill) when Human enters the collaborative area
	Level 2 - Hand guiding						Safe. Human approaches the collaborative area, the robot program and movements interrupt. As the operator activates the hand guiding device, the robot state switches to safety-rated monitored speed functionality to allow direct movement of robot
	Level 3 - Speed and separation monitoring						Safe. The three zone will ensure any human's safety
	Level 4 - Power and force limiting						Safe. Robot and human can present side-by-side
General safety(including human intrusion) Requirements		Robot in fence	At least fence	At least safety rated monitored stop	At least safety rated monitored stop and safety-rated monitored speed functionalities	At least speed and separation monitoring	Power and force limiting
Safety Level		Level 0	>= Level 0	>= Level 1	>= Level 2	>= Level 3	= Level 4
Solution Design	Hardware Solution (to make robot aware of the surroundings with safety devices)	Industrial robot and Physical fences. If fence opens, robot stops.	Industrial robot/collaborative robot and Hardware that can trigger robot to stop when human enters the robot working area. E.g. Light Curtain, 2D scanner	Industrial robot/collaborative robot and Hardware that can trigger robot to stop when human enters the robot working area. E.g. Light Curtain, 2D scanner	Collaborative robot and Hardware that allow the robot state switches to safety-rated monitored speed functionality to allow direct movement of robot. E.g. hand guiding device	Collaborative robot and Hardware that has zone/position control capabilities. E.g. 2D Scanner, pressure pads/floors, 3D safety vision scanner	Collaborative robot and Hardware that can control the the limitation of motor power and force. E.g. 3D safety vision scanner, vision system
	Software Solution	No additional software functionalities needed	No additional software functionalities needed	No additional software functionalities needed	safety-rated monitored speed functionalities	speed/position control	collision handling, distance, force, velocity, power of robot limitation

**Figure 5.** Combining HRC Task Types and HRC Modes.

- Coexistence Fence (CF)

This type (Figure 6) combines the task type “Coexistence” and collaboration mode “Fence”. This task has the lowest HRC level 0 and should activate at least safety level 0 (Fence) in the background. We recommend using an industrial robot, since the human and robot do not work together in the same workplace. The fence for the robot can be in the form of physical barriers or certified hardware like a light curtain/2D scanner. As long as Fence mode is insured, this task is safe. No additional software functionalities are needed.



**Figure 6.** Coexistence Fence Notation.

- Sequential Cooperation SMS (SS)

This type (Figure 7) combines the task type “Synchronized” and collaboration mode “Safety-rated monitored stop”. “Safety-rated monitored stop” means “condition where the robot is stopped with drive power active, while a monitoring system with a specified sufficient safety performance ensures that the robot does not move” as described in [47].

This task has collaboration level 1 and should activate at least safety level 1 (SMS) in the background. We recommend using either an industrial robot or a collaborative robot with safety certified hardware like light curtain/2D scanner to ensure SMS (robot stops when human enters the robot working area). As long as SMS mode is insured, this task is safe. No additional software functionalities are needed.



**Figure 7.** Sequential Cooperation SMS Notation.

- Teaching HG (TH)

This type (Figure 8) combines the task type “Teaching” and collaboration mode “Hand Guiding”. This task has collaboration level 2 and should activate at least safety level 2 (HG) in the background. We recommend using a collaborative robot with safety certified hardware that allows the robot state to switch to safety-rated monitored speed functionality to allow direct movement of robot. e.g., hand guiding device. If a minimum safety HG mode is insured, this task is safe. Safety-rated monitored speed functionalities need to be programmed at the software level. In ISO 10218-1 safety-rated monitored speed is written as “safety-rated function that causes a protective stop when

either the Cartesian speed of a point relative to the robot flange (e.g. the TCP), or the speed of one or more axes exceeds a specified limit value" [47].



**Figure 8.** Teaching HG Notation.

- Parallel Cooperation SSM (PS)

This type (Figure 9) combines the task type “Parallel Cooperation” and collaboration mode “Speed and Separation Monitoring”. This task has collaboration level 3 and should activate at least safety level 3 (SSM) in the background and put in place the hardware that insures the SSM mode. We recommend using a collaborative robot with safety certified hardware that has zone/position control capabilities, e.g., 2D scanner, pressure pads/floors, 3D safety vision scanner. Speed/position control functionalities need to be programmed at the software level. The work area should be divided into three zones: green zone (robot runs at full speed), yellow zone (robot runs at reduced speed), red zone (robot stops). As long as SSM mode is insured, this task is safe.



**Figure 9.** Parallel Cooperation SSM Notation.

- Collaboration PFL (CP)

This type (Figure 10) combines the task type “Collaboration” and collaboration mode “Power and Force Limiting”. This task has collaboration level 4 and should activate the safety level 4 (Power and Force Limiting) at the background. This type is the most collaborative task type, requires the highest safety controls and demands the most software engineering effort. We recommend using a collaborative robot with safety certified hardware that can control the limitation of motor power and force, e.g., a 3D safety vision scanner or a vision system. For force control and limitation force-torque sensors in end-effectors and/or in the joints of the robot are required. At the software level, functionalities need to be developed for collision handling, distance monitoring, force control, velocity tracking, and power of robot limitation. For example, a simple calculation for the maximum separation distance can be defined by multiplying the  $Trs$  (time for robot to stop) and  $Sr$  (robot speed). Therefore,  $Drh$  (Distance between robot and human),  $Max\ Drh = Trs * Sr$ . As long as PFL mode is insured, this task is safe. In the property of this notation, one must assign an HRC mode and task type to it.



Figure 10. Collaboration PFL Notation.

2.2.4.2. Workspace

To align the modeling with the realistic robotic cell, we introduced the workspace notation. This notation is similar to the BPMN pool/lane (see Figure 11). In BPMN, pools represent companies, departments, or roles. Lanes represent sub-entities within these organizations and appear as swim lanes inside the pool [3]. In robotic processes, there are usually only limited humans involved in the process. In the robotic cell the focus is not the different entities like roles or departments, but the different areas in the cell. Therefore, we propose to use pools/lanes as workspaces for separating the different areas in the robotic cell.

One can use workspace when tasks in the process are carried out in multiple workspaces. When tasks switch from one workspace to another, there will be a pop-up window or voice reminder of a workspace change. A workspace can have input and output, precondition and postcondition assigned to it. With these properties one can set up specific requirements for the workspace. Figure 12 shows an example of using the workspace notation.

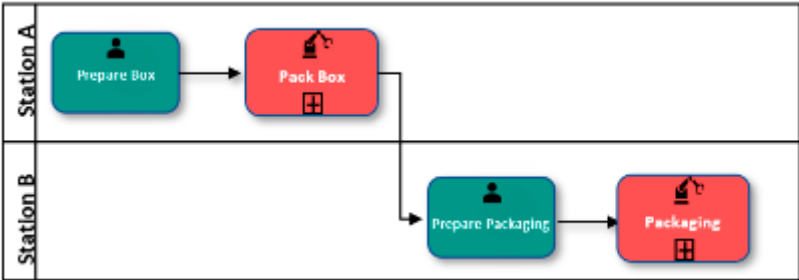


Figure 11. Example of Using Workspace Notation.



Figure 12. Workspace Notation.

2.2.4.3. Decision Making

We model decision making with two different notations. We call one “Condition” (See Figure 13) which is rule-based decision making and the other “Decision” (See Figure 14) which refers to human decision making.





Figure 13. Decision Notation.



Figure 14. Condition Notation.

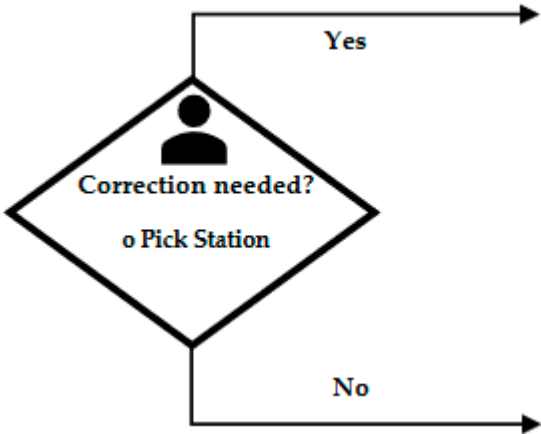
Rule-based decision making is enabled by a rule table where users can specify their decision logic. Table 2 shows how the rule table looks, with some examples to demonstrate the usage of it.

Table 2. Rule Table.

Sequence	Parameter	Equation	Value	Action
1	Bin Status	=	Empty	Fill up bin
		≠	Empty	Do nothing
2	Distance Robot&Human	=	Trs * Sr Trs(time for robot to stop), Sr(robot speed), Dhr(Distance between robot and human)	Stop robot
		<	Trs * Sr	Robot run at collaborative speed
3	Quality check	=	accepted	Move product to next station
		=	rejected	Move to rejected bin

The rule table checks whether the conditions are met. Depending on whether the conditions are met or not, a corresponding action will be followed. Often there is only one parameter that is being checked, for example “quality check”. But there are also cases where multiple parameters are checked. In this case we need to define the sequence of the condition check. We use a sequence column to indicate that the parameters are checked sequentially one by one. When the rule table is defined, this sequential execution must be followed.

Human decision making allows operators/users to specify a question that will be answered at run time. At this step, a question will pop up and ask the operator/user to give an input (commonly a yes or no answer). Based on what the operator chooses (yes or no) a different branch of the process flow will be executed. Figure 15 gives an example of using the decision notation.



**Figure 15.** Example of Using Decision Notation.

2.2.5. Other Extensions and Modifications

In RTMN 2.0 there are other extensions. We also made some modifications to the notations in RTMN first version.

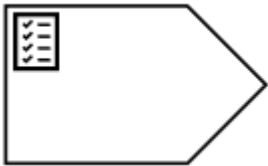
2.2.5.1. Requirements and KPI

Key Performance Indicators (KPIs) form a specialized set of metrics steering management activities towards improving company performance, representing quantitative measures of critical success factors (CSFs) in enterprises [48]. They serve as variables reflecting progress toward goals while aligning with the company's vision and strategies [48]. Displaying KPIs is essential for supervisors and employees to seek improvements and reaching for higher performance [48]. In an HRC environment, safety and ergonomics are widely acknowledged as crucial pillars, while in a business context, Productivity, encompassing both Effectiveness and Efficiency, and Economics must be considered [48].

Therefore, after analysing numerous research papers Caiazzo et al. divided the various KPIs related to HRC to the following four areas: 'Productivity', 'Economics', 'Safety', and 'Ergonomics' [48]. We adopt their systematic approach and categorise the requirements and KPIs into these four categories.

The requirement notation (see Figure 16) and KPI notation (see Figure 17) are added to RTMN 2.0 to visualize the business requirements and KPIs on the process level. We can also assign requirements and KPIs at the task level if needed.

The requirement notation (see Figure 16) represents business requirements of a robotic process/task. It has the following properties: name, formula, KPI, input, output, precondition, and postcondition. One can set constraints using preconditions and postconditions. For example, one can add a postcondition - the sum of net machine utilization and net operator actuation is 100%. The KPI notation defines the key performance indicators linked to the requirements (see Figure 17). Table 3 illustrates the general KPIs for the HRC robotic processes, among which a few are related to human: Accident Rate, Net Operator Actuation, Human Exposure to Chemical.



**Figure 16.** Requirement Notation.

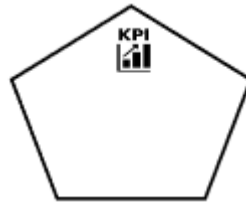


Figure 17. KPI Notation.

Table 3. Requirements and KPIs.

Requirement	Requirement Formula	Name of KPI	KPI Formula
Decrease cycle time by 10%	$(\text{Cycle Time old} - \text{Cycle time new}) / \text{Cycle Time old} \leq 10\%$	Cycle Time	$\text{Cycle Time} = \text{Process end time} - \text{Process begin time}$
Increase productivity by 20%	$(\text{Productivity old} - \text{Productivity new}) / \text{Productivity old} \geq 20\%$	Productivity	Units of Product/ production time (hours)
Reach success rate of 80%	Success Rate $\geq 80\%$	Success Rate	Nr of success action/ Nr of total actions
Reach reprogramming time less than 10 minutes	Reprogramming Time $< 10\text{min}$	Reprogramming Time	time finish reconfiguration - Time start reconfiguration
Reduce scrap product to less than 5%	Nr of Scrap Product/Nr of total product $\leq 5\%$	Scrap Product	the number of products not accepted in quality inspection
Increase machinery utilization to more than 75%	Net Machine Utilization $\geq 75\%$	Net Machine Utilization	machine run hours per process/ process duration
Reduce Net operator actuation to less than 25%	Net Operator Actuation $\leq 25\%$	Net Operator Actuation	Human working hours per process/ process duration
Reduce accident rate to 0	Accident Rate = 0	Accident Rate	the number of Reportable health & safety incidents per month
Increase Human safety /reduce exposure to chemicals/danger	1- Human exposure to chemical rate $\geq 30\%$	Human Exposure to Chemical	Time of human exposure to Chemicals/danger/ cycle time

#### 2.2.5.2. Traceability of Requirements and KPIs

The inputs for calculating the KPIs come from ROS - the robot control level. From primitives and skills, we can get the input values and use them for calculating the KPIs and validate the requirements. Through this, the traceability of requirements and KPIs are achieved.

To be more specific, KPIs are calculated automatically based on the values created in the different layers of the robot system. For example, for a pick skill, whether the pick is successful or not is tracked. The success rate of picking is calculated by the number of success picks / total pick attempts.

Another example – the robotic task Scrap Product – quality inspection of the number of accepted products and not accepted products is tracked. If the product is within the error tolerance, it will be accepted and vice versa. The system can add up the number of accepted products and the number of not accepted products, and then Product quality is calculated by dividing the number of products not accepted by the number of total products.

2.2.5.3. Robotic Process

A robotic process involves robots in the process. We have added this notation (see Figure 18) because we want to visually show the relationship between business requirement and the overall process. A process can contain human tasks, robot tasks and HRC tasks.



Figure 18. Robotic Process Notation.

The robotic process has data properties: name, owner, and organization. Requirement/Objective and KPIs are no longer just properties of the process. They have their own notations as described earlier.

Figure 19 provides an example of modelling KPIs and requirements for a process. The dashed arrows indicate data association.

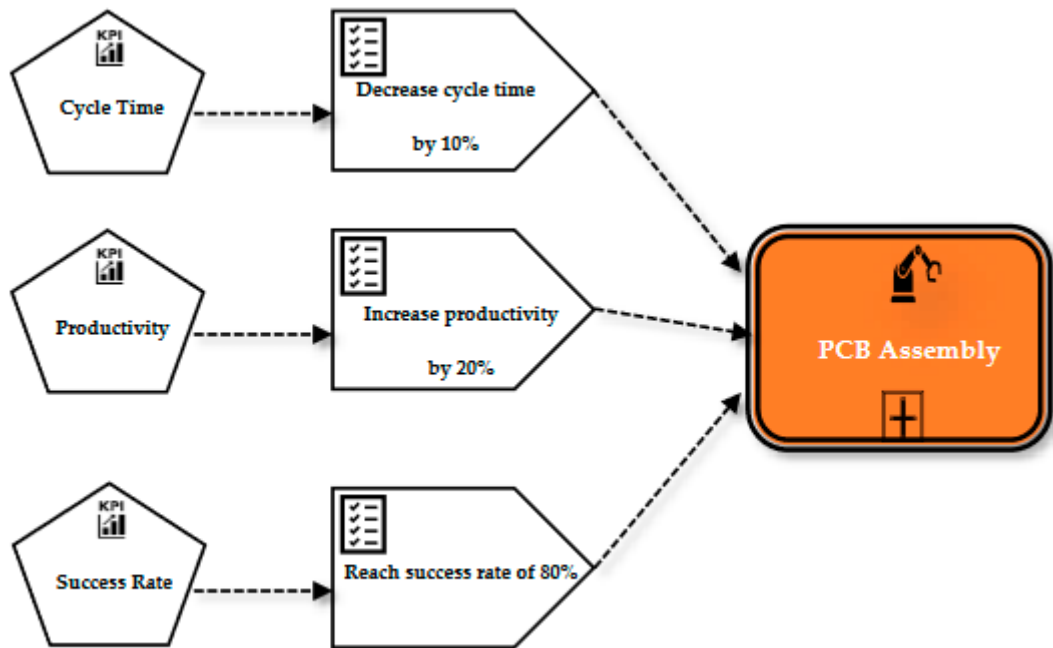


Figure 19. Process-Requirement-KPI Example.

2.2.5.4. Skill and Primitive

We modified the icons of the skill and primitive notation in order to easily distinguish the robot task and robot skill visually, whilst bringing skill and primitive closer. Now both skill and primitive have a robot body icon (See Figure 20 and 21). Robot task has a robot arm icon.



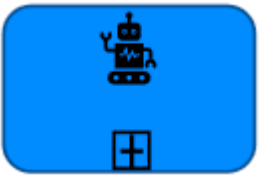


Figure 20. Skill Notation.

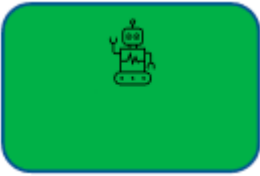
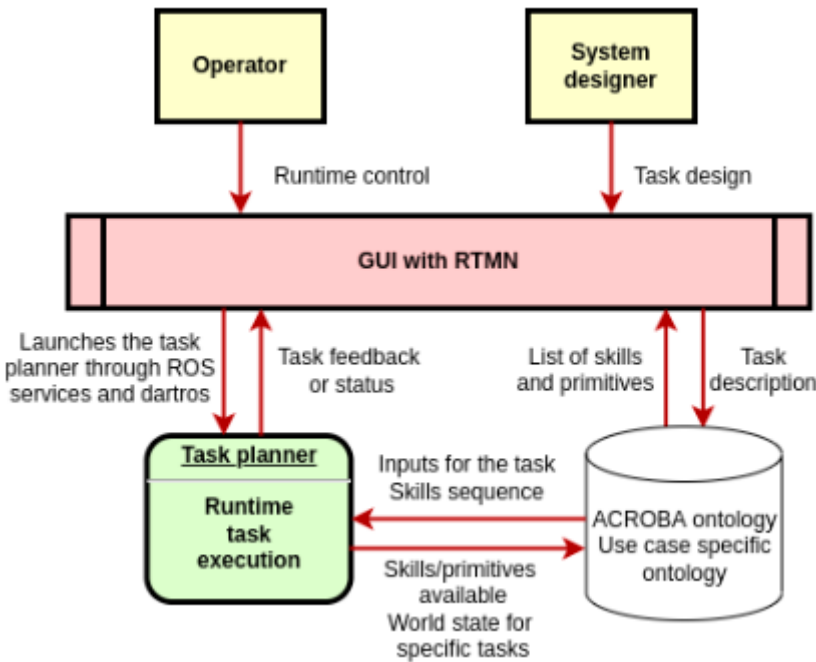


Figure 21. Primitive Notation.

2.2.6. The Implementation

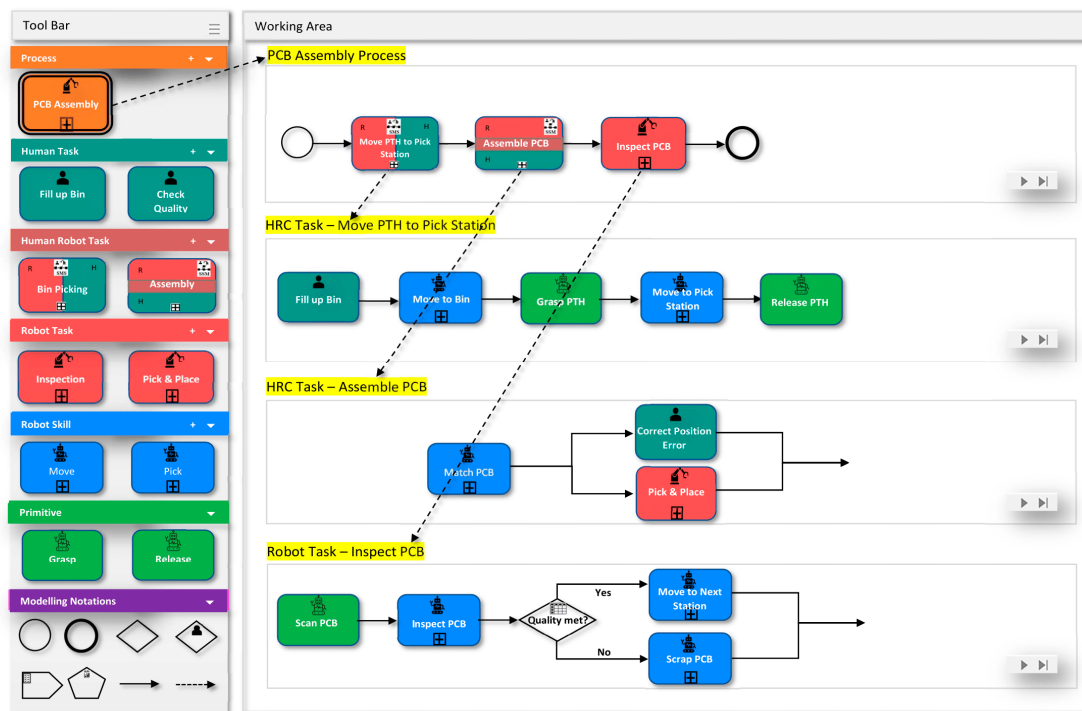
As described in the former RTMN paper [2], the modelling language RTMN is used in the graphical user interface (GUI) of the ACROBA platform. The notations from RTMN 2.0 are defined as drag and drop elements in the GUI. The GUI is developed on the Flutter framework [49]. Flutter uses Dart as its programming language. The special package Dartros [50] enables Flutter to communicate with ROS [51] by implementing the ROS client library. The overall architecture is shown in Figure 22 (taken from the former paper).

The GUI using RTMN 2.0 serves as a modelling tool. The users can create a process model in the GUI. The process sequence is then sent from the GUI to the task planner through Datros. The task planner takes the sequence and “translates” it to a behaviour tree which is connected to ROS. Task planner also provides feedback and status of the tasks to the GUI. The population of the available modelling elements such as skills and primitives are managed through an ontology where the domain knowledge such as tasks, skills and primitives are defined. The characteristics of ontology ease the sharing of knowledge, provide clear structure of data, and enable accurate reasoning of data[52]. Therefore, we also adopt the benefits of ontology for knowledge-sharing among heterogeneous systems and human-machine interaction.



**Figure 1.** RTMN 2.0 Framework [2]**Demonstration**

Figure 22 shows the design of the Graphical User Interface (GUI) for RTMN 2.0. The left side is divided into different abstraction levels: processes, tasks, skills and primitives. At the bottom left corner, the connecting notations are shown. One can simply drag and drop the notations on the left to model an HRC process.

**Figure 23.** HRC Process Example - PCB Assembly.

We illustrated one example of an HRC process. This example is a modification of a use case process due to confidentiality. A Printed Circuit Board (PCB) assembly process is a process to assemble a PCB. The assembly works by putting different kinds of Pin through Hole (PTH) components on the PCB. This process consists of three tasks: Move PTH to Pick Station, Assemble PCB, Inspect PCB. Move PTH to Pick Station and Assemble PCB are two HRC tasks, while Inspect PCB is a robot task.

For the HRC task Move PTH to Pick Station, safety mode SMS should be ensured because it is a sequential HRC task with human and robot working one after another. It has a human task, Fill Up Bin, followed by a robot skill Move to Bin, then a primitive, Grasp PTH, robot skill Move to Pick Station, and primitive, Release PTH.

For the HRC task Assemble PCB, safety mode SSM should be ensured as human and robot work in parallel in this task. They work side by side on the same PCB, but not at the same time. When a PTH is wrongly positioned, the human needs to correct it. As the human approaches the robot and enters the “red” zone, the robot stops. This task consists of a simple robot skill – match PCB (this skill scans PCB, then, based on the scanned image loads the CAD model of the PCB) and two parallel tasks (combination of skills)- Correct Position Error by human and Pick&Place by robot. The join/split notation is used here to split the sequence to two, then merge the two back to one again.

The robot task Inspect PCB does not involve any human. It is purely done by a robot. In this task a condition notation is used to decide where the PCB goes after inspection. If it is accepted, it is moved to the next station. If it is rejected, the PCB is scrapped.

### 3. Results

As we described in our former paper [2], we adopt the “technology acceptance model” [53] from Davis and Venkatesh for developing and validating our approach. Based on Davis and Venkatesh, perceived ease of use and perceived usefulness of prototypes have been shown to predict the real use of the technology months later [53]. We performed an early evaluation of our approach and the feedback received from 17 partners and 82 participants was positive. They perceive the GUI with RTMN as useful for managing their work and expressed their willingness to use the tool. Additionally, they found the design easy to understand and intuitive to use. Detailed data were shown in our former paper [2]. According to [53] this early evaluation of the tool will very well predict its intended use by the users after implementation.

We plan to perform another round of evaluation of the GUI after implementation. We will analyze the results and assess the benefits and drawbacks of the GUI. Based on the feedback, further improvements will be made.

### 4. Discussion

This paper does not yet cover the details of modeling of human factors related to mental safety (mental stress and anxiety) induced by close interaction with robots. The authors will continue their research in this area and aim to develop a comprehensive approach for modeling human factors.

### 5. Conclusions

This paper covers the extension and modification of the RTMN. RTMN 2.0 focuses on modelling of human robot collaboration processes. Modelling of requirements and KPIs are added to enable the traceability from a process to the lower robotic control level. Notations related to decision making are modified as an improvement of RTMN. These additional features of RTMN 2.0 together extend RTMN to the HRC process domain and increase the intuitiveness and reusability of robot process modelling. This allows people without programming expertise to plan and control robots in an easier way.

**Author Contributions:** Conceptualization, C.Z.S., J.A.C.R. and N.U.B.; methodology, C.Z.S., J.A.C.R. and N.U.B.; software, X.X.; validation, C.Z.S.; formal analysis, C.Z.S.; investigation, X.X.; resources, C.Z.S.; data curation, C.Z.S.; writing—original draft preparation, C.Z.S.; writing—review and editing, C.Z.S., J.A.C.R. and N.U.B.; visualization, C.Z.S.; supervision, J.A.C.R. and N.U.B.; project administration, C.Z.S. and N.U.B.; funding acquisition, J.A.C.R. and N.U.B.. All authors have read and agreed to the published version of the manuscript.”.

**Funding:** This research was supported by ACROBA. The ACROBA project has received funding from the European Union’s Horizon 2020 research and innovation program under grant agreement No 1017284. Juan Antonio Corrales Ramón was funded by the Spanish Ministry of Universities through a ‘Beatriz Galindo’ fellowship (Ref. BG20/00143) and by the Spanish 1 Ministry of Science and Innovation through the research project PID2020-119367RB-I00.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Data collected from use cases is available on request. We store them safely under the ACROBA project repository.

**Acknowledgments:** The authors would like to thank the ACROBA project for providing the opportunity and funding for this research. Special thanks to the ACROBA consortium for the contributions and support. The authors would like to extend their thanks also to Katrina Mugliett for her excellent assistance in reference management, texts writing and formatting.

**Conflicts of Interest:** The authors declare no conflict of interest.

### References

1. Project - Acroba Project Available online: <https://acrobaproject.eu/project-acroba/> (accessed on 28 February 2022).
2. Sprenger, C.Z.; Ribeaud, T. Robotic Process Automation with Ontology-Enabled Skill-Based Robot Task Model and Notation (RTMN). *2022 2nd International Conference on Robotics, Automation and Artificial Intelligence (RAAI)* **2022**, 15–20, doi:10.1109/RAAI56146.2022.10092996.
3. BPMN Specification - Business Process Model and Notation Available online: <https://www.bpmn.org/> (accessed on 27 February 2022).
4. Li, Y.; Ge, S.S. Human-Robot Collaboration Based on Motion Intention Estimation. *IEEE/ASME Transactions on Mechatronics* **2014**, *19*, 1007–1014, doi:10.1109/TMECH.2013.2264533.
5. Weiss, A.; Wortmeier, A.K.; Kubicek, B. Cobots in Industry 4.0: A Roadmap for Future Practice Studies on Human-Robot Collaboration. *IEEE Trans Hum Mach Syst* **2021**, *51*, 335–345, doi:10.1109/THMS.2021.3092684.
6. SIGCHI (Group: U.S.); SIGART; IEEE Robotics and Automation Society; Institute of Electrical and Electronics Engineers *HRI'16: The Eleventh ACM/IEEE International Conference on Human Robot Interaction: March 7-10, 2016, Christchurch, NZ*; ISBN 9781467383707.
7. Institute of Electrical and Electronics Engineers.; Institute of Electrical and Electronics Engineers. Region 3. *IEEE SoutheastCon 2015: April 9-12, Fort Lauderdale, Florida, Hilton Fort Lauderdale Marina.*; ISBN 9781467373005.
8. Freedy, A.; DeVisser Gershon Weltman, E. Measurement of Trust in Human-Robot Collaboration;
9. IEEE Staff; IEEE Staff 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems.; ISBN 9781424466764.
10. Kumar, S.; Savur, C.; Sahin, F. Survey of Human-Robot Collaboration in Industrial Settings: Awareness, Intelligence, and Compliance. *IEEE Trans Syst Man Cybern Syst* **2021**, *51*, 280–297, doi:10.1109/TSMC.2020.3041231.
11. 2011 IEEE International Symposium on Assembly and Manufacturing.; IEEE, 2011; ISBN 9781612843438.
12. Matheson, E.; Minto, R.; Zampieri, E.G.G.; Faccio, M.; Rosati, G. Human-Robot Collaboration in Manufacturing Applications: A Review. *Robotics* **2019**, Vol. 8, Page 100 **2019**, 8, 100, doi:10.3390/ROBOTICS8040100.
13. Villani, V.; Pini, F.; Leali, F.; Secchi, C. Survey on Human-Robot Collaboration in Industrial Settings: Safety, Intuitive Interfaces and Applications.
14. Kim, A. A Shortage of Skilled Workers Threatens Manufacturing's Rebound.
15. Vysocky, A.; Novak, P. Human - Robot Collaboration in Industry. *MM Science Journal* **2016**, 2016-June, 903–906, doi:10.17973/MMSJ.2016\_06\_201611.
16. International Federation of Robotics Available online: <https://ifr.org/> (accessed on 18 July 2023).
17. Müller, R.; Vette, M.; Geenen, A. Skill-Based Dynamic Task Allocation in Human-Robot-Cooperation with the Example of Welding Application. *Procedia Manuf* **2017**, *11*, 13–21, doi:10.1016/J.PROMFG.2017.07.113.
18. Wang, L.; Gao, R.; Váncza, J.; Krüger, J.; Wang, X. V; Makris, S.; Chryssolouris, G. Symbiotic Human-Robot Collaborative Assembly. **2019**, doi:10.1016/j.cirp.2019.05.002.
19. Maschinenbau, F.; Thiernemann Aus Gernlinden, Dipl.-I.S. Direkte Mensch-Roboter-Kooperation in Der Kleinteilemontage Mit Einem SCARA-Roboter. **2004**.
20. Müller, R.; Vette, M.; Mailahn, O. Process-Oriented Task Assignment for Assembly Processes with Human-Robot Interaction. *Procedia CIRP* **2016**, *44*, 210–215, doi:10.1016/J.PROCIR.2016.02.080.
21. Vincent Wang, X.; Kemény, Z.; Váncza, J.; Wang, L. Human-Robot Collaborative Assembly in Cyber-Physical Production: Classification Framework and Implementation. **2017**, doi:10.1016/j.cirp.2017.04.101.
22. Krü, J.; Lien, T.K.; Verl, A. Cooperation of Human and Machines in Assembly Lines., doi:10.1016/j.cirp.2009.09.009.
23. White, S.A. Introduction to BPMN; 2004;
24. 2018 IEEE 16th International Conference on Industrial Informatics (INDIN).; IEEE, 2018; ISBN 9781538648292.
25. Pantano, M.; Pavlovskiy, Y.; Schulenburg, E.; Traganos, K.; Ahmadi, S.; Regulin, D.; Lee, D.; Saenz, J. Novel Approach Using Risk Analysis Component to Continuously Update Collaborative Robotics Applications in the Smart, Connected Factory Model. *Applied Sciences (Switzerland)* **2022**, *12*, doi:10.3390/app12115639.
26. Universidad de Zaragoza; Institute of Electrical and Electronics Engineers; IEEE Industrial Electronics Society; Universidad de Zaragoza. Aragón Institute for Engineering Research *Proceedings, 2019 24th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA): Paraninfo Building, University of Zaragoza, Zaragoza, Spain, 10-13 September, 2019*; ISBN 9781728103037.
27. Schmidbauer, C.; Schlund, S.; Ionescu, T.B.; Hader, B. Adaptive Task Sharing in Human-Robot Interaction in Assembly. In *Proceedings of the IEEE International Conference on Industrial Engineering and Engineering Management*; IEEE Computer Society, December 14 2020; Vol. 2020-December, pp. 546–550.
28. Guiochet, J. Hazard Analysis of Human-Robot Interactions with HAZOP-UML. *Saf Sci* **2016**, *84*, 225–237, doi:10.1016/j.ssci.2015.12.017.



29. Martin-Guillerez, D.; Guiochet, J.; Powell, D.; Zanon, C. A UML-Based Method for Risk Analysis of Human-Robot Interactions. In Proceedings of the Proceedings of the 2nd International Workshop on Software Engineering for Resilient Systems; Association for Computing Machinery: New York, NY, USA, 2010; pp. 32–41.
30. Guiochet, J.; Motet, G.; Baron, C.; Boy, G. TOWARD A HUMAN-CENTERED UML FOR RISK ANALYSIS Application to a Medical Robot;
31. Annual IEEE Computer Conference; IEEE International Conference on Rehabilitation Robotics 13 2013.06.24-26 Seattle, Wash.; ICORR 13 2013.06.24-26 Seattle, Wash. 2013 *IEEE International Conference on Rehabilitation Robotics (ICORR) 24-26 June 2013, Seattle, Washington, USA*; ISBN 9781467360241.
32. Von Borstel, F.D.; Villa-Medina, J.F.; Gutiérrez, J. Development of Mobile Robots Based on Wireless Robotic Components Using UML and Hierarchical Colored Petri Nets. *Journal of Intelligent and Robotic Systems: Theory and Applications* **2022**, *104*, doi:10.1007/s10846-021-01549-1.
33. Carroll, L.; Tondu, B.; Baron, C.; Geffroy, J.C. UML Framework for the Design of Real-Time Robot Controllers;
34. Verband der Elektrotechnik, E.; Institute of Electrical and Electronics Engineers.; German Conference on Robotics (8th : 2014 : Munich, G. *ISR/Robotik 2014 ; 41st International Symposium on Robotics ; Proceedings of: Date 2-3 June 2014.*; ISBN 9783800736010.
35. Ohara, K.; Takubo, T.; Mae, Y.; Arai, T. SysML-Based Robot System Design for Manipulation Tasks;
36. Candell, R.; Kashef, M.; Liu, Y.; Foufou, S. A SysML Representation of the Wireless Factory Work Cell: Enabling Real-Time Observation and Control by Modeling Significant Architecture, Components, and Information Flows. *International Journal of Advanced Manufacturing Technology* **2019**, *104*, 119–140, doi:10.1007/s00170-019-03629-x.
37. Avram, O.; Baraldo, S.; Valente, A. Generalized Behavior Framework for Mobile Robots Teaming With Humans in Harsh Environments. *Front Robot AI* **2022**, *9*, doi:10.3389/frobt.2022.898366.
38. Peterson, J.L. *Petri Nets\**; 1977;
39. Casalino, A.; Zanchettin, A.M.; Piroddi, L.; Rocco, P. Optimal Scheduling of Human-Robot Collaborative Assembly Operations With Time Petri Nets. *IEEE Transactions on Automation Science and Engineering* **2019**, doi:10.1109/TASE.2019.2932150.
40. Chao, C.; Thomaz, A. Timed Petri Nets for Fluent Turn-Taking over Multimodal Interaction Resources in Human-Robot Collaboration. *Int J Rob Res* **2016**, *35*, 1330–1353, doi:10.1177/0278364915627291.
41. Chao, C.; Thomaz, A. Timing in Multimodal Turn-Taking Interactions: Control and Analysis Using Timed Petri Nets. *J Hum Robot Interact* **2012**, *4*–25, doi:10.5898/jhri.1.1.chao.
42. Institute of Electrical and Electronics Engineers. Robotics and Automation (ICRA), 2014 IEEE International Conference on : May 31 2014-June 7 2014 : [Hong Kong, China].; ISBN 9781479936854.
43. Yagoda, R.E.; Covert, M.D. How to Work and Play with Robots: An Approach to Modeling Human-Robot Interaction. *Comput Human Behav* **2012**, *28*, 60–68, doi:10.1016/j.chb.2011.08.011.
44. Casalino, A.; Cividini, F.; Zanchettin, A.M.; Piroddi, L.; Rocco, P. Human-Robot Collaborative Assembly: A Use-Case Application.; Elsevier B.V., January 1 2018; Vol. 51, pp. 194–199.
45. Völzer, H. An Overview of BPMN 2.0 and Its Potential Use. *Lecture Notes in Business Information Processing* **2010**, *67 LNBIP*, 14–15, doi:10.1007/978-3-642-16298-5\_3.
46. Business Process Model and Notation (BPMN), Version 2.0. **2010**.
47. ISO - International Organization for Standardization Available online: <https://www.iso.org/obp/ui/en/#iso:std:iso:10218:-1:ed-2:v1:en> (accessed on 7 November 2023).
48. Caiazzo, C.; Nestić, S.; Savković, M. A Systematic Classification of Key Performance Indicators in Human-Robot Collaboration. *Lecture Notes in Networks and Systems* **2023**, *562 LNNS*, 479–489, doi:10.1007/978-3-031-18645-5\_30/TABLES/1.
49. Flutter Available online: <https://flutter.dev/> (accessed on 15 August 2022).
50. Dartros | Dart Package Available online: <https://pub.dev/packages/dartros> (accessed on 15 August 2022).
51. ROS: Home Available online: <https://www.ros.org/> (accessed on 28 February 2022).
52. Pang, W.; Gu, W.; Li, H. Ontology-Based Task Planning for Autonomous Unmanned System: Framework and Principle. In Proceedings of the Journal of Physics: Conference Series; IOP Publishing Ltd, April 19 2022; Vol. 2253.
53. Davis, F.D.; Venkatesh, V. Toward Preprototype User Acceptance Testing of New Information Systems: Implications for Software Project Management. *IEEE Trans Eng Manag* **2004**, *51*, doi:10.1109/TEM.2003.822468.

**Disclaimer/Publisher's Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.