

ArtWork

SMART AND CONNECTED WORKER

ITEA 4 – 22019

Work package 3

Intelligent Generation of intuitive Work Instructions

Deliverable 3.4

Constant knowledge update pipeline
with connectivity to production systems

Document type	: Deliverable
Document version	: 1.0
Document Preparation Date	: 2026-09-25
Classification	: Confidential
Contract Start Date	: 2024-03-01
Contract End Date	: 2027-02-28



Final approval	Name	Partner
Review Task Level	All partners involved	
Review WP Level	Olov Schelén	LTU
Review Board Level	Atieh Hanna Thomas Baer	Volvo Daimler

Content

1	Background and Scope	2
2	Overview of mixed model assembly line planning.....	3
3	Proof-of-Concepts	4
3.1	Product sequencing decision support	4
3.2	Assembly Line Rebalancing and Task Sequencing	11
3.3	Worker assignment - decision support for employee-job matching	15
3.4	Workstation layout planning and task balancing.....	19
3.5	Task sequencing and worker instructions.....	21
3.6	Teaching instructions generation	34
4	Statements from the usecase owners	41
4.1	Volvo	41
4.2	Daimler.....	41
4.3	JVC.....	41
5	Conclusions	42
6	Dissemination results	42

1 Background and Scope

The Scope of WP3 is on software tools to assist workers (e.g., engineers, line managers, assembly line operators) in more effective planning to facilitate efficient production on assembly lines. The target assembly lines support mixed model production with large product variability, typically involving many human operators performing advanced task sequences. Human-centric aspects are therefore essential, including the mix of different operator roles, rotation, and ergonomics.

There are many kinds of planning needed to cater for different aspects of the production such as line balancing, operator role planning, task sequencing across and inside stations, station layout planning, dynamic manning of the operator roles, product sequencing, etc. This is explained in section 2.

To tackle this complex and multi-dimensional planning space, we use a divide and conquer approach with the development of a few proof-of concept implementations that focus on some parts of the problem space that can be tackled independently on the detailed level with inner iterations inside the tools, yet with dependencies so that feedback and replanning on the overall level can be performed across the tools with outer iterations.

The scope of this deliverable (D3.4) is to give an overview of the different kinds of planning that are needed for assembly lines (Section 2), and the the Proof-of-Concept (PoC) tools developed as part of WP3, including workflows, results and outcomes from simulation and verification (Section 3). The presentations are supported by screen dumps of GUIs and/or examples of API input output commands and data. Statements from the use-case owners on the business value of the PoCs are presented (Section 4). The results are concluded (Section 5) and finally the past and ongoing dissemination plan is presented (Sections 6).

2 Overview of mixed model assembly line planning

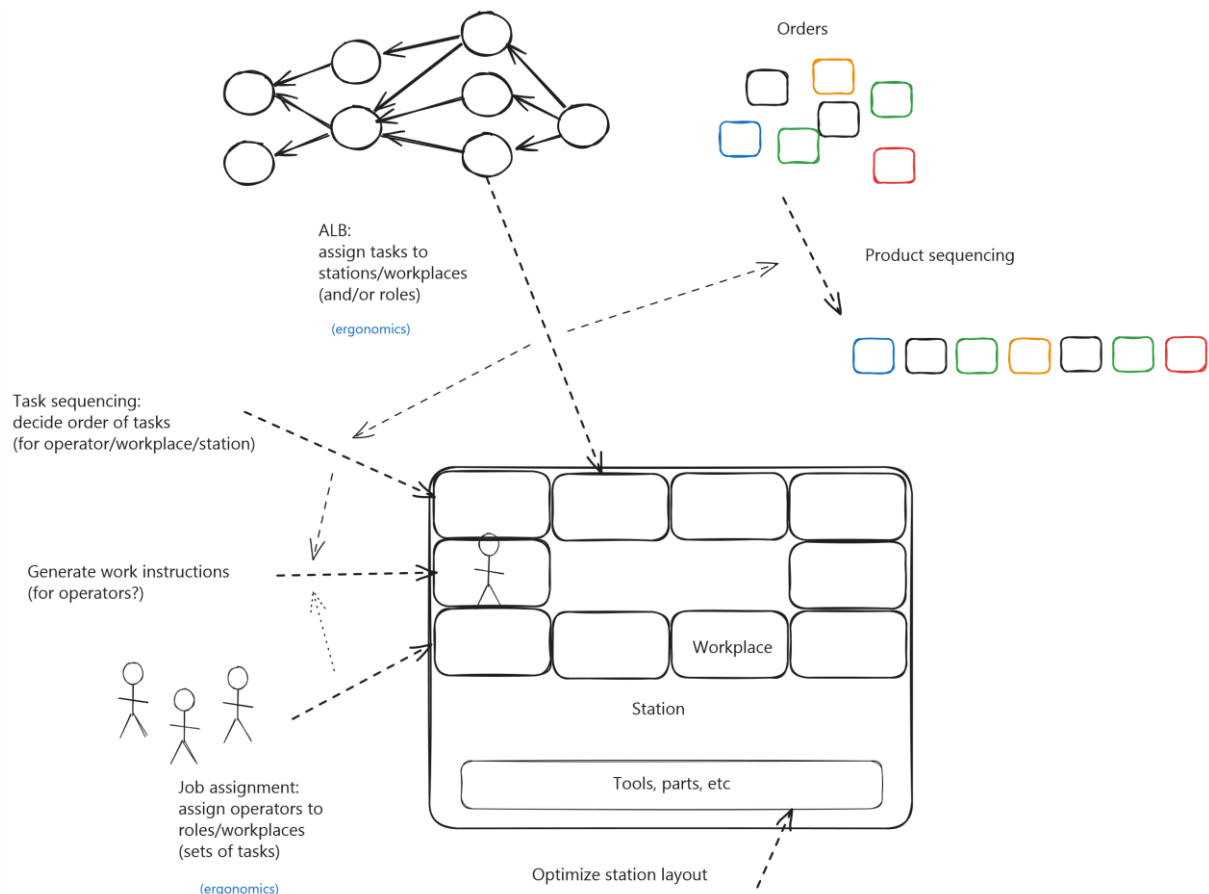
Mixed model production is characterized by producing many different products or variants of products in the same plant, using the same production equipment. There are many ways of arranging the line(s) to meet the challenges. Examples include:

- Fixed physical line, including a main assembly line which is supported by feeder lines in a fishbone structure. This is suitable for large products, involving heavy and fixed production machines, and hand tools. Subclasses of this include
- Driven line, where products move continuously across stations at a synchronous takt time (clock cycle). The use case is truck manufacturing.
- Stop-and-go, where products are carried across stations on independent carts, or roll by themselves if they have wheels. Although there is an average takt time, this allows for some flexibility and limited buffering along the line. Use cases include bus manufacturing and construction equipment (e.g., wheel loaders). The model is also suitable for vehicles' engine and powertrain manufacturing.
- The physical lines are arranged for each kind of product. This requires that stations are movable and can be rearranged into multiple parallel lines. This model is suitable for lighter products where no heavy equipment is needed. The principle is typically stop-and-go, including buffering between the stations. Our use case is textile manufacturing (e.g., clothing).

The different kinds of planning needed in mixed model production with high variability are depicted in the figure below.

- Product sequencing is to plan how to mix the orders on the line so that products that require lots of work on the same station(s) are spread out in time.
- Assembly line balancing (ALB) is to plan for each product which operations should be carried out on which stations.
- Task sequencing is for planning the order of tasks and operations inside each station and sometimes across stations.
- Generation of work instructions includes making detailed plans and descriptions for each operator to carry out the tasks inside a station.
- Teaching work instructions includes the process of workers learning how to perform the operator tasks correctly and efficiently.
- Job assignment is about assigning workers (people) to the operator roles on stations for the upcoming shifts.

All planning is human-centric with objectives of natural workflows, ergonomics, and variation of tasks.



3 Proof-of-Concepts

This section presents the Proof-of-Concepts including the simulation performed and visualized related to data from the real-world use-cases.

3.1 Product sequencing decision support

This proof-of-concept (PoC) tool provides decision support for engineers and assembly line managers for advanced planning and late-stage re-planning of product sequences on mixed-model production lines with a configured nominal takt time. Such lines are synchronous across all stations on the line (or per segment if the line is segmented). In vehicle manufacturing, primary cases are “*driven lines*” and “*stop-and-go lines*”. In driven lines, products move slowly on a belt, continuously traversing the stations. In stop-and-go lines, products stand still at the stations and simultaneously (at each takt) move stepwise to the next station. In both cases, there are methods to allow some limited slack between stations (explained below). Product sequences are planned long in advance for efficient logistics, but disturbances like missing parts often require late-stage replanning and jump-outs from the sequence just before a product was planned for production to start.

The background for the challenge is that different models use different time durations in stations, and in vehicle assembly it is allowed for some chassis to overuse time at some stations, provided that some prior or following chassis use less time. In stop and go lines, some buffering can be allowed between stations (but it is costly in space since products are big). In driven lines, overused time ripples through the sequence until there are some products using less than the nominal takt time. There is a maximum drift time per station, which in case of a driven line is related to the maximum drift distance in metres (e.g., depending on equipment mobility limits). The maximum drift is double ended, i.e., allows both earlier start and later end. In this driven mixed-model situation, the product mix, as defined by the sequence, is of utmost importance for feasibility, risk assessment, and production efficiency.

Traditionally, the allowed sequences are set by static rules on vehicle properties (e.g., number of axles, electric or diesel, etc.). This tool can be instantiated either for rule-based operation, or time-based operation where information about the nominal time needed for tasks and operations serves as a basis for sequencing decisions. An objective for this tool is not to replace existing tools and workflows, but to complement them, for example, in workflows for late changes that cannot be reverted to the long-term pre-planning workflow.

This tool is designed based on requirements from two different use cases. One use case is a driven line for truck manufacturing (Volvo GTO, i.e., Trucks), where focus for the PoC is to support late-stage replanning of the product sequence, including jump-outs and jump-ins, particularly in response to critical conditions such as blocked orders, e.g., due to missing critical parts. The other use case is a stop-and-go line for construction-machine manufacturing (Volvo CE). The tool is configurable, so it should be applicable to other similar use cases of vehicle manufacturing. The tool primarily serves production engineers performing sequence planning and line managers who need to adjust the planned sequence quickly and efficiently. In the following, we will explain the tool setup and support separately in each of the abovementioned specific cases: driven line time-based late replanning and stop-and-go rule-based preplanning. The PoC is available online for invited users.

3.1.1 Driven line time-based re-planning

A key feature is the Support for Rescheduling Blocked Orders, which enables line managers to make adaptations to handle late notices on missing preconditions, such as disruptions in the supply of critical parts. The vision is to provide a flexible and interactive workflow that allows both automated and manual interventions. The tool computes and visualizes the time drift per vehicle per station (in % of takt time) occurring across the line as a response to the proposed changes.

The workflow includes two main scenarios and a combination of those by drag and drop in the sequence:

Jump-Out Workflow - when an order is blocked and needs to be removed from the sequence:

- The blocked order is removed from the plan.
- The tool recomputes timing and cumulative drift per station and per chassis.
- Sequences are visualized to help users understand their impact.
- Users can manually adjust the sequence, for example, using drag-and-drop functionality.
- Once a desirable solution is identified, it is accepted and implemented.

Jump-In Workflow - when one or several jumped out orders shall be inserted into the sequence:

- Each order can be inserted (tried) manually by drag-and-drop, or the optimization algorithm can be invoked on a set of jumped-out orders, obtaining proposals for feasible solutions where the jumped-in orders are merged into the otherwise unchanged sequence.
- Visualization tools help users examine the proposed solutions.
- Manual modifications can be made to fine-tune the sequence.
- The most suitable solution is then accepted.

Combined jump-out and jump in

- By drag-and-drop, a vehicle can be moved in the sequence as one operation in the late-replanning workflow. For example, if some preconditions for a chassis are not yet fulfilled, it can be moved back a bit in the sequence. It can also be dragged to a different day of production.
- The optimization algorithm can be invoked, with constraints on how far movements are allowed, obtaining improved proposals.
- Instant visualization helps users examine the proposed solutions before accepting.

This dynamic and interactive approach ensures that production planning remains resilient and adaptable, even in the face of unexpected disruptions.

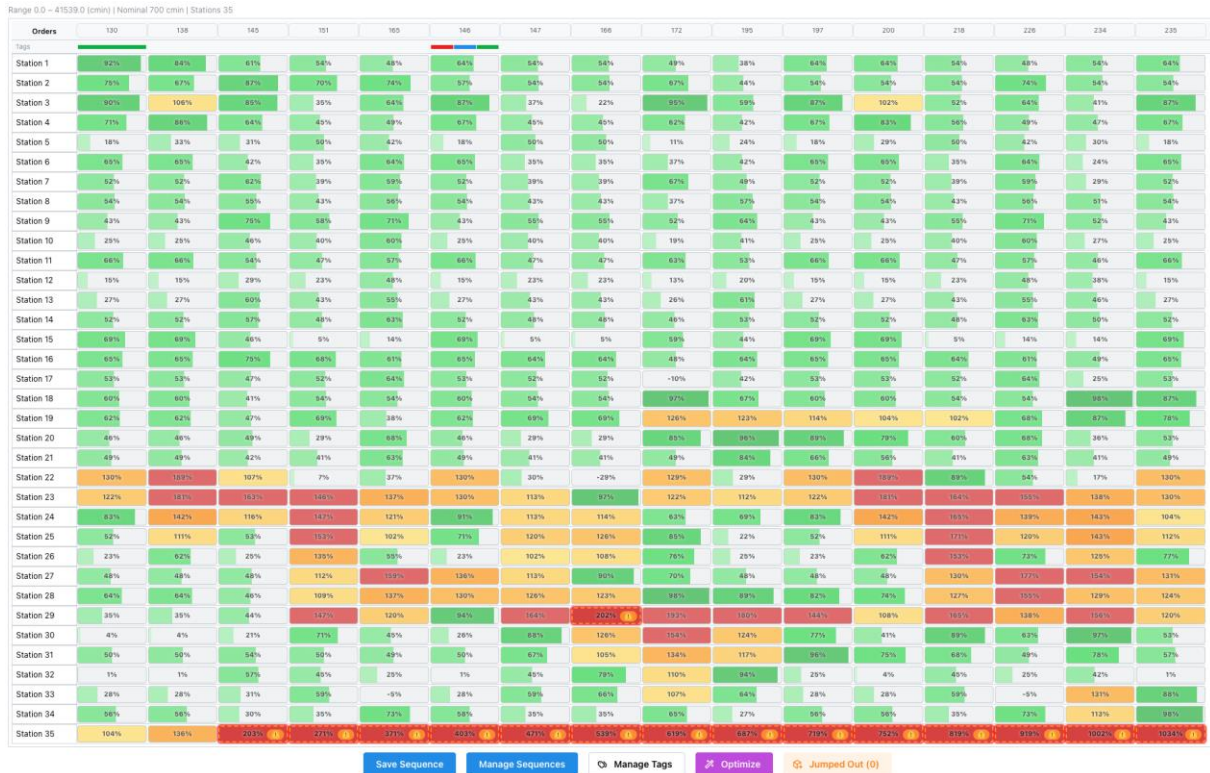
An important step in creating the tool was to determine how to visualize the sequence of orders. We herein describe only one kind of visualization, while other ways can be explored in the online PoC. An objective is to have a stable visualization of the line that does not require complex eye movements to follow products through the line, while still

guiding and helping the user to see where and what potential issues occur in the sequence.

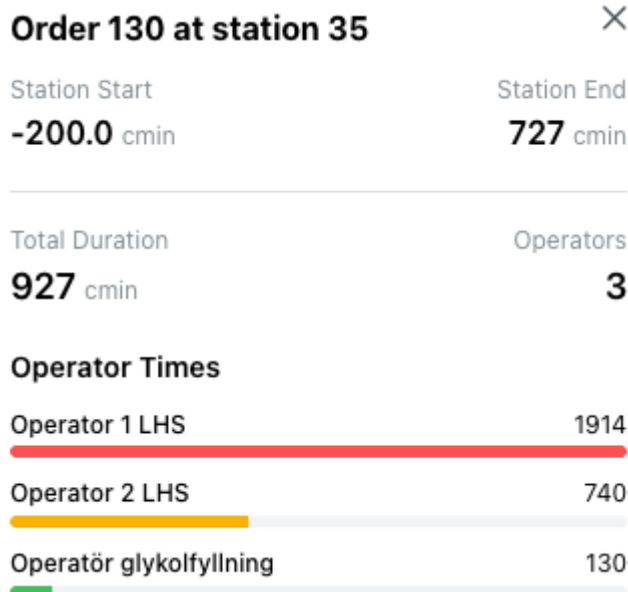
Proof of concept and simulation

In the visualized example, station numbers are shown on the y-axis and chassis numbers on the x-axis. The takt time is 700 cmin (centi-minutes), the earliest allowed start is –200 cmin (max negative drift into previous station, even if other neighboring conditions would allow for more negative drift), and the expected finalization time per station is shown in percent of the takt time (there is an option for actual time instead of %). A number less than 100% on a selected station means that the product there finishes before the takt ends. A number over 100% means that the product will finish a bit after the takt ends, resulting in a drift, causing a delayed start for this product on the next station AND a delayed start for the next product on this station. In addition to percentage, the per-vehicle and per-station conditions are shown by heat colors, using green, yellow, and red (possibly nuanced) to indicate the situation. A product that can be handled entirely before the takt end is indicated in green. Products that moderately overrun the takt are shown as yellow (in stations where this occurs), while red is used to show a heavily overrunning process. An impossible to resolve violation is red with additional dashed border and an icon indicating this. The threshold per color can be configured. The result of this coloring scheme is a heatmap on the timeline where a user can easily spot where a drift starts to happen and follow how it ripples through and changes until resolved or until impossibility.

For simulations, real anonymized production data is used. An example of a sequence that is not yet organized and therefore involves several colors is shown in the screen dump below. Situations that may require some action or changes are clearly shown.



The line manager can click on any box to see detailed information on operators and task times (shown below). In the shown setting, the total duration per station equals the average of the operator times (i.e., they help each other when finished with their primary assigned task).

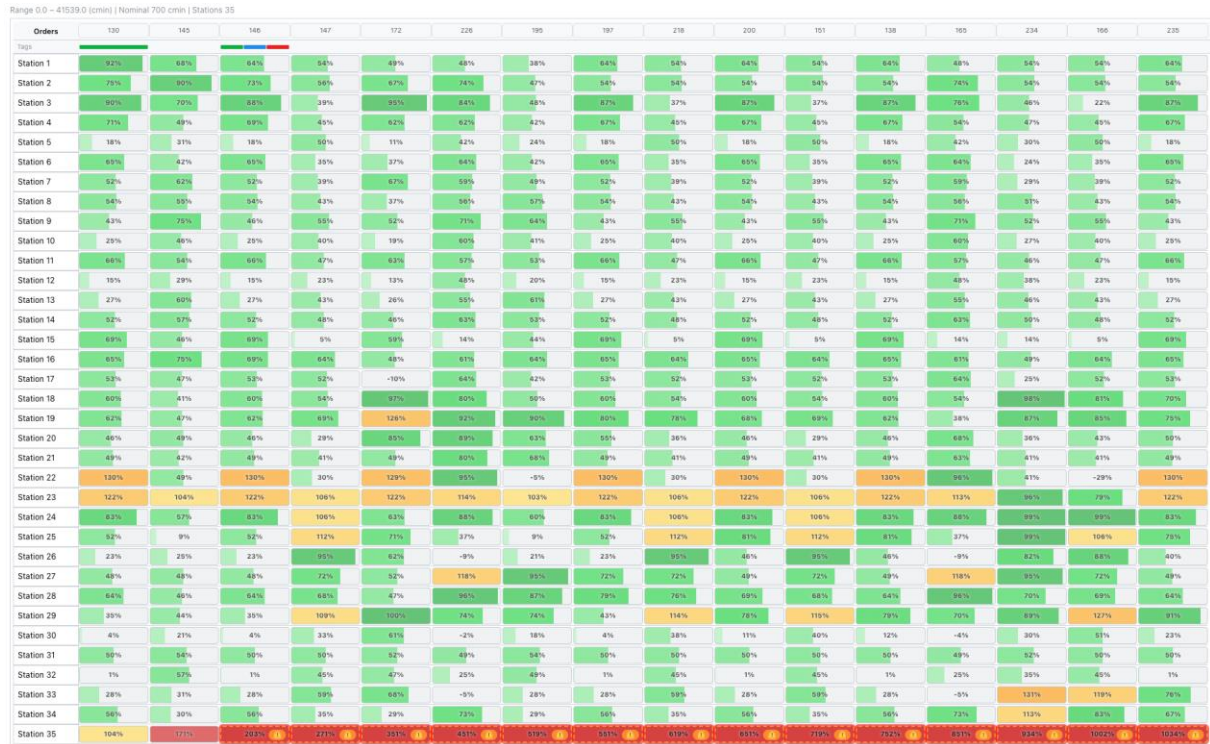


It would be possible to handle some situations by changing the preconditions, e.g., adding another operator for that chassis at that station (a runner operator). Or try another sequence. The visualization should deliberately be a bit conservative. It is

better to show red a bit too often than to miss a situation that needs caution or attention. Since orders are represented vertically, a user can follow a specific order through the whole sequence and easily modify the sequence by dragging and dropping an order to another place in the sequence and directly see the result of such a modification.

In the sequence shown above, the user can spot that there is something wrong at station 35, where even the first slot is not green, which continues at the station until there are severe violations. To further understand what is happening at the station, we can click on the process of order 130 at station 35 (see figure above). There, the total station duration is 927 cmin (i.e., centi-minutes), mostly caused by operator 1 needing 1914 cmin, while the takt time is 700. By checking the process of other products, we can confirm that this is a recurring issue at the station and not for that specific product. This may be because the input data is not correct, for example, that operator 1 is really two people where their sum of time is recorded rather than the clock duration (which should be half). In general, the user can solve problems by adding operators permanently or temporarily (so called runners), then correcting the data accordingly.

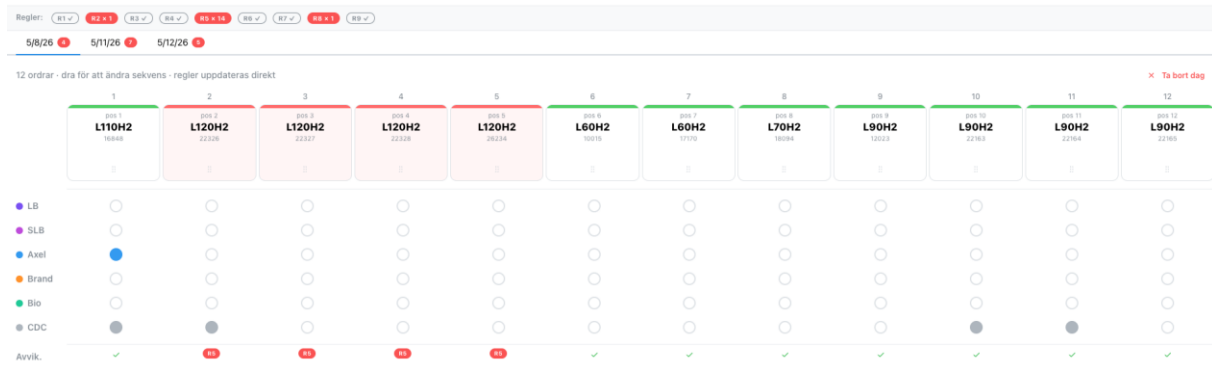
In the above example, all other critical (i.e., red-colored) conditions can be fixed by changing the sequence, either by drag and drop or by applying the “optimize” algorithm. There is typically no single optimal solution, but several alternatives that are sufficiently good. An example of a sequence that is as good as possible regarding the drift of the worst-case slots is shown below. In this sequence, all the worst-case slots of 130% or higher are self-inflicted, meaning that they all start at the earliest allowed negative drift –200 cmin (which one can see since the slot above and to the left are all green with low percentage). Note that all yellow drifts are acceptable and that they ripple horizontally to the right and vertically downwards without getting worse and in some cases until becoming green (of course not yet seen in the rightmost column and in the bottom line, as explained above). The limits on drift are related to the station design. Should more drift (earlier start or later end) be allowed, more of the time gaps between stations could be utilized.



Lastly, the user can save satisfactory sequences, and re-load previously saved sequences for further modification before committing. The primary need in the use case is support for late-stage replanning, which typically focuses on jumping in and out, or moving one or a few vehicles in a preset sequence. The computational problem in visualizing resulting sequences is linear, so the results can effectively be shown directly. The tool has an algorithm that can jump in a set of jumped-out products in an existing sequence. The algorithm can also be used to plan sequences by taking an original sequence as input (e.g., a set of items) and re-sequence either freely or within constrained limits (i.e., any item may move max x steps). The proposals are presented to the engineer to allow for selections, manual modifications, and acceptance.

3.1.2 Stop-and-go line with rule-based planning

The use case shown here is a stop-and-go assembly line in wheel loader manufacturing where the sequencing decisions are rule-based. The shown visualization includes the manufacturing sequence on the x-axis and some key properties on the Y-axis (in this example 6 properties). The rules are shown horizontally above the diagram (in this case 9 rules) and can be hovered to get the details. An example of raw data before any attempt to make a feasible sequence is shown below.



The products in the sequence are color marked to indicate whether the position in the sequence is feasible. The red color indicates over-drift, while the green fits entirely in the takt. In the raw and not yet sequenced example, the placement of chassis 22326, 22327, 22328, and 26234 in positions 2 to 5 in the sequence is unfeasible. By drag-and-drop or by pressing the *optimize* button, a feasible proposal is obtained.



Besides using a rule-based model instead of a time-based one, much of the functionality is like the driven case where the operator also can click an item to get more details (e.g., the reason why the colour is red), manually drag and drop items in the sequence, study the effects and eventually commit a desirable solution. In this case, sequencing is done in advance for many days, and last-minute re-planning can be done when needed. However, instead of jump-out and jump-in, the desirable mode of operation is to move a product to a future place in the sequence, for example later in the day or to one of the following day(s).

3.1.3 Optimization algorithm

We have three automatic solvers that can be invoked from the user interface: one (Solver A) for the Volvo GTO use-case, and two (Solver B and C) for the Volvo CE use-case.

Solver A and B are both based on a custom construction heuristic followed by refinement by a metaheuristic. Solver C is entirely based on constraint programming (CP).

In the GTO case, the main use case is not product sequencing per se (i.e. producing an initial solution) but the focus is rather on late modifications of an existing sequence, i.e. jump-out/in. This creates trade-offs between minimizing disruptions to the existing sequence (at the expense of a sub-optimal final sequence) and maximizing solution

quality (at the expense of greater disruptions to the existing sequence). Therefore, the solver must present the user with multiple reasonable options to choose from. Solver A suggests several different solutions. Some of these differ in which heuristics are used, and some differ in the weights for different optimization objectives.

The construction heuristic actually uses multiple heuristics internally and chooses the best solution.

In the CE case, the main use case is product sequencing per se, and modifications are not considered. This greatly simplifies the problem, since there are no trade-offs to consider. In addition, the constraints are rule-based rather than time-based. This means a solution is either feasible or not, and a ranking of different feasible solutions is not defined. Hence, the notion of optimization (beyond feasibility) does not apply.

Solver C is complete and will find a feasible solution if one exists. Solver B, on the other hand, uses heuristics and may therefore fail to find a feasible solution. In practice, it often finds a feasible solution and does so in approximately 1/1000 of the time compared to Solver C (i.e. a few milliseconds rather than a few seconds).

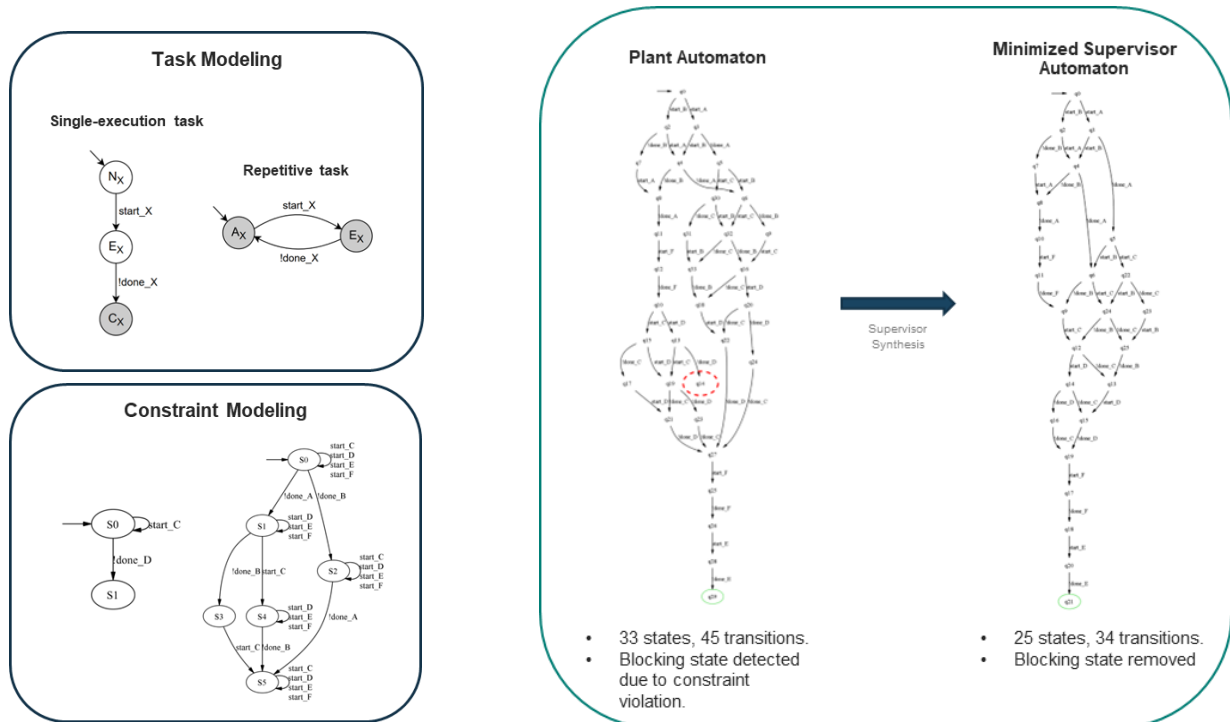
3.2 Assembly Line Rebalancing and Task Sequencing

3.2.1 Task Sequencing PoC

This tool supports engineers in the identification of minimally restrictive assembly task sequences by providing a formal procedure to model tasks, constraints, and admissible execution orders. The assembly tasks are modeled as finite automata with controllable start events and uncontrollable completion events, allowing concurrent task execution and the representation of process-related dependencies. The task automata are composed to form the plant, which captures all possible task executions.

Precedence relations and process constraints are modeled as specification automata and synchronized with the plant. Supervisor synthesis based on Supervisory Control Theory is then applied to remove blocking and uncontrollable behaviors that could prevent completion of the assembly process. The resulting supervisor enforces all constraints while disabling only those task executions that would inevitably lead to deadlocks or violation of constraints, thus guaranteeing non-blocking and minimally restrictive behavior.

From the supervised automaton, all task sequences leading from the initial state to a marked completion state can be extracted. These sequences constitute the set of minimally restrictive feasible assembly sequences, ensuring that any selected sequence allows completion of the process while preserving the maximum admissible flexibility. This set provides a structured basis for optimization, exception handling, and the generation of adaptive work instructions for manual assembly operations.



The modeling and supervisor synthesis is carried out in the supervisory control tool Supremica. For details, please see the paper *Identification of Minimally Restrictive*

Assembly Sequences Using Supervisory Control Theory, by Vinetti and Fabian, presented at the 2025 IEEE 8th International Conference on Industrial Cyber-Physical Systems (ICPS). Ongoing investigations indicate that searching for the time-optimal sequence fulfilling all constraints is faster done over the supervisor, compared to over the non-supervised model.

3.2.2 Assembly Line rebalancing PoC

Assembly lines often need to be adapted when demand, product mix, task times or available resources change. Rebuilding a line from scratch, however, can disrupt equipment, material supply, workstation layouts, and established operator routines. The Assembly Line Rebalancing Problem (ALRBP) proof of concept supports production engineers in revising the current task allocation to meet a new takt time while limiting unnecessary changes to the line already in use.

The optimizer starts from the existing allocation and considers task times, precedence relations, ergonomic information and the areas from which tasks can be performed. It supports both single- and multi-worker stations. In a multi-worker station, each operator works within a dedicated area, whereas a single worker can access the whole station. The proposed allocation is assessed in terms of workload balance, ergonomic balance and preservation of the current configuration. The complete mathematical formulation is described in Vinetti et al. (2026b).

To preserve the existing configuration selectively, the PoC uses a relevance-guided approach. It prioritizes task groupings whose reassignment is expected to have a greater operational impact, while leaving the optimizer more flexibility for the remaining tasks. A parameter K defines the share of highest-ranked tasks to which this preservation mechanism is applied. Further details on the relevance measure and its integration into the rebalancing model are provided in Vinetti et al. (2026a).

The evaluation covered 21 benchmark instances up to 94 tasks. Table 1 summarizes the results obtained with $K = 30\%$; the full experimental setup and instance-generation procedure are reported in Vinetti et al. (2026a).

Table 1. Overview of the computational results for $K = 30\%$. Source: Vinetti et al. (2026a).

Size class	Instances	Runtime (s)	MSF _p	Ergonomic imbalance (%)	Workload imbalance (%)	Optimal
7–30 tasks	10	0.36	0.80	19.19	12.24	10
31–60 tasks	5	38.13	0.90	15.53	2.87	5
61–94 tasks	6	15.82	0.91	15.24	6.36	4

All 15 instances with up to 60 tasks and four of the six larger instances were solved to optimality within the 90-minute time limit. The preservation score (MSF_p) remained between 0.80 and 0.91, indicating that a large share of the prioritized task relationships was retained while the line was adapted.

The station configuration also affects the ergonomic result. Figure 1 compares the ergonomic imbalance obtained with single-worker and multi-worker stations for the minimum-station configuration of each benchmark instance. In most cases, the single-worker setting achieves a lower imbalance. The comparison highlights how the work-

area restrictions associated with multi-worker stations can influence the ergonomic balance achieved by the rebalancing model (2026a).

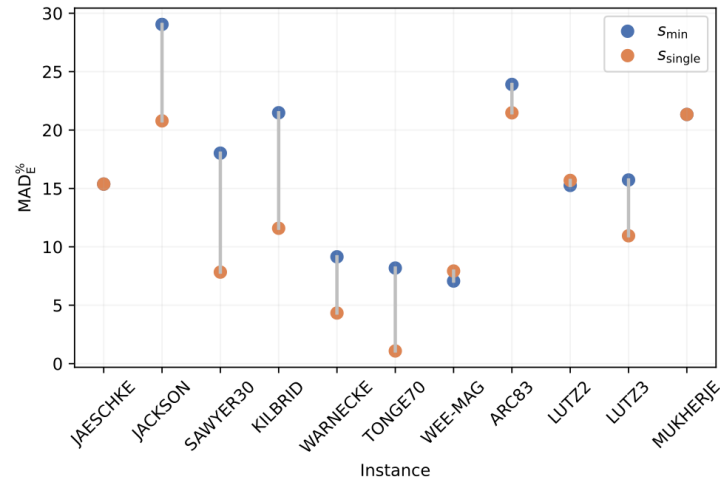


Figure 1. Ergonomic imbalance for single-worker and multi-worker station configurations. Source: Vinetti et al. (2026a).

Figure 2 illustrates the effect of K on the three evaluation criteria. Workload and ergonomic imbalance change only slightly across the tested values, while the preservation score decreases as a larger share of tasks is included. In these experiments, K therefore provides a practical way to adjust how broadly configuration preservation is applied. The complete sensitivity analysis is reported in Vinetti et al. (2026a).

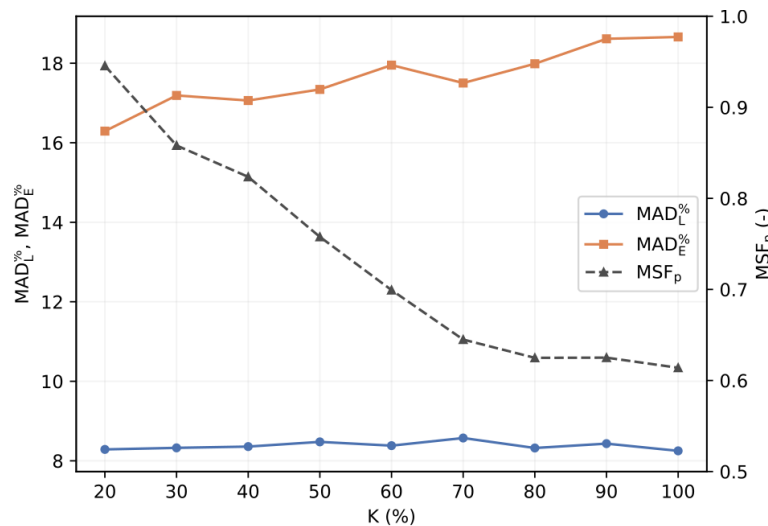


Figure 2. Sensitivity of workload balance, ergonomic balance and configuration preservation to K. Source: Vinetti et al. (2026a).

The PoC returns a revised task-to-station allocation that can support subsequent decisions on workstation layout, task sequencing and worker assignment. The current implementation and benchmark instances are available in the project repository:

<https://github.com/Chalmers-Control-Automation-Mechatronics/ALRP>

Future work will focus on validation with industrial data, computational scaling and an interactive interface for comparing alternative rebalancing scenarios.

3.3 Worker assignment - decision support for employee-job matching

This is a decision support tool designed to assist team leaders in final assembly stations, where the daily assignment of employees to workstations is inherently complex. Leaders must balance multiple conflicting objectives, such as operator competence, individual job preferences, and strict ergonomic constraints, while simultaneously managing real-time disturbances like unexpected absences or tardiness. When performed manually under time pressure, this redistribution often leads to sub-optimal assignments that can increase the risk of physical injury, negatively affect work quality, and foster a sense of unfairness among the operators.

To address these challenges, the solution employs an Integer Linear Programming (ILP) model to automate the assignment process. The model relies on four critical data inputs: a competence matrix detailing trained operations, a preference matrix of employees' favored jobs, a history of past assignments to track rotation, and ergonomic scores for individual tasks. Its weighted sum objective function balances these inputs to maximize total employee preference and ensure fair job rotation, while minimizing the need for external staffing. The objective function has since been refined to reduce configuration complexity by eliminating two weighting parameters, and the influence of the remaining weights on assignment outcomes has been analyzed to produce practical heuristics that let decision makers tune the model toward targeted operational outcomes rather than by trial and error.

The current scope extends the tool from a stand-alone solver into a connected system with a constant knowledge update pipeline. The system is backed by a Redis data store that holds the model's data inputs, operator, and operation data as live state rather than as manually curated files, and exposes them through a defined interface so that the tool is ready for connectivity to real factory systems. The enabling change is that assignment history is now collected as work is performed: every assignment the tool produces, together with the actual outcome on the line, is written back into the store. This means the rotation and fairness measures used by the model reflect what really happened rather than what was planned, and that an operator's record builds up automatically from the day they arrive. For a newly hired or newly transferred operator, the tasks they perform are recorded as they go, so their history, and in turn their observed exposure to ergonomically demanding work, accumulates without any manual bookkeeping. Because the model consumes this continuously updated state, it stays valid as products, station layouts, and workforce composition change, and the data maintenance burden that would otherwise limit adoption is removed.

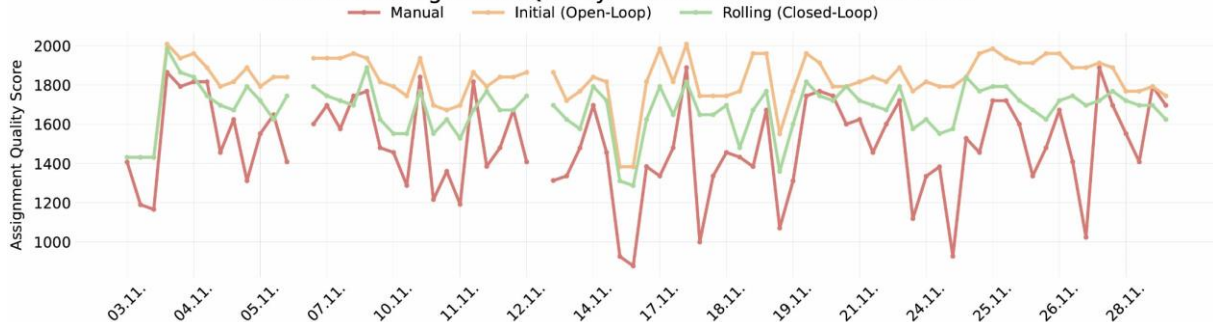
The tool provides real-time support, allowing team leaders to dynamically adapt assignments at the moment a disruption occurs without compromising safety or

efficiency. The method has been validated on operational datasets from active manufacturing stations at Volvo Trucks Tuve and Volvo Construction Equipment in Sweden, where it consistently and rapidly generated high-quality assignments. Benchmarked against historical, human-generated schedules from both plants, the ILP model outperformed manual scheduling, yielding higher assignment quality, stricter adherence to ergonomic rotation, and better preference satisfaction.

Volvo GTO Station Daily Assignments in November 2025 (4 Passes / Day)



Volvo GTO Assignment Quality Evaluation for November 2025



Ergonomic Assgner
1. Data
2. Roster
3. Today
CE - 16 operators - 12 jobs

Choose how to get your competences and preferences in.

Load a matrix file
A JSON file describing the station's jobs, ergonomic scores, employees, competences and preferences.
[Choose file...](#)
...or drag one onto this window.

Set up from scratch
Define the jobs, then add employees with their competences and preferences. You can export the result as a matrix file to reuse later.
[Start setup](#)

Try an example
Real anonymised rosters, useful for seeing what the tool does.
[VCE \(16 operators, 12 jobs\)](#)
[GTO \(11 operators, 8 jobs\)](#)

Assignment history (optional)
History is what makes the fairness term work: it stops the same person getting the same job day after day. Both the day-based and pass-based file formats are accepted.
[Load history...](#)

Loaded the VCE (16 operators, 12 Jobs) example. [dismiss](#) solver: same origin

Ergonomic Assgner
1. Data
2. Roster
3. Today
CE - 16 operators - 12 jobs

Jobs
Each job has an ergonomic score: higher means physically harder.

01	ergo	1	x
010	ergo	3	x
011	ergo	4	x
012	ergo	3	x
02	ergo	2	x
03	ergo	2	x
04	ergo	3	x
05	ergo	3	x
06	ergo	4	x
07	ergo	3	x
08	ergo	3	x
09	ergo	2	x

New job id, e.g. 01 [Add job](#)

Employees

A Operator x
Competences
✓ O1 ✓ O10 ✓ O11 ✓ O12 ✓ O2 ✓ O3 ✓ O4 ✓ O5 ✓ O6 ✓ O7 O8 O9
Preferences (most wanted first — order matters)
1. 04 ☐ ☐ x
2. 012 ☐ ☐ x
Add preference: choose...

B Operator x
Competences
✓ O1 ✓ O10 ✓ O11 ✓ O12 ✓ O2 ✓ O3 ✓ O4 ✓ O5 ✓ O6 ✓ O7 ✓ O8 ✓ O9
Preferences (most wanted first — order matters)
1. 08 ☐ ☐ x
2. 012 ☐ ☐ x
Add preference: choose...

C Operator x
Competences
✓ O1 ✓ O10 ✓ O11 ✓ O12 ✓ O2 ✓ O3 ✓ O4 ✓ O5 ✓ O6 ✓ O7 ✓ O8 O9
Preferences (most wanted first — order matters)
1. 08 ☐ ☐ x
2. 011 ☐ ☐ x
Add preference: choose...

D Operator x
Competences

Station id: CE [Export matrix JSON](#) [Continue to Today](#)

Loaded the VCE (16 operators, 12 Jobs) example. [dismiss](#) solver: same origin

Ergonomic Assgner | 1. Data 2. Roster 3. Today

CE - 16 operators - 12 jobs

Today's operators
Mark anyone not available for normal work, and force an assignment where you have to.

A TL Available
Force job: —

B TL Available
Force job: —

C TL Available
Force job: —

D TL Available
Force job: —

E TL Available
Force job: —

F TL Available
Force job: —

G TL Available
Force job: —

H TL Available
Force job: —

I TL Available
Force job: —

J TL Available
Force job: —

Weights
Balanced Preference-first Fairness-first Manual
Default trade-off between preferences, fairness and external operators.

offset 2000
 ω preference 1
 α leader 192
 β external 384
 τ history days 8
 γ repetition 24

Calculate assignments

Assignments

Operator	Job
A	O4
B	O12
C	O8
D	O1
E	O10
F	O9
G	O5
H	O7
I	O6
J	O11
L	O2
M	O3

L: loaned E: absent T: training S: supervision TL: team leader

Metrics

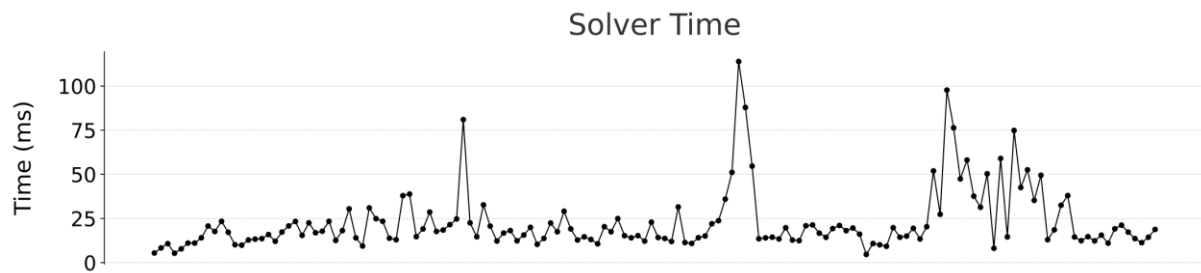
Total score 2099
Preference reward 99
Leader penalty 0
External penalty 0
History penalty 0
Solver time 100 ms

Needs external operator
None — all jobs covered

Free internal operators
• K
• N
• O
• P

No history loaded — the fairness term has nothing to work with.

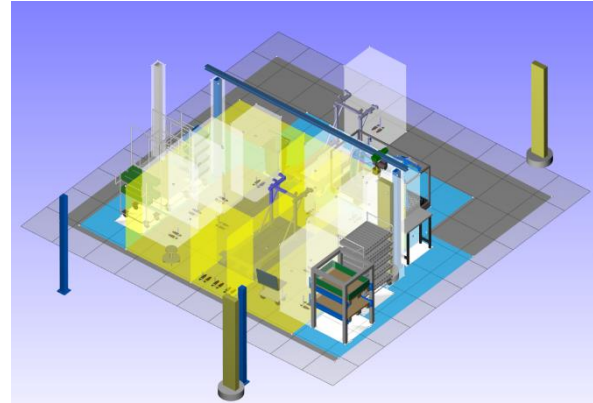
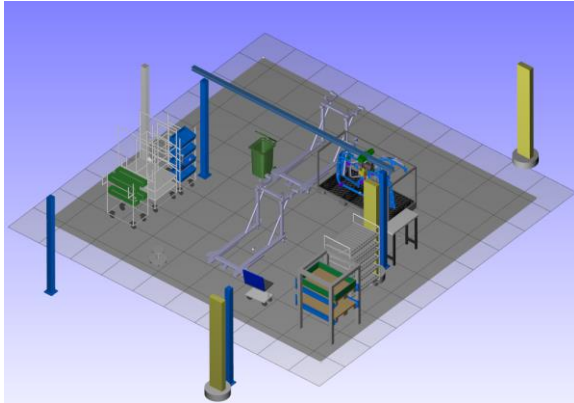
solver: same origin



3.4 Workstation layout planning and task balancing

This software tool supports engineers in workstation design and the allocation of tasks between workers. Today, these activities are typically performed separately even though they are intimately connected. The impact of the workstation design is not considered while sequencing and balancing tasks within the workstation or at best, only crudely estimated. Conversely, the workstation design is often performed manually without decision support tools that take the work sequence into account and often without 3D visualization.

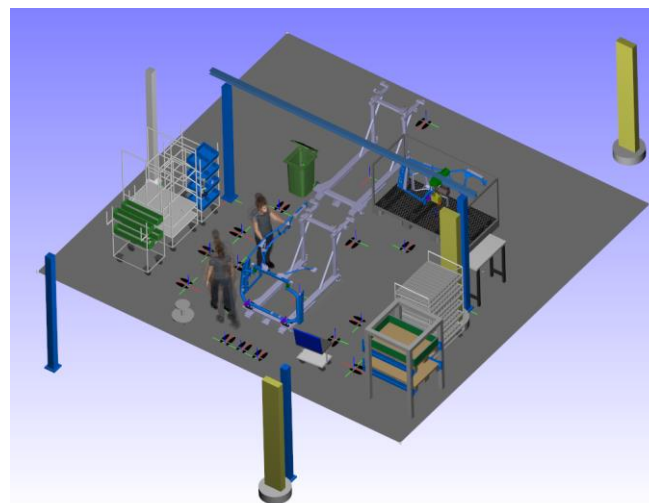
The software enables 3D visualization of the factory layout and the various spatial constraints instead of the 2D drawings that are typically used today. The constraints that can be represented include space claims, shared work areas, and designated areas for selected resources. The user gets real-time feedback concerning the validity of the current placement of resources in the workstation based on the modelled constraints.



3D workstation layout without and with representation of constraints

The software provides optimization algorithms based on a mix of branch-and-bound enumeration and local search heuristics. These algorithms can automatically suggest layouts such that the walking distance between the resources is minimized, and the constraints are respected. The user can iteratively decide to fix the position of certain resources and rerun the optimization algorithm until they are satisfied with the result. Each layout suggestion can also be simulated using the state-of-the-art DHM simulation model IPS IMMA to visualize the work sequence and evaluate its ergonomics based on different criteria.

For stations with multiple workers, the balancing of tasks needs to consider the ergonomic workload for each operator as well as the target cycle time of the station. To improve the time estimation of the work sequence during optimization, a walking simulation model is used. This provides the optimization algorithm with information about the impact of the workstation layout and potential collisions between workers during their parallel work.



Simulation of operators working in parallel

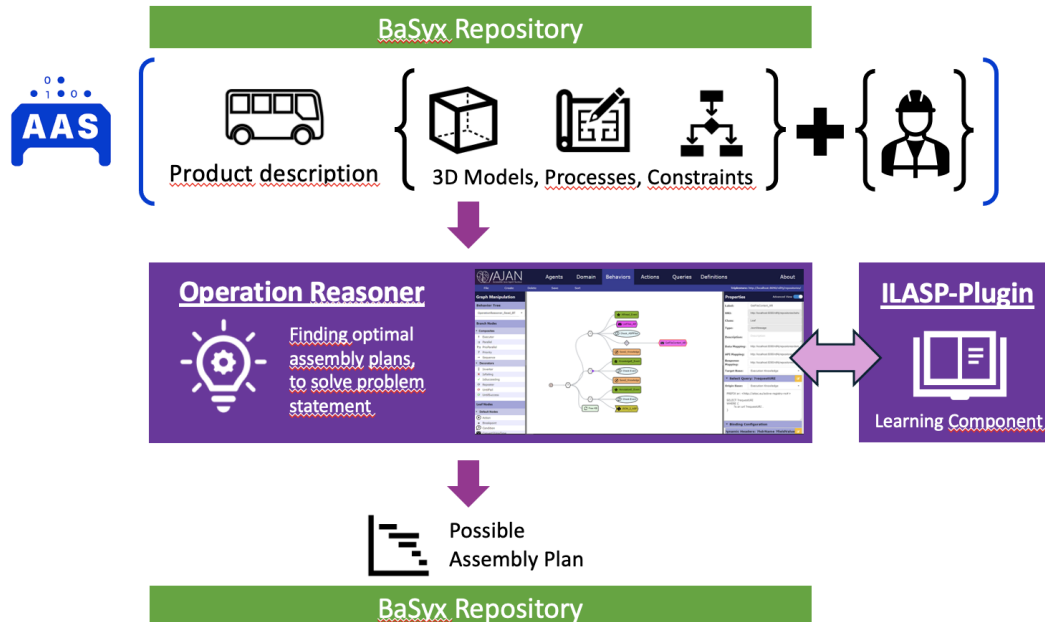
3.5 Task sequencing and worker instructions

The objective is a knowledge management and planning framework for production engineers, line managers, and workers to enable scalable auditing, safe rule evolution, seamless rule transfer, and continuous integration (of both legacy and new knowledge).

The generation of a work plan including assembly instructions is done through an operational reasoner, combining general process rules and specific product annotations which are updated automatically based on real time observations of a assembly plan execution and expert feedback. Additionally, the generated assembly steps are dynamically visualized in an intuitive and user-centric manner based on digital shadow information about the process.

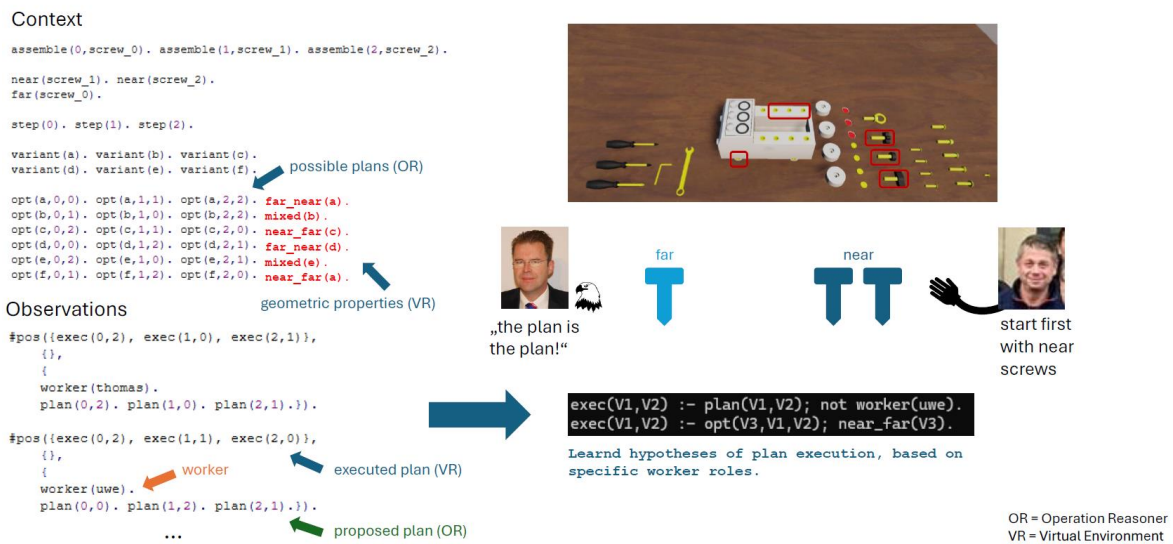
3.5.1 Plan generation and task sequencing

Increasing product individualization makes automated, flexible assembly planning essential, as manual planning becomes complex and error-prone, especially for lot size one. While automated planning exists, integrating it into heterogeneous environments and capturing expert knowledge from workers remains challenging. The Operation Reasoner (OR), an AJAN-based agent using Answer Set Programming (ASP), generates optimal assembly plans from product data, assembly actions, and constraints, converting between standards and formats for visualization and reuse.



To improve interoperability, the integration of Asset Administration Shells (AAS) into the OR is proposed, enabling standardized data exchange and reducing manual integration effort. To do so, we integrated the [py-aas-rdf.py](#) to convert AAS to an RDF representation. So that the Operation Reasoner can interact with AAS data. Additionally, we integrated a LLM-module to automatically generate SPARQL queries based on natural language input, to increase usability. Further, adaptive planning using ILASP allows worker feedback and execution deviations to be learned iteratively, incorporating expert knowledge into future plans and improving robustness during

early production phases. In the following Figure, the usage of said logic-based learning approach is depicted. On the left-hand side, the context with possible (opt) plans, single assemble steps etc. are presented. In red, facts from the VR-based training environment are shown. These are created, depending on geometric properties available in the VR assembly environment, giving the system hints of possible reasons why an expert is adjusting a given plan. The Observations, on the other hand, are recorded plan executions of a worker within the VR-training environment. Given those observations, the context and a previously defined hypothesis space, a logic model is learn representing the behavior of specific workers, (e.g., Uwe or Thomas, or even worker roles like expert or novice). This model (see exec-rules on the right-hand side) is explicitly given and can be integrated into the ASP-based planning ruleset.



Furthermore, it is not only possible to identify workflow preferences; the initial retrieval of nearby screws (as shown in the example) can already provide a clue as to a more optimal placement of work materials. For example, the VR environment can use colliders to identify bottlenecks or additional actions required by the worker that result from a specific sequence of steps for assembling a workpiece. An expert can use their experience to avoid such bottlenecks or additional actions caused by deviations from the plan—deviations that are detected through observations and are explicitly accounted for in the learned rules for plan optimization. See the following learned rules, where we see, that a novice is always executing a given plan, whereas else the expert is adapting the plan to avoid narrow operations.

```
exec(V1,V2) :- plan(V1,V2); worker(novice).
exec(V1,V2) :- opt(V3,V1,V2); not narrow(V3,V1,V2); not worker(novice).
```

For information sourcing and storage, BaSyx is used to share asset administration shell (AAS) based digital models of parts, tools and worker (see Deliverables D2.2 and D2.3). The MontaViz instruction Viewer is used for presentation and optimization.

Geometric Feasibility Check and 3D Collision Validation: A simulation-based 3D validation pipeline was developed to verify automatically generated assembly plans in a deterministic, reproducible, and scalable manner. The primary focus is on geometric feasibility: the system checks whether a theoretically collision-free path exists in SE(3)

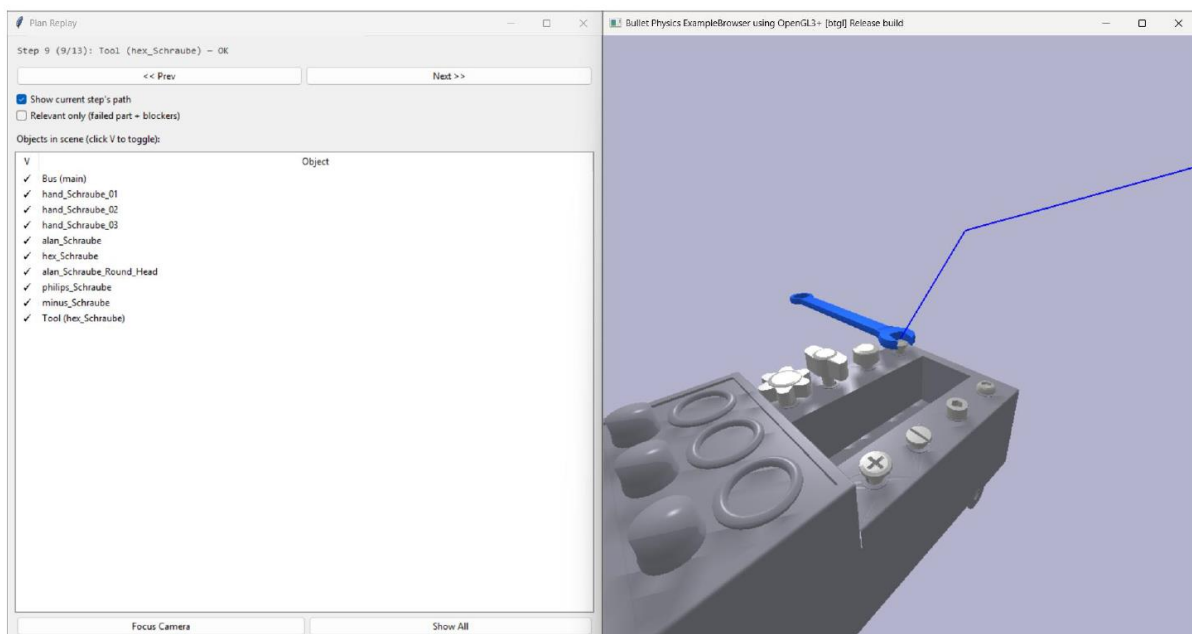
space for each component and its associated assembly tools. Technically, the solution is based on Klamp't for sampling-based path planning using Rapidly Exploring Random Trees (RRT) and PyBullet for exact collision detection. For this purpose, movable components are first decomposed into convex hulls using V-HACD, while static main components remain concave.

Embedded in a modular Flask-REST service, each assembly step of an iterative assembly process undergoes a multi-stage verification process:

- **Geometric Feasibility Check:** Checks in advance, without path planning, whether the component can be placed in its final installation position without collisions.
- **Pass 1 (Quick Check):** Rapidly searches for a collision-free standard feed path using 2,000 RRT iterations.
- **Pass 2 (Detailed Check):** If the first pass fails, initiates an extended retry with 5,000 iterations and heuristics such as reverse planning for tight installation spaces.
- **Tool Validation:** Ensures that the required assembly tool can be positioned and guided to the target object without collisions.

If a sub-step fails, the service provides a detailed error diagnosis (e.g., obstructing objects, penetration depth), which serves both for automated adjustment of upstream planners and for 3D visualization.

Following the develop 3D-Pan Validator is shown:



3.5.2 Visualization and verification of tasks

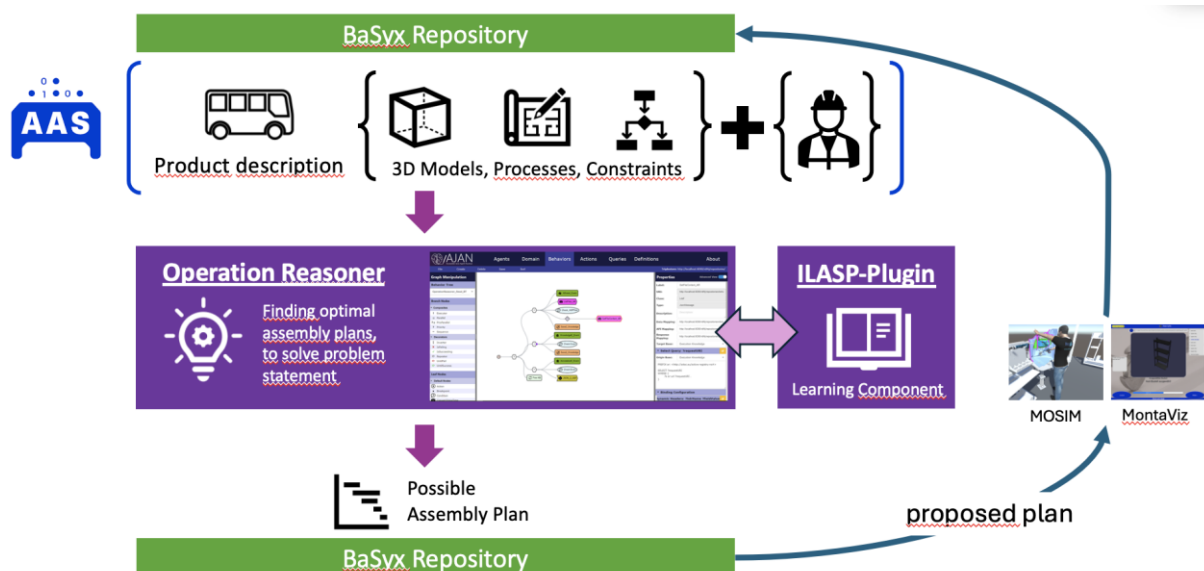
MontaViz is a work instruction viewer which can dynamically generate human-understandable assembly instructions from digital twins, and the Operation Reasoner and allowing real-time adjustments based on the current context of the executed assembly.

To accommodate for the workers' individuality, the MontaViz UI is structured in a widget style. The main widgets of the current version are:

- 3D Viewer
- Minimap which shows the position of necessary tools
- Textual instruction (i.e. "Insert screw x in screw hole y")
- Instruction details
- Task sequence
- Thumbnail of assembly part

The widget approach allows for easy add-ons. Potential add-ons include:

- information hubs (i.e. for system wide announcements)
- smart pdf viewer for manuals and other contextual information



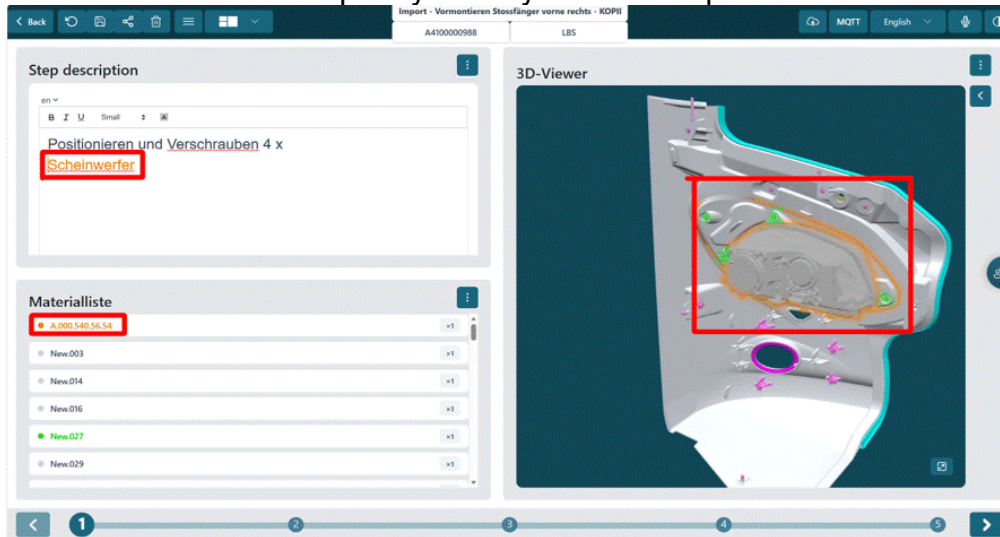
To improve the usability of the MontaViz system, the user interface was re-designed with optimizations based on the ISO 9241-110 (2020) norms. The optimizations can be categorized as follows:

- **Suitability for the task:** The user interface can be adjusted to better suit the task via the aforementioned widget-system. The description of tasks can be adjusted dynamically with an AI-based system, as described in section 3.5.3. User interactions for adjusting Tasks are improved via useful buttons in edit-mode, including delete, duplicate and publishing buttons.



- **Self-descriptiveness:** The descriptiveness of the tasks is improved by color-coding the relevant parts of the instruction step in the 3D-Viewer and text,

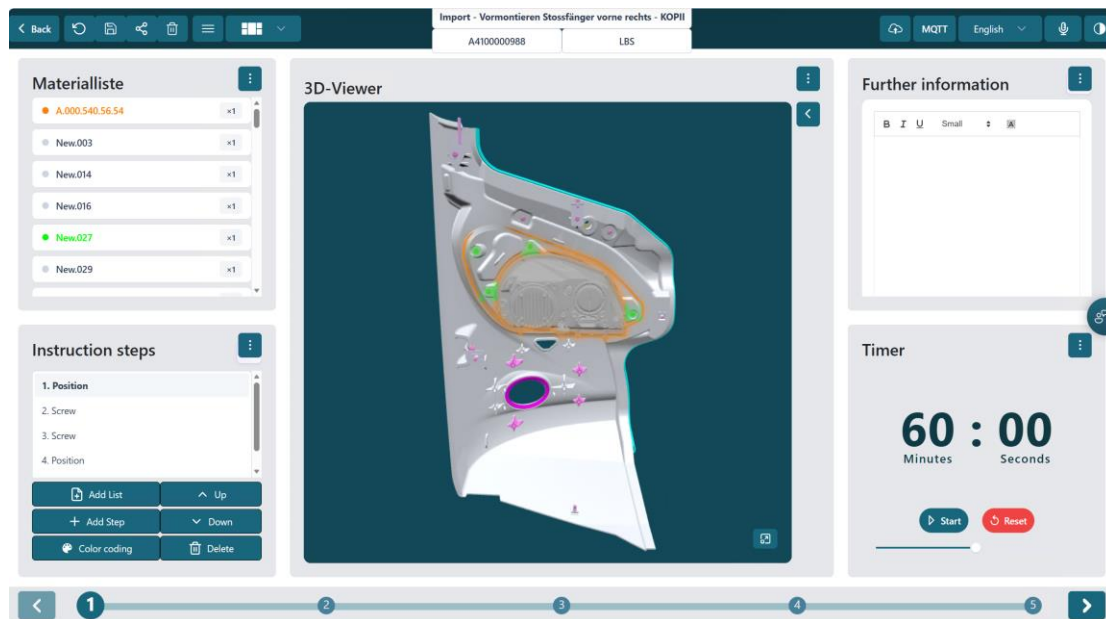
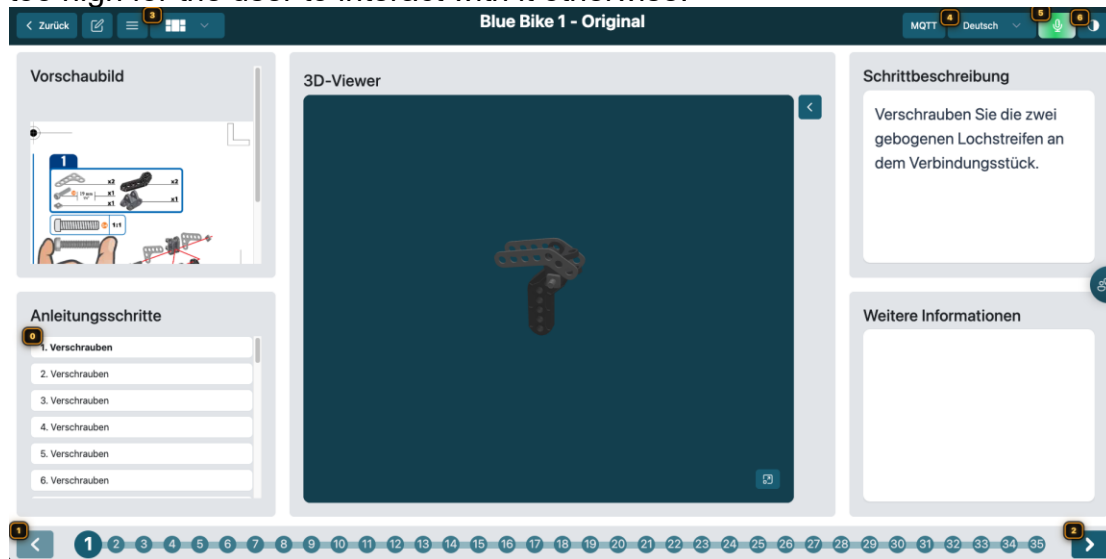
such that the user can quickly identify the correct parts.

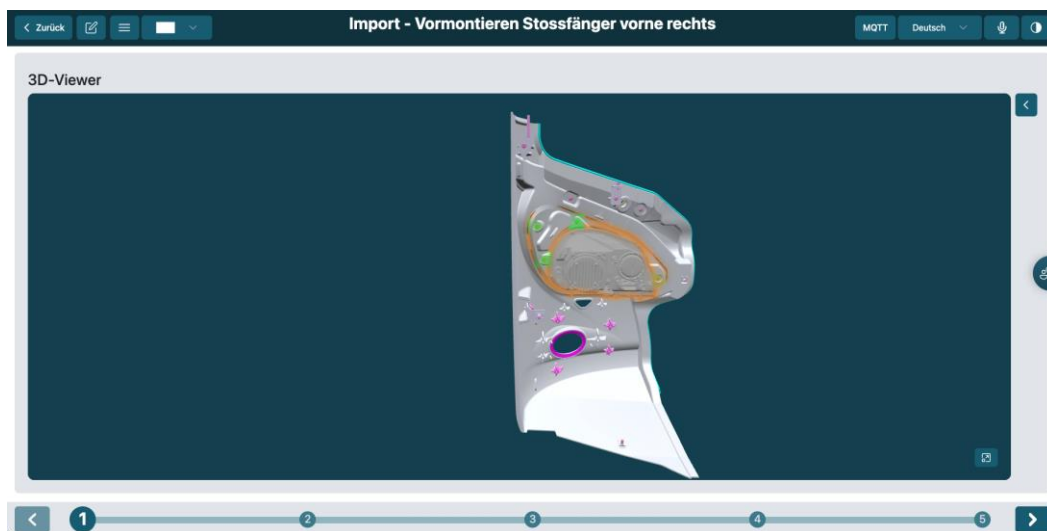
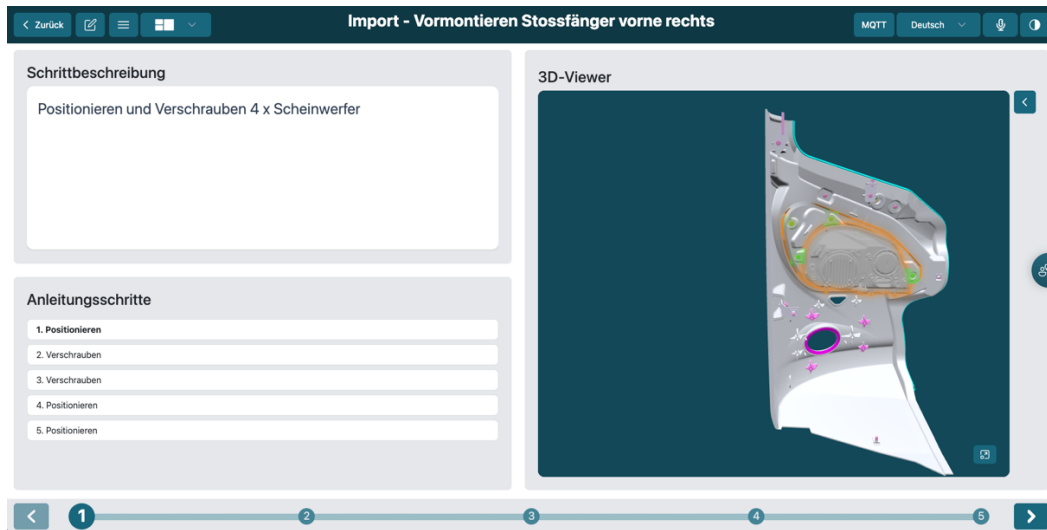


- **Conformity with expectations:** All user interface elements were designed such that all elements follow a consistent design language and color scheme. Self-descriptive icons were used for ease of interaction.
- **Learnability:** An undo button was included for a risk-free editing experience. Support for different languages including French, Turkish and English was implemented, partly via AI.
- **Controllability:** The central user interface during assembly tasks was designed to include multiple adaptation modalities for both visual aspects and content. This was achieved with a theme-button, language selection drop-down menu and the aforementioned widget-system.
- **Robustness:** Through the use of descriptive error messages and confirmation popups during deletion of instructions, the unwanted loss of data and mistakes during task performance are minimized.
- **User engagement:** To better align with the existing design language at deployment, the MontaViz system inherited the design language and colors from the existing infoX-System which is currently being used at Daimler Busses.

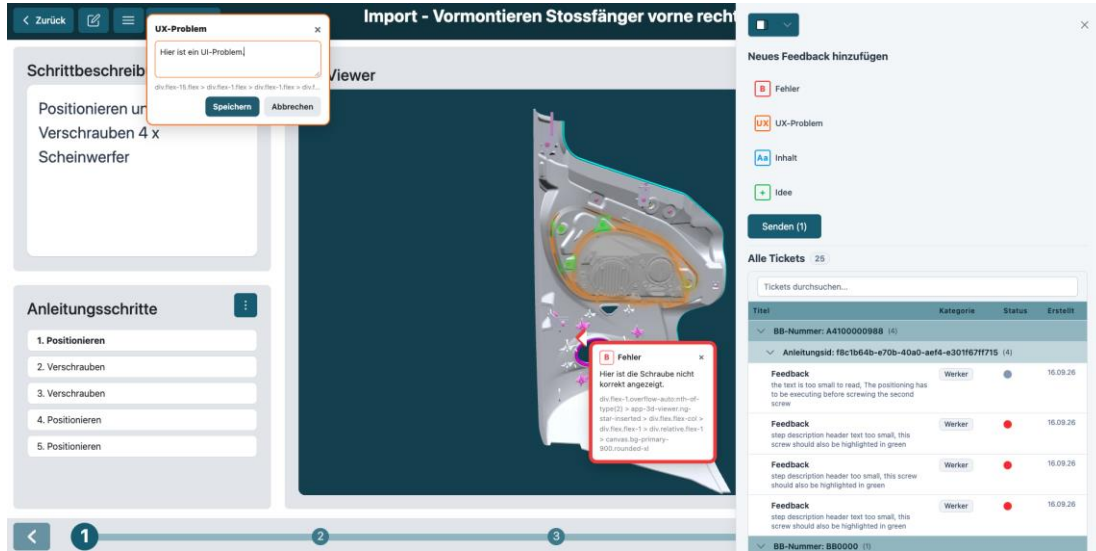
A further user interaction improvement was implemented in the form of a speech-recognition system that attempts to avoid issues that arise due to aspects such as dialects by allowing the operation via a number system. Each interactable element of the user interface is assigned a number upon activation of the system and simply by saying one of the numbers, the user interface can be operated. For example, the number one is always mapped to go one step back and the number two is always mapped to the advancement in the instruction set. More complex UI elements require a multi-step command, such as selecting the language menu with the number 5 and then selecting English with number 1. Or choosing the instruction list with number 0 and then speaking the full number of the step to move on to, such as “29”. This interaction modality allows for the improved accessibility of the system, particularly in situations where the physical distance between the worker and the system may be

too high for the user to interact with it otherwise.





The feedback feature was implemented to ensure the continuous consistency and accuracy of assembly instructions by enabling users to report errors and suggest improvements directly during the assembly process. Users can select a feedback category and drag and drop it onto the affected interface element, optionally providing additional context through a textual description. The feedback is linked to the specific instruction and displayed content, with both technical references (e.g., HTML elements and data paths) and a screenshot recorded to preserve the visual context. This also allows issues in previous software versions to be documented and traced, even when the technical references are no longer valid. Based on element-to-person mapping, the feedback should automatically be forwarded to the person responsible, distinguishing, for example, between UI-related issues and errors in textual instructions. Beyond direct error correction, the collected feedback can support future evaluations by identifying discrepancies between generated instructions and real-world assembly processes, potentially improving the overall quality of MontaViz.



Therefore, a centralized ticket system was developed to collect and manage feedback generated during the assembly process. It serves as a central point for reviewing and processing reported issues, with a potential future interface for forwarding more complex problems directly to the development team or other instances. Tickets can be supplemented with additional notes and managed through three status levels: "Open", "In Progress", and "Done". The system is also accessible within the instruction editing mode (AV mode), allowing assembly planners to review and address reported issues directly while modifying instructions. Furthermore, the ticket system was designed to accommodate future AI-generated suggestions for improving assembly instructions, providing a common platform for managing both user feedback and automated improvement proposals.

Back

MQTTEnglish

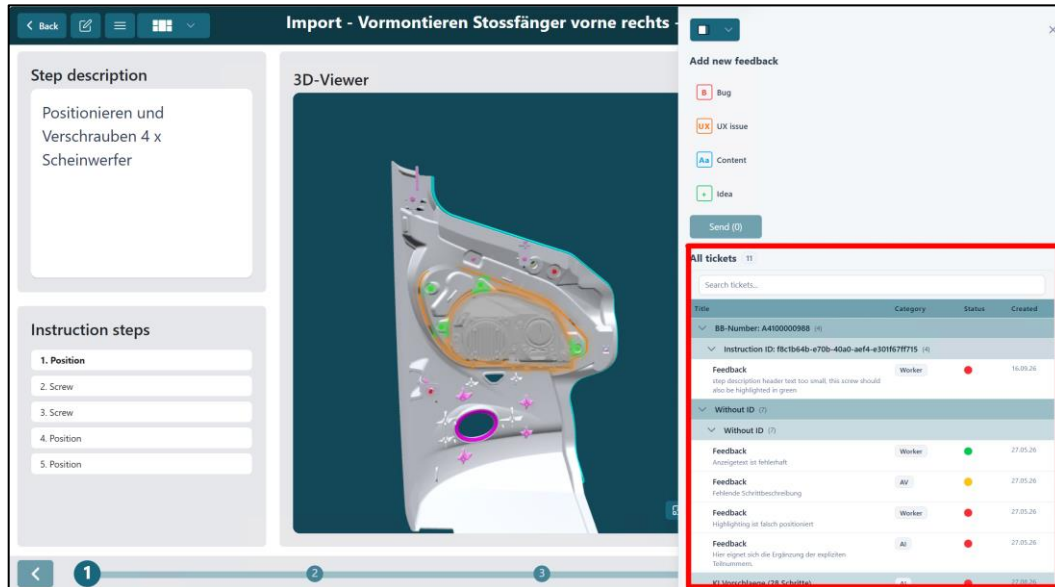
AV1

WORKER6

AI4

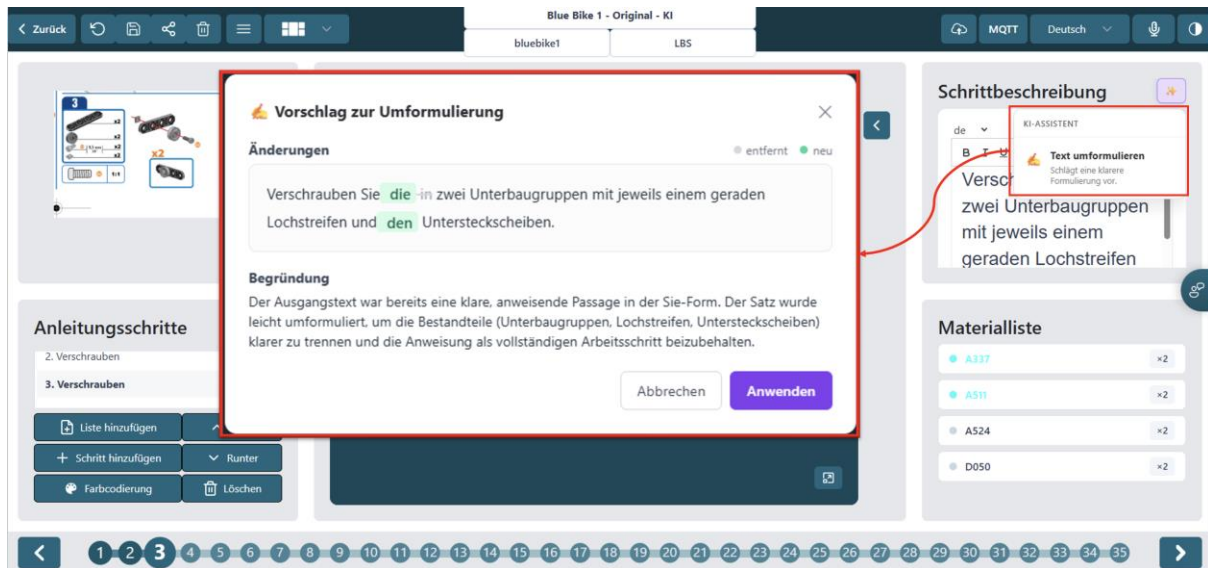
Tickets durchsuchen...

TITEL	KATEGORIE	STATUS	ERSTELLT AM	TI	Nach Schritt
BB-Number: A4100000988 (4)					
Instruction ID: f8c1b64b-e70b-40a0-ae4f-e301f67f715 (4)					
Feedback step description header text too small, this screw should also be highlighted in green	Worker	New	16.09.2026 15:01		
Without ID (7)					
Without ID (7)					
Feedback Anzeigetext ist fehlerhaft	Worker	Done	27.05.2026 16:34		
Feedback Fehlende Schrittbeschreibung	AV	In progress	27.05.2026 16:35		
Feedback Highlighting ist falsch positioniert	Worker	New	27.05.2026 16:37		
Feedback Hier eignet sich die Ergänzung der expliziten Teilnummern.	AI	New	27.05.2026 16:40		

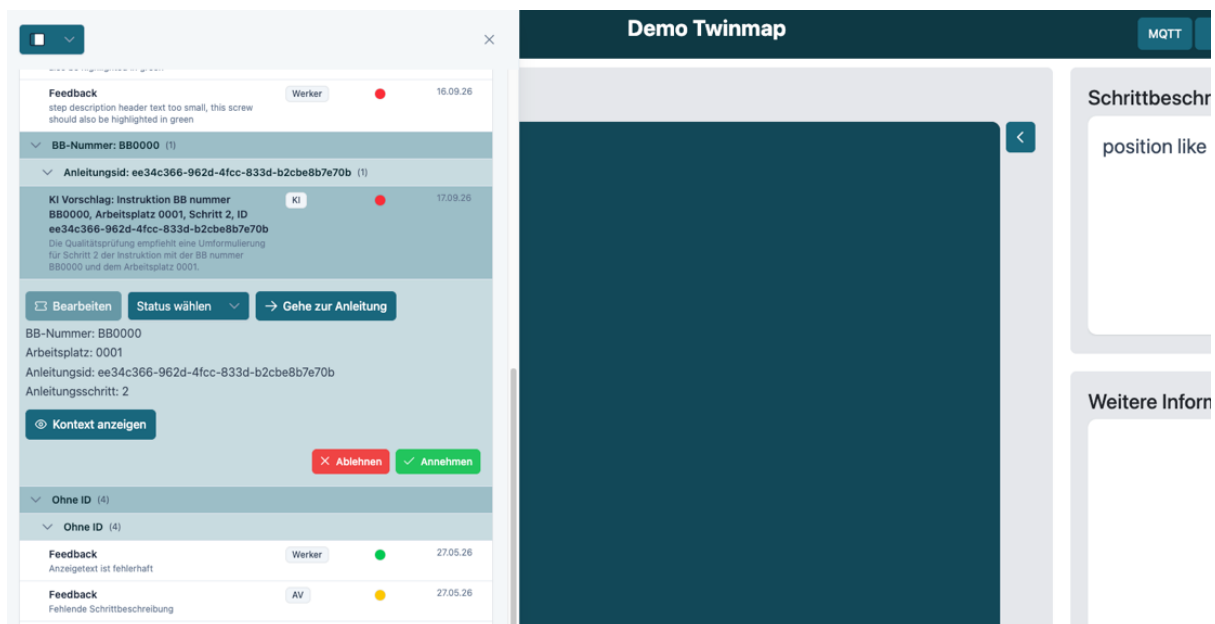


3.5.3 Adapted instructions

To improve the clarity of the instructions and reduce mistakes while creating or editing instructions, an AI-based System was conceived to semi-automatically check the quality of instruction descriptions and, if need be, suggest an improved reformulation of the instruction. To do this, the question of what constitutes a “good” instruction must be answered. To this end, multiple multiple multiple multiple multiple quantifiable metrics for quality assessment were determined from literature and summarized in a paper for the ICPS conference in Emden in 2025 titled “Towards Computational Quality Management for Automatically Generated Adaptive Work Instructions in Industry 5.0”. These metrics include relevance, correctness and completeness of the content of the instruction, as well as visual aspects, such as accessibility and the format of the instruction. These metrics could be used to assess the instruction descriptions and prompt a large language model to suggest a fitting reformulation. The system was implemented using Python and consists of a server, an Ollama instance of a carefully selected model (after an internal benchmarking process), and an instance of Unsloth for continuous, incremental improvement via Q-LoRA fine-tuning using the approved reformulations from MontaViz. During the creation or editing of an instruction, the Operator may use the user-interface to send a Rest-API-Call to the Python server and request a reformulation of a given instruction. This will be represented as shown below, with the reformulation content being shown in a Git-Style representation of the differences, alongside the reasoning response for the reformulation itself.

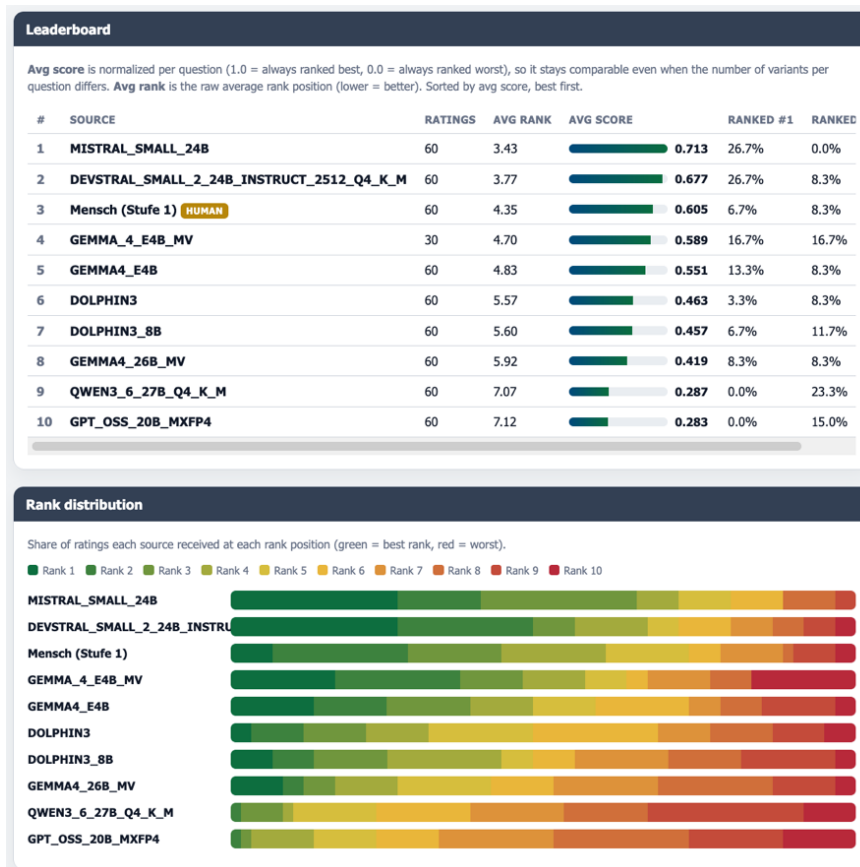


Should the operator require an entire work-plan to be reformulated, the overview of all instruction-sets also includes a button enabling batch-processing of all instructions in that given instruction set. The API call checks the quality of each instruction. In the case of a violation of any of the quality metrics, a reformulation will be submitted and shown to the user in form of a ticket. Alongside the reformulation, the results of the quality assessment and reasoning for the reformulation are included in the ticket.



To select an appropriate model for this task, the provided exports of technical assembly steps from the SAP system were parsed to the JSON format for further processing. These instructions were then manually re-written by two independent parties. These reformulations served as a baseline which were used to compare different large language models for the task at hand. Alongside an automated comparison, a small internal survey was conducted where participants ranked their preferred formulation. In total, eight models were compared to the human formulation, and results were

weighted by model parameter size to prioritize ease of deployability and reduced loading times for the user. The results of this comparison showed the models *Mistral 24B* and *Devstral_Small_2 24B* to have the best overall results, with the model *Gemma 4 E4B* having the overall best performance to size ratio. Thus, the model *Gemma 4 E4B* was used as a baseline to be fine-tuned using Unsloth for further performance improvements. The fine-tuned model was named *Gemma 4 E4B MV* and was included in a subsequent comparison run. The image below depicts the results of the manual ranking. While the fine-tuned model is surpassed by more complex generic models, the fine-tuned model only requires four billion parameters and thus can be hosted by a larger number of standard PC workstations, not relying on dedicated AI hardware. It could thus potentially run on edge devices at the shopfloor. At the same time, the fine-grained model achieves similar to human performance.



The image below depicts the results of the automated comparison.

Bestes Effizienz-/Leistungsverhältnis

GEMMA4_E4B

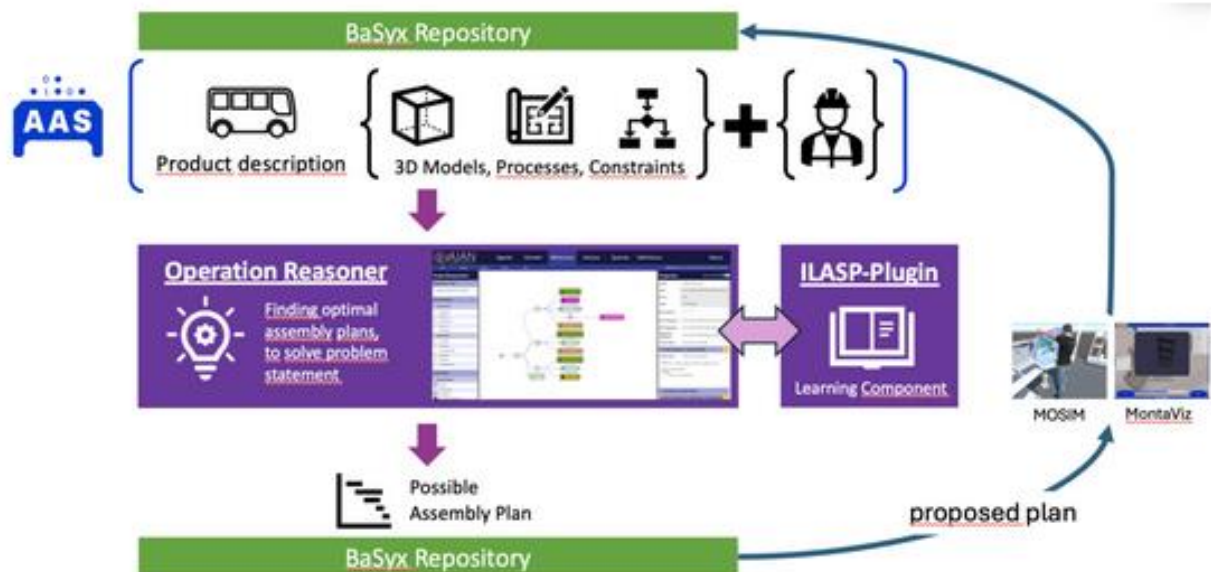
Groesse: 4B | Ähnlichkeit: 69.2% | Effizienz-Score: 0.3458

Effizienz-Rangliste (Genauigkeit relativ zur Modellgröße)

#	MODELL	GROESSE	ÄHNLICHKEIT	EFFIZIENZ-SCORE
1	GEMMA4_E4B	4B	69.2%	0.3458
2	GEMMA_4_E4B_MV	4B	63.5%	0.3176
3	DOLPHIN3_8B	8B	66.9%	0.2366
4	DEVSTRAL_SMALL_2_24B_INSTRUCT_2512_Q4_K_M	24B	70.8%	0.1445
5	MISTRAL_SMALL_24B	24B	70.5%	0.1438
6	GPT_OSS_20B_MXFP4	20B	62.7%	0.1403
7	GEMMA4_26B_MV	26B	68.1%	0.1336
8	QWEN3_6_27B_Q4_K_M	27B	62.7%	0.1207

Integration of MontaViz in the overall system:

- Integration of real-time data and expert feedback.
- Deep coupling between planner (agent).
- Novelty: Bi-directional feedback between feasibility checks and plan refinement.



3.6 Teaching instructions generation

3.6.1 Context and Identified Problems

The company JVC, operating in the textile sector and within a highly demanding market environment, is characterized by a service-based industrial business model, in which production capacity is offered to third parties in the form of manufacturing hours and processes. This operating context entails high production variability, very short production line life cycles, and a constant need to adapt to specific customer requirements, frequently associated with the luxury market. In addition, the high number of new production orders gained each year requires a rapid and consistent response in terms of production planning and execution.

Within this framework, JVC faces structural challenges related to the shortage of specialized human resources, particularly personnel with sufficient experience to simultaneously support multiple operators on the factory floor. The transmission of operational know-how is often highly dependent on a limited number of experts, whose availability is constrained, making operator training and supervision processes time-consuming and difficult to scale. Whenever an operator does not fully master a task, the need to wait for specialized support or the incorrect execution of operations results in delays, rework, and increased production costs, negatively affecting overall line performance.

Shortage of specialized workforce, absenteeism, and constant pressure to meet short lead times further exacerbate these challenges, making it difficult to ensure consistent execution aligned with the methods defined as optimal. The absence of structured and standardized mechanisms to support task execution contributes to result variability and the loss of operational efficiency.

3.6.2 Contribution of the teaching instructions generation tool

The solution implemented within this work package directly addresses these challenges through the deployment of a structured system for generating and presenting work instructions. This system enables existing production know-how to be translated into clear, sequential, and operator-accessible instructions, ensuring that each task is executed according to an established reference method.

By providing direct guidance to operators at the point of execution, the system has reduced the need for continuous specialist intervention, allowing expert resources to focus on higher value-added activities. The use of visual content, such as videos or illustrated sequences, facilitates the understanding of operations and ensures a uniform interpretation of work methods, even in environments characterized by high production variability.

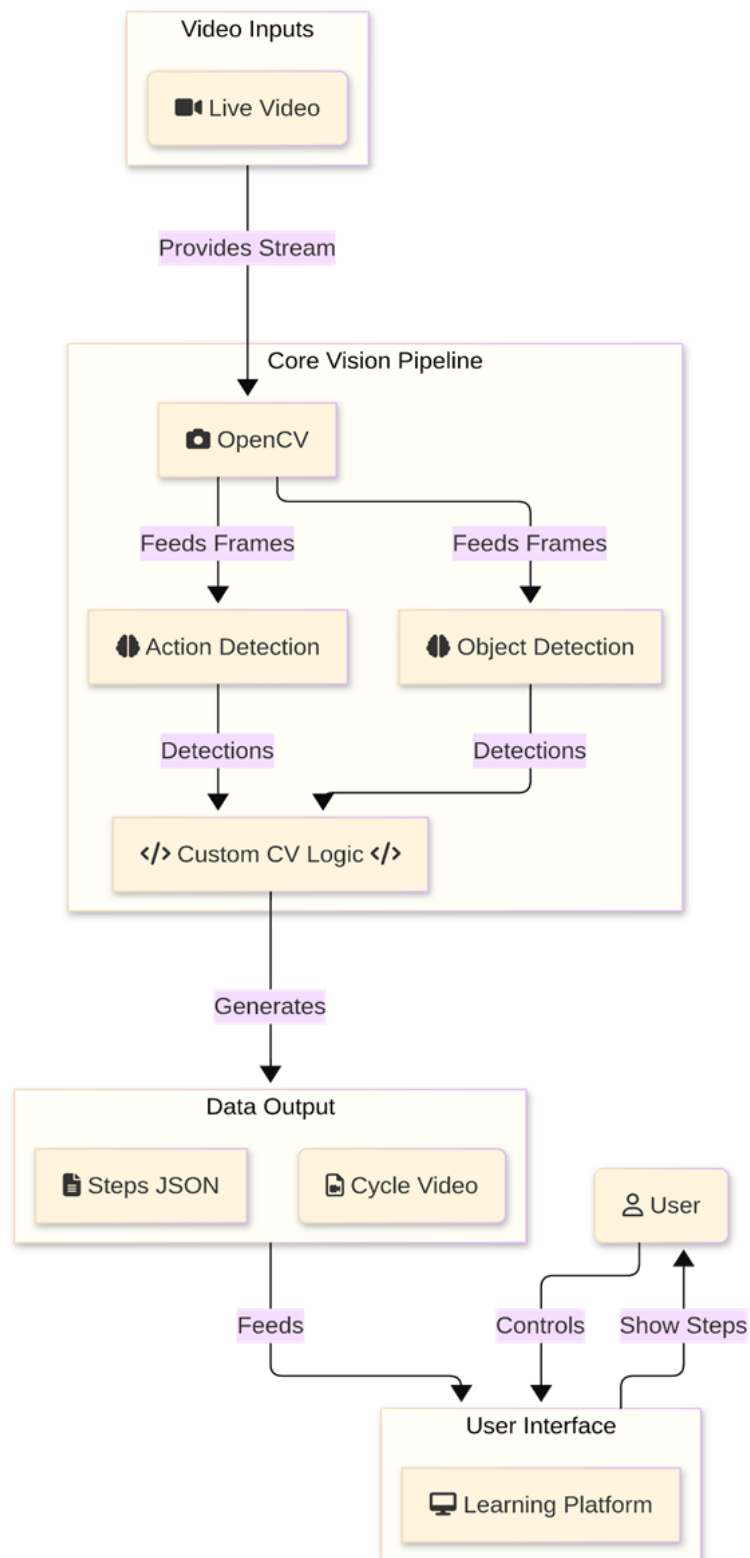


Figure 1 – Teaching instructions generation: High-level architecture

3.6.3 Teaching instructions generation Overview

To support the creation of accurate, adaptive, and context-aware work instructions, the implemented system enables the structured capture of expert knowledge directly from real production environments. The procedure begins with an experienced worker or trainer performing a simulation of a specific task associated with a defined work order, executed under real operational conditions at the workstation. This simulation reflects best practices, correct sequencing, and optimal ergonomics, serving as the reference execution for subsequent training and assistance activities.

The workstation has been instrumented with two video cameras positioned to cover both the worker and the relevant operational area, including the machine, tools, fabric, and surrounding workspace. These cameras continuously capture visual data throughout the session, ensuring sufficient coverage to observe body posture, hand movements, object manipulation, and interactions with equipment.



Figure 1 - Workstation areas definition

Before initiating the task, the worker or trainer explicitly signals the start of a training session to the system through the user interface. Upon confirmation, the system begins recording the video stream(s) and activates the contextual recognition modules. During task execution, the system applies computer vision techniques to digitally recognise and track equipment, tools, materials, and relevant components, as well as the worker's actions and movements. This enables the identification of which objects are being manipulated, how they are being used, and in which sequence.

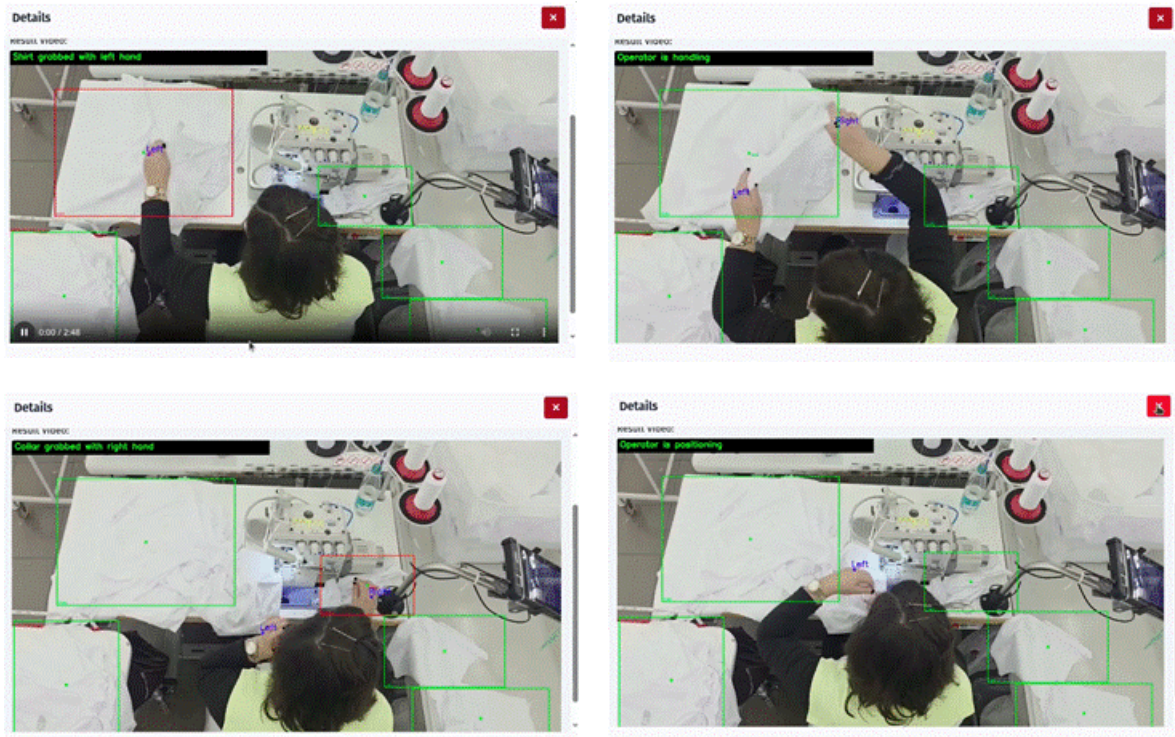


Figure 2 - Movements and actions recognition

As the task progresses, the system generates an initial textual description of the actions performed, capturing key elements such as movements, postures, object interactions, and transitions between steps. These descriptions are derived from sensor data, vision-based tracking, and interaction events, resulting in a structured, time-aligned representation of the expert's execution.

To ensure usability and pedagogical effectiveness, the generated action descriptions are processed using AI and Large Language Model (LLM) tools. This processing step transforms raw, technical descriptions into clear, concise, and human-readable instructions, filtering unnecessary technical jargon and adapting the language to the target trainees' qualifications, experience level, and linguistic context. The resulting instructions are expressed in simple, unambiguous sentences suitable for training and real-time assistance.

Last Processes			
Check the last detection processes			
			REFRESH
Process Name	Status	Start Time	End Time
Shirt_4Components	COMPLETED	13/01/26 15:36:35	13/01/26 16:56:59
teste	COMPLETED	14/01/26 09:08:20	14/01/26 09:15:40
DETAILS OPEN LOGS DELETE RE-PROCESS DOWNLOAD REPORT			

Analysis Report for: Shirt_3Components	
Components Used:	
- Shirt	
- Collar	
- Left	
- Right	
Best Standard Cycle Time: 892.00 seconds	
Time Resume for Best Standard Cycle:	
- Total Handling Time: 24.00 seconds	
- Total Sewing Time: 266.00 seconds	
Best Standard Circuit:	
1. 00:00:00.000 Shirt grabbed with left hand	
2. 00:00:03.010 Operator is handling	
3. 00:00:10.145 Operator is positioning	
4. 00:00:34.175 Operator is sewing	
5. 00:00:36.840 Collar grabbed with right hand	
6. 00:00:38.410 Operator is handling	
7. 00:00:50.919 Operator is positioning	
8. 00:00:53.189 Operator is handling	
9. 00:00:58.893 Operator is positioning	
10. 00:01:24.063 Operator is sewing	
11. 00:01:27.604 Operator is positioning	
12. 00:01:29.627 Operator is sewing	
13. 00:01:35.716 Operator is positioning	
14. 00:01:48.339 Operator is handling	
15. 00:01:48.530 Operator is positioning	
16. 00:02:00.610 Operator is handling	
17. 00:02:02.671 Operator is positioning	
18. 00:02:09.342 Operator is sewing	
19. 00:02:21.278 Operator is positioning	
20. 00:02:27.931 Operator is sewing	
21. 00:02:30.020 Operator is positioning	

Figure 3 Storing video and generating instructions

The final set of instructions constitutes a formal reference model of the task, representing the expected sequence of actions, object interactions, and movements. These instructions are not only used for training purposes but also serve as a baseline for runtime monitoring and comparison, forming a core input for the context-based worker assistance system (Work Package 4).

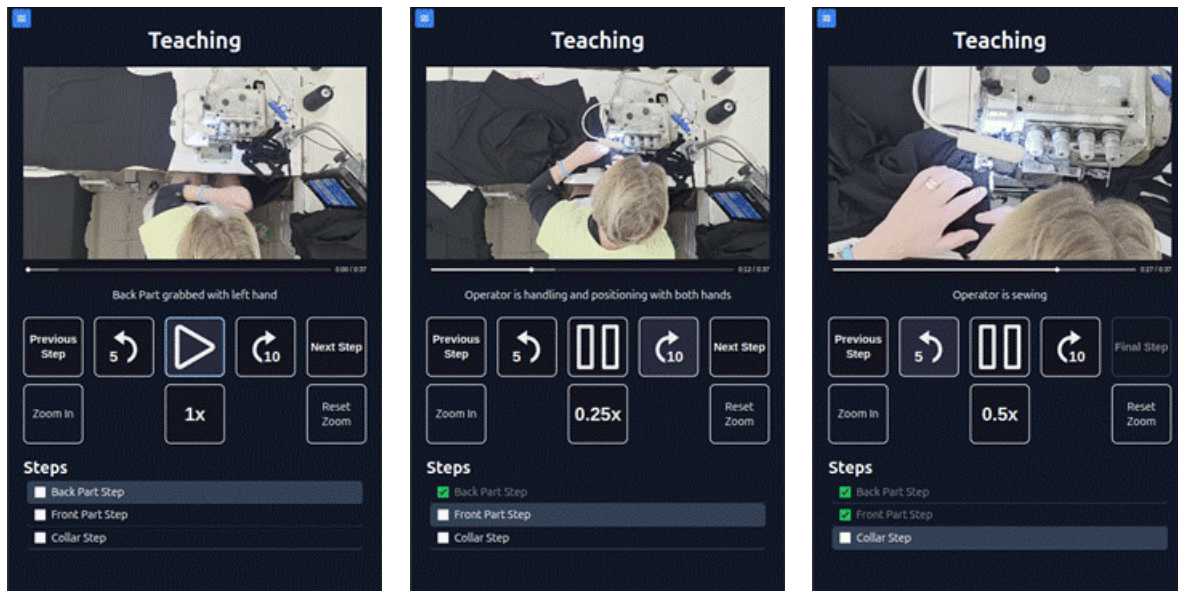


Figure 4 - User interface for the Learning Platform

During live operation, the system compares the actions performed by trainees or operators against the expected actions defined in the instruction set, enabling real-time guidance, deviation detection, and adaptive feedback.

Through this procedure, expert tacit knowledge is systematically captured, formalised, and operationalised, ensuring consistency, scalability, and continuous improvement of work instructions across training and production environments.

4 Statements from the usecase owners

4.1 Volvo

The work carried out in Work Package 3 provides tangible examples of how Volvo's ongoing digitalization journey could evolve in the future. It illustrates a next step in which planning, workforce management, work instructions, ergonomics, and production data are more tightly integrated and continuously updated through real-time feedback. The project demonstrates how digital tools can enable dynamic replanning, adaptive work instructions, and systematic capture of knowledge from experienced operators. It also shows how competence, productivity, and ergonomics can be managed as interconnected dimensions of performance rather than as separate optimization targets. Perhaps most importantly, it highlights how production data can be continuously fed back into planning and learning processes, creating a more adaptive, data-driven, and continuously improving production system.

Throughout the project, Volvo engineers have played an active role in defining requirements for several of the developed tools, including solutions for task sequencing, layout planning, production sequencing and balancing, and workforce planning. They have also participated in evaluating and testing tool prototypes in realistic use-case scenarios. The results have been highly encouraging, and the tools were positively received by the Volvo employees involved in the evaluations.

In this respect, the project provides a practical vision of what future manufacturing operations at Volvo could look like as the company continues to advance its digital capabilities and build on the insights generated through this work.

4.2 Daimler

The work carried out in **Work Package 3** directly supports the digitalization strategy of **Daimler Buses** by demonstrating how key manufacturing processes can be connected through an integrated digital ecosystem. The developed prototype solutions illustrate how production planning, workforce management, work instructions, ergonomics, and production data can be seamlessly linked and continuously updated through real-time feedback mechanisms.

The project demonstrates the potential of digital technologies to enable dynamic production planning, adaptive work instructions, and the systematic capture and transfer of knowledge from experienced operators. Furthermore, it shows how workforce competence, productivity, and ergonomics can be managed as interconnected dimensions of operational performance, rather than being optimized in isolation.

Most importantly, the project highlights how production and operational data can be continuously fed back into planning, decision-making, and learning processes. This

creates the foundation for a more agile, data-driven, and continuously improving production system, supporting Daimler Buses in its journey toward smarter and more resilient manufacturing operations.

Throughout the project, experts from Daimler Buses played an active role in defining use cases, specifying requirements, and evaluating the developed solutions. They also participated in the validation and testing of tool prototypes, ensuring that the outcomes were closely aligned with real manufacturing needs. In addition, several PhD students contributed to the development of innovative solution approaches, bringing cutting-edge research and technical expertise into the project.

Overall, the project achieved highly positive results, demonstrating the value and potential of the developed tools and methodologies. Feedback from Daimler Buses employees was consistently encouraging, confirming both the relevance and practical usability of the solutions. The project provides a realistic vision of future manufacturing operations at Daimler Buses and illustrates how advanced digital technologies can be leveraged to further enhance efficiency, operational excellence, and innovation across the production environment.

4.3 JVC

The work carried out within Work Package 3 directly addresses the structural challenges faced by JVC. Operating in the textile sector, the company is characterised by a service-based industrial business model, dealing with high production variability, very short line life cycles, and a continuous need to adapt to specific customer requirements, frequently linked to the luxury market.

The implementation of the teaching instructions generation tool provides an essential solution to the shortage of specialised human resources and the strong dependence on a limited number of experts for operator training and supervision. Through the use of cameras and computer vision techniques, the system allows us to systematically capture best practices directly from the real production environment, recognising and tracking tools, materials, equipment, and worker movements.

The system's ability to process this data using AI tools and Large Language Models (LLMs) to transform raw technical descriptions into clear, concise work instructions tailored to the experience level and qualifications of trainees represents a major technological milestone.

Adopting this solution significantly reduces the need for continuous intervention and supervision by our specialists, enabling them to focus on higher value-added activities. At the same time, it provides direct guidance at the point of execution, ensuring a uniform interpretation of work methods and minimising operational errors that previously led to delays, rework, and increased production costs.

5 Conclusions

The assembly lines for mixed model production can be arranged in different ways depending on the properties of the products and production equipment needed. In the workflows of planning the production, there are identified challenges that can be

addressed in parallel, either independently or with a feedback loop. The challenges may vary a bit depending on how lines are arranged, but they are similar. The key observation and motivation for this project is that there is a need for decision support tools to organize work and reduce stress while quality and productivity increase. The project outcome regarding assembly line planning (WP3) included the Proof-of-concepts presented in this document (section 3). They all address some essential planning and instruction generation challenges faced in mixed model production that include human operators on the line.

6 Dissemination results

Ongoing and coming, the Proof-of-Concepts were disseminated to the use-case owners for evaluation and validation in Q1 2026 (some in the beginning and some in the end of Q1). There have been iterations and improvements of the PoCs, which continue until the end of the project. This internal dissemination includes presentations to the business units.

We arranged a workshop called “Smart Workers in Industry 5.0 — Human-Centric Assistance and Work Planning” at the 19th Pervasive Technologies Related to Assistive Environments (PETRA 2026), chaired by Dr. Jonas Blattgerste. <https://mixality.de/SWI5/>

Previously, we arranged a special session “Integrating Smart Workers in ICPS: Work Planning, Scheduling and Active Assistance” at IEEE ICPS in Emden in May 2025, chaired by Prof. Thies Pfeiffer. https://icps2025.ieee-ies.org/special_sessions/SS6_CFP.pdf The PoCs described in this document originates from results in our published papers, as follows:

- E. Erős, A. Hanna, N. Sosdean and K. Åkesson, "Towards Fair Employee-Job Assignments in Final Assembly Stations," 2025 IEEE 8th International Conference on Industrial Cyber-Physical Systems (ICPS), Emden, Germany, 2025, pp. 1-6, doi: 10.1109/ICPS65515.2025.11087879. [3.1]
- M. Vinetti and M. Fabian, "Identification of Minimally Restrictive Assembly Sequences Using Supervisory Control Theory," 2025 IEEE 8th International Conference on Industrial Cyber-Physical Systems (ICPS), Emden, Germany, 2025, pp. 1-4, doi: 10.1109/ICPS65515.2025.11087917. [3.1]
- M. Vinetti, S. F. Roselli and M. Fabian (2026a). Multi-Worker Assembly Line Rebalancing with Relevance-Guided Configuration Preservation. 2026 IEEE International Conference on Automation Science and Engineering (CASE), Shenyang, China.
- M. Vinetti, S. F. Roselli and M. Fabian (2026b). A Multi-Worker Assembly Line Rebalancing with Spatial and Ergonomic Considerations. 2026 International Federation of Automatic Control (IFAC), Busan, South Korea.
- C. Fink, O. Schelén and U. Bodin, "Work in Progress: Decision Support System for Rescheduling Blocked Orders," 2025 IEEE 8th International Conference on Industrial Cyber-Physical Systems (ICPS), Emden, Germany, 2025, pp. 1-4, doi: 10.1109/ICPS65515.2025.11087891. [3.1]
- C. Fink, O. Schelén and U. Bodin, "Improving a SALBP-1 Solver by Tuning Flexible Termination Criteria with Simulated Solving," 2025 IEEE 8th International Conference on Industrial Cyber-Physical Systems (ICPS), Emden, Germany, 2025, pp. 1-7, doi: 10.1109/ICPS65515.2025.11087837. [3.1]

- A. Antakli and D. Spieldenner, "Adaptive Assembly Plan Generation in Industry 4.0," 2025 IEEE 8th International Conference on Industrial Cyber-Physical Systems (ICPS), Emden, Germany, 2025, pp. 01-05, doi: 10.1109/ICPS65515.2025.11087860. [3.2 + 3.3]
- J. Sprenger, A. Antakli and K. Fischer, "The MOSIM Framework: Simulating Smart Workers in the Industrial Metaverse," 2025 IEEE 8th International Conference on Industrial Cyber-Physical Systems (ICPS), Emden, Germany, 2025, pp. 1-8, doi: 10.1109/ICPS65515.2025.11087833. [3.3]
- J. Sprenger, N. Braun, S. Reichardt and B. Kellner, "Towards a Continuous Ergonomic Evaluation of Manual Workstations in the Digital Factory," 2025 IEEE 8th International Conference on Industrial Cyber-Physical Systems (ICPS), Emden, Germany, 2025, pp. 1-4, doi: 10.1109/ICPS65515.2025.11087913. [3.3 + 3.4]
- A. Dirks, J. Franssen and T. Pfeiffer, "Towards Computational Quality Management for Automatically Generated Adaptive Work Instructions in Industry 5.0," 2025 IEEE 8th International Conference on Industrial Cyber-Physical Systems (ICPS), Emden, Germany, 2025, pp. 1-6, doi: 10.1109/ICPS65515.2025.11087874. [3.4]
- B. P. P. M. Pathirathna and M. Thron, "Real-Time Operation Identification Using a Structure Evolving Fuzzy Finite State Machine," 2025 IEEE 8th International Conference on Industrial Cyber-Physical Systems (ICPS), Emden, Germany, 2025, pp. 1-4, doi: 10.1109/ICPS65515.2025.11087852. [3.4]
- L. R. Escobedo, S. Straßburger and T. Bär, "Manufacturing Cost Optimization Through KPI Monitoring and Intelligent Digital Machine Twins," 2025 IEEE 8th International Conference on Industrial Cyber-Physical Systems (ICPS), Emden, Germany, 2025, pp. 1-4, doi: 10.1109/ICPS65515.2025.11087863.
- L. R. Escobedo, S. Straßburger and T. Bär, "Demonstration-Based AI Learning for Dynamic Motion Response in Metal Powder 3D Printing," 2025 IEEE 8th International Conference on Industrial Cyber-Physical Systems (ICPS), Emden, Germany, 2025, pp. 1-6, doi: 10.1109/ICPS65515.2025.11087910.
- L. R. Escobedo, S. Straßburger and T. Bär, "Towards Energy Optimization in Metal 3D Printing: An Assisted Simulation Approach for TruPrint 3000," 2025 IEEE 8th International Conference on Industrial Cyber-Physical Systems (ICPS), Emden, Germany, 2025, pp. 1-4, doi: 10.1109/ICPS65515.2025.11087853.