Materials Handling and Intralogistics

VDMA

LIF – Layout Interchange Format

Definition of a format of track layouts for exchange between the integrator of the driverless transport vehicles and a (third-party) master control system.

Version 1.0.0 – March 2024

Abstract

The following embodiment describes an interchange format for a track layout (e.g.: collection of edges, nodes and stations). By means of this interchange format, the integrator of the driverless transport vehicles will be able to initially transfer a track layout to a central (third-party) master control system for use and integration.

This document represents a non-binding approach. Whoever uses it must ensure the correct application in the specific case. It is influenced by the state of the art at the time of the respective edition, in particular the VDA 5050 interface definition. Ascribing to the suggestions described herein does not absolve parties of the responsibility for their own actions. No text in this document claims completeness nor provides exact interpretation of the existing legal provisions. The contents of this document must not replace the study of the relevant directives, laws and regulations. Furthermore, the special features of the respective products as well as their different possible applications must be considered. In this respect, all parties act at their own risk. Any liability of the VDMA and those involved in the development or application of the suggestions is excluded.

Should you encounter any inaccuracies or the possibility of incorrect interpretation in the application of the proposals, please notify VDMA immediately so that any deficiencies can be rectified.

Publisher Verband Deutscher Maschinen- und Anlagenbau e. V. (VDMA)
Lyoner Strasse 18, 60528 Frankfurt am Main

Status January 2024

Version 1.0.0

Contents

Contents

1 Terms

Terms are generally used as they are in the VDA 5050 interface.
The following table is intended to describe supplementary terms:

Item	Description
deadlock	A situation where two or more devices are awaiting one another in a circular fashion, resulting in a system that is unable to exit this state and continue regular operation. Example: Vehicle A is waiting on vehicle B to get out of the way, but vehicle B is also waiting on vehicle A to do the same.
facility	Facility in which the driverless transport system is used. The facility can consist of several levels. The facility could be made up by several LIF files from multiple vehicle integrators. The facility is controlled by one (third-party) master control system.
integrator	Integrator refers to the manufacturer of driverless transport vehicles or a vendor that integrates a manufacturer's driverless transport vehicles into the driverless transport system.
layout	A collection of nodes, edges and stations. A layout represents a level of a facility or a part of a level of a facility.
level	A level of a facility that is used by the driverless transport systems
re-entry	The induction of a vehicle into automatic management under the (third-party) master control system, such as after having been taken under manual operation, or when the vehicle is first inducted into the system after having been switched off.
station	Any point at which a vehicle can explicitly interact with the environment, including but not limited to physical interactions.

2 Applicable Documents

Document	Description
VDI-Richtlinie 2510	Driverless transport systems
VDI-Richtlinie 4451 Blatt 7	Compatibility of driverless transport systems – Master control for driverless transport systems
DIN EN ISO 3691-4	Industrial trucks – Safety requirements and verification – Part 4: Driverless trucks and their systems
VDA 5050	Interface for communication between automated guided vehicles (AGV) and a master control

3 Foreword

The Layout Interchange Format (LIF) was defined at Verband Deutscher Maschinen- und Anlagenbau e. V. Fachverband Fördertechnik und Intralogistik (VDMA).

Proposals for changes to the standard format are to be submitted to the VDMA and will be adopted in a future version if the decision is positive.

4 Aim of the Document

This document describes the LIF, its purpose and examples of how to use it. This document does not describe any logical processes that a (third-party) master control system must implement to interpret the data contained in the LIF.

5 Aim of the LIF

The objective of the Layout Interchange Format is to standardize a way for the definition of automated vehicle track layouts to be presented toward (third-party) master control system providers.

The first primary goal is to complement the VDA 5050 interface's goal of facilitating decoupling between a vehicle manufacturer and a (third-party) master control system provider. It uses the same terminology and much of the same structure as the VDA 5050 interface.

The LIF described in this document is intended to map a common set of necessary information to enable a (third-party) master control system to steer/navigate a vehicle on a track layout specified by the vehicle integrator. The LIF contains information on how the vehicle integrator's vehicles can interact with their environment and navigate inside of a track layout. This satisfies the LIF's second primary goal to allow a clear separation of responsibility between a vehicle integrator and a (third-party) master control system.

5.1 Requirements

- The LIF concept, standard, and definition must always be compatible with the current status, terminology and developments of the VDA 5050 interface.
- The LIF format may only contain track layouts from one vehicle integrator.
- The LIF may contain multiple track layouts for multiple vehicle types of one vehicle integrator.
- A (third-party) master control system must be able to accept multiple LIFs from multiple vehicle integrators for one facility.
- The LIF must not preclude the inclusion of vehicles with different levels of autonomy.

5.2 Further Assumptions

- The communication between the (third-party) master control system and the vehicle corresponds to the VDA 5050 interface definition.
- The vehicle integrator will also provide the (third-party) master control system with the AGV Fact Sheet per the VDA 5050 specification, which will contain information about vehicle geometry, kinematics and other “capabilities of the vehicle”.

5.3 LIF Limitations

- The LIF does not describe any logical processes by which a (third-party) master control system must perform its tasks. This includes, but is not limited to the handling of route planning, traffic management, intersections of multiple vehicles from the same or different vehicle integrators, interaction with stationary equipment and so forth. It is merely a definition of what a vehicle is capable of doing, and where, that a (third-party) master control system can use as input when determining these operations.
  Section 7.2, Import and Processing of the LIF File by the (Third-party) Master Control System, goes into further detail.
- The LIF does not affect, and is not affected by, different localization technologies that vehicles may use, nor does it contain any information pertaining to localization methods.
- The LIF is never intended to flow in the reverse direction of from a (third-party) master control system toward a vehicle or vehicles. If a vehicle integrator requires some information from a master control system or those responsible for it, it must be transferred outside of the context of the LIF.

6 LIF Format

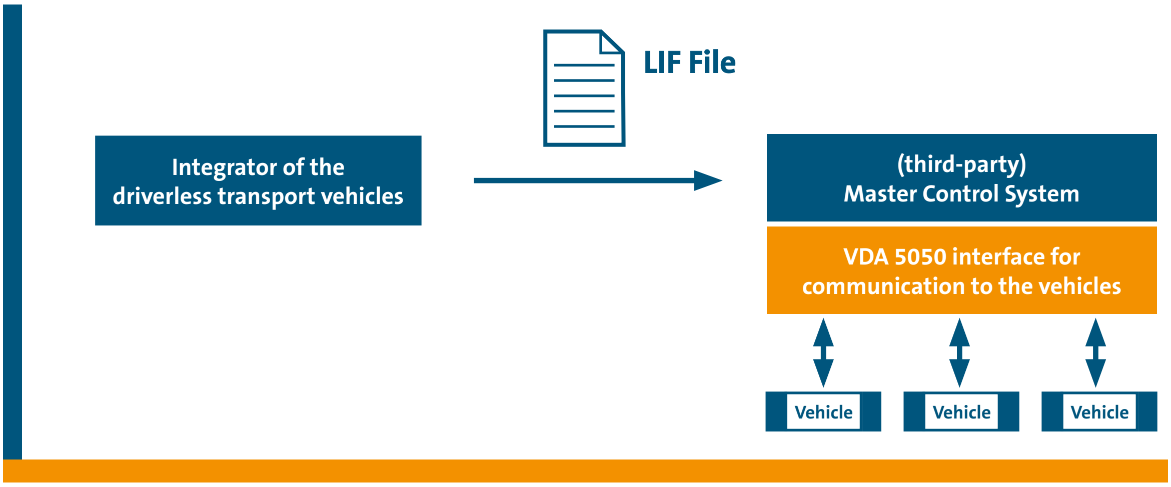
A JSON structure is used for the exchange format. JSON strings must conform to the RFC 8259 description for object notation. Keys must be strings and values must be a valid JSON data type (string, integer, float, object, array, boolean or null). The data is case sensitive.

The JSON structure allows for future extension of LIF with additional parameters. The parameters are described in English to ensure that LIF is also readable, understandable and applicable to the broadest possible audience.

7 LIF Transfer and Responsibilities of Vehicle Integrator and (Third-party) Master Control System

The following section describes the exchange of a LIF file between the integrator of driverless transport vehicles and a (third-party) master control system, and includes:

1. Export of the LIF file by the integrator of the driverless transport vehicles.
2. Import and processing of the LIF file by the (third-party) master control system.
3. Further exports of the LIF file and imports into the (third-party) master control system, such as incremental updates or changes.


LIF File
Integrator of the driverless transport vehicles
(third-party) Master Control System
VDA 5050 interface for communication to the vehicles
Vehicle
Vehicle
Vehicle

7.1 Export of the LIF File by the Integrator of the Driverless Transport Vehicles

The planning and definition of the track layout is done by the integrator of the driverless transport vehicles (e.g. by means of a planning or design tool). The vehicle integrator should plan the track layout in compliance with safety relevant standards (e.g.: minimum distances, speed reduction on certain edges, etc.) and considering the analysis of the envelope of the vehicles.

After the vehicle integrator has physically tested and verified that the track layout can be followed by the vehicles in compliance with the safety-relevant standards, the vehicle integrator should present the track layout to the (third-party) master control system by means of a LIF file via data transfer. The process of transfer can be agreed individually between the vehicle integrator and the (third-party) master control system.

The elements that are exported into the LIF file must include:

- The collection of all pathway nodes and any node-specific actions.
- The collection of all edges between these nodes and any edge-specific actions.
- The collection of stations on which the vehicle may perform actions.

7.2 Import and Processing of the LIF File by the (Third-party) Master Control System

The (third-party) master control system should import the LIF data to understand how a vehicle or vehicles can move on the given track layout, as well as the actions that can be performed at the various places within it.

The (third-party) master control system is responsible for the logic ensuring that all commands sent to a vehicle or vehicles based on information from a LIF file or files never result in conflicting commands with other vehicles also under its control, including but not limited to examples such as commanding two vehicles to drive through an intersection at the same time, creating deadlocks between multiple vehicles, and so forth. The (third-party) master control system is further responsible for ensuring that any actions it sends to vehicles that are not explicitly defined for a node or edge in the LIF file are indeed valid—this may require further coordination and communication between the (third-party) master control system and the vehicle integrator. It is always the responsibility of the (third-party) master control system to ensure it has all of the information required to make such determinations.

Based on the provided track layout, the routes for the individual vehicles are to be calculated dynamically at runtime by the (third-party) master control system that has consumed one or more LIF files from one or more vehicle integrators and/or for one or more vehicle types. The communication between the (third-party) master control system and the driverless transport vehicles takes place via the VDA 5050 interface.

Further information about the behaviour of a system must be obtained from outside of the definition of the LIF file. These things may include, but are not limited to:

- Traffic control of the vehicles on the track layout:
  - Method of concurrent route calculation for the vehicles
  - Regulation of intersections
  - Regulation of right of way
  - Congestion avoidance
- Attributes and parameters required for the management of the vehicles:
  - Disposition of the vehicles
  - Battery management of the vehicles
- Communication with the system periphery (e.g.: automatic stations, elevators, doors, etc.)
- Connection to higher-level systems (e.g.: material flow computer, warehouse management systems, etc.)
- Expansion to include specific elements of the master control system

7.3 Further Exports of the LIF File and Imports into the (Third-party) Master Control System

As soon as changes are to be made to the track layout or vehicle behaviour, the vehicle integrator must provide the (third-party) master control system with an updated or adapted LIF file which reflects them. The vehicles utilizing the new information in the updated LIF file should not be used; the vehicle integrator then must await confirmation from the (third-party) master control system provider that this updated LIF file has been processed and its changes incorporated into the (third-party) master control system. It is the responsibility of the (third-party) master control system to re-process the new LIF file, incorporating any changes, and then to notify the vehicle integrator that this has been completed. Both parties then confirm that they are ready to use the updated system definition. Then and only then are the changes to the system complete and ready for use, and the vehicles should resume operation.

Attention

Changing a vehicle's behaviour without also updating the LIF file possessed by the (third-party) master control system leads to inconsistencies—potentially harmful or destructive ones. Likewise, a (third-party) master control system that changes information gained from the LIF file (e.g. change of track layout) without asking the vehicle integrator to also implement these changes to supply a new LIF file reflecting them, removes and adopts all liability from the vehicle integrator, and this can lead to potentially harmful outcomes.

8 Specification of LIF

The following section describes the structure and details of the Layout Interchange Format and the contents of a Layout Interchange Format file. Proposals for changes to the standard format are to be submitted to the VDMA and will be adopted in a future version if the decision is positive.

8.1 Table Symbols and Meaning of Formatting

Each table contains the name of each variable, its data type, its unit if applicable, and a description.

8.1.1 Optional Variables

If a variable is marked as optional, it is optional for the vehicle integrator's vehicles. The (third-party) master control system must be able to handle optional variables being either specified or not.

If the LIF file contains an optional variable, the (third-party) master control system must not ignore the variable. If the (third-party) master control system cannot process the variable accordingly, it is expected that the (third-party) master control system will provide a warning or an error message when importing the LIF file.

Variables that are optional in the LIF, but are strictly required by the vehicle, must be clearly communicated toward the (third-party) master control system.
The LIF does not denote such variables; this agreement must be made between the vehicle integrator and (third-party) master control system. It is suggested this is written in an agreement parallel to the AGV Fact Sheet as defined in the VDA 5050 standard.

8.2 Element ID Uniqueness

Certain elements, namely: Layouts, Nodes, Edges and Stations have IDs associated with them. These IDs should be unique among their type.

8.1 Table Symbols and Meaning of Formatting

Identification	Description
standard	Variable is an elementary data type.
bold	Variable is a non-elementary data type (e. g. JSON object or array) and defined separately.
italics	Variable is optional.
arrayName [squareBrackets]	Variable (here arrayName) is an array of the data type specified in the square brackets (here the data type is squareBrackets).

8.3 Elements of LIF

8.3.1 LIF Structure

The facility is described by a collection of track layouts (here "layout"), which is represented in a JSON object as follows:

8.3.1 LIF Structure

Object structure	Unit	Data type	Description
{		JSON-object	
metaInformation		JSON-object	Contains meta information.
layouts**[layout]**		array of JSON-object	Collection of layouts used in the facility by the driverless transport system. All layouts geometrically refer to the same project-specific global origin.
}			

The objects contained in this structure are described in more detail below.

8.3.2 MetaInformation

Object structure	Unit	Data type	Description
metaInformation {		JSON-object	Contains meta information.
projectIdentification		string	Human-readable name of the project (e.g., for display purposes).
creator		string	Creator of the LIF file (e.g., name of company, or name of person).
exportTimestamp		string	The timestamp at which this LIF file was created/updated/modified. Used to distinguish LIF file versions over time. Timestamp format is ISO8601 in UTC (YYYY-MM-DDTHH:mm:ss.ssZ, e.g., „2017-04-15T11:40:03.12Z").
lifVersion		string	Version of LIF: [Major].[Minor].[Patch] (0.11.0). **Note** This is the semantic version of the LIF format, as defined at the beginning of this document.
}			

8.3.3 Layout

Object structure	Unit	Data type	Description
layout {		JSON-object	A layout for order generation and routing. This layout holds relevant information independently from possible vehicles or (third-party) master control systems. It is intended to hold the information for all different vehicle types. Nodes and edges model a graph structure that is used as foundation for order generation and routing. A layout holds information that can be topologically considered a „plane“, i.e., multiple levels must be modelled in different layouts. It is also possible to partition the facility into multiple layouts even if the encoded information can be considered to lie on the same level. Each layout has the same origin of coordinates.
layoutId		string	Unique identifier for this layout.
layoutName		string	Human-readable name of the layout (e.g., for displaying).
layoutVersion		string	Version of the layout. **Note** It is suggested that this be an integer, represented as a string, incremented with each change, starting at „1“.
layoutLevelId		string	This attribute can be used to explicitly indicate which level or floor within a building or buildings a layout represents in a situation where there are multiple, such as multiple levels in the same facility, or two disconnected areas in the same facility.
layoutDescription		string	Brief description of the layout.
nodes**[node]**		array of JSON-object	Collection of all nodes in the layout.
edges**[edge]**		array of JSON-object	Collection of all edges in the layout.
stations**[station]**		array of JSON-object	Collection of all stations in the layout.
}			

8.3.4 Node

Object structure	Unit	Data type	Description
node {		JSON-object	Refers to VDA 5050 node definition. All properties that have the same name are meant to be semantically identical. However, the number of properties differs from VDA 5050 specification. Some properties are only meaningful as soon as an order is generated. Others only provide information for order generation (e.g., routing) itself.
nodeId		string	Unique identifier of the node across all layouts contained in this LIF file. **Note** Different LIF files, especially from different vehicle integrators, may contain duplicate nodeIds. In this case, it is the responsibility of the (third-party) master control system to track whichever internal unique nodeId it wishes to use, and to map this to a vehicle integrator‘s nodeId for its specific LIF.
nodeName		string	Name of the node. This should only be for visualization purposes. This attribute must not be used for any kind of identification or other logical purpose. Therefore, this node name need not necessarily be unique.
nodeDescription		string	Brief description of the node. This should only ever be for visualization or diagnostic purposes.
mapId		string	Unique identification of the map in which the node oder node‘s position is referenced. Each map has the same project specific global origin of coordinates. When a vehicle uses an elevator, e.g., leading from a departure floor to a target floor, it will disappear off the map of the departure floor and spawn in the related lift node on the map of the target floor.
nodePosition {		JSON-object	Geometric location of the node.
x	metre	float64	X position on the layout in reference to the global origin.
y	metre	float64	Y position on the layout in reference to the global origin.
}			
vehicleTypeNode Properties **[vehicleTypeNode Property]**		array of JSON-object	Vehicle type specific properties for this node. This attribute must not be empty. There must be an element for each vehicle type that may use this node. If no element exists for a particular vehicle type, the (third-party) master control system must consider that node invalid for use with that vehicle type.
}			

8.3.5 VehicleTypeNodeProperty

Object structure	Unit	Data type	Description
vehicleTypeNode Property {		JSON-object	
vehicleTypeId		string	Unique Id for type of vehicle to which these properties apply on this node. Only one vehicleTypeNodeProperty can be declared per vehicle type per node. **Note** It is suggested that this be a combination of [factsheet.manufacturer]. [factsheet.seriesName]
theta		float64	Range: [-Pi … Pi] Absolute orientation of the vehicle on the node in reference to the global origin's rotation.
*actions**[action]***		array of JSON-object	Holds actions that can be integrated into the order by the (third-party) master control system each time any vehicle with the corresponding vehicle TypeId is sent an order/order update that contains this node. The decision of which action is integrated into the order is the responsibility of the (third-party) master control system. If no actions can be integrated, the attribute may be omitted.
}			

8.3.6 Action

Object structure	Unit	Data type	Description
action {		JSON-object	Refers to VDA 5050 action definition. All properties that have the same name are meant to be semantically identical.
actionType		string	Name of the action same as described in the VDA 5050 specification document (section 6.8.2 in VDA 5050 2.0 specification document). **Note** Manufacturer-specific actions can be specified. Such actions must be agreed with the (third-party) master control system.
actionDescription		string	Brief description of the action.

8.3.6 Action continued

Object structure	Unit	Data type	Description
requirementType		string	Enum {REQUIRED, CONDITIONAL, OPTIONAL} „REQUIRED“ – The (third-party) master control system must always communicate this action to the vehicle on this node or edge. „CONDITIONAL“ – The action may or may not be required contingent upon various factors. Discussion between the vehicle integrator and the (third-party) master control system is required. „OPTIONAL“ - The action may or may not be communicated to the vehicle at the (third-party) master control system‘s discretion and responsibility. The vehicle must be able to execute without issue if OPTIONAL actions are never, sometimes, or always sent to it. **Note** The LIF does not specify a rigid definition of behaviour for anything other than at most one required action. If more than one action is marked as required on a node or edge, it is the responsibility of the vehicle integrator to define the implications of this to the (third-party) master control system, either be it that *all* of the required actions are always required, or that *one* of the actions is always required, or some other combination thereof.
blockingType		string	Enum {NONE, SOFT, HARD} „NONE“ - allows moving and other actions. „SOFT“ - allows other actions, but not moving. „HARD“ - is the only allowed action at this time.
actionParameters ***[actionParameter]***		array of JSON-object	Exact list of parameters and their statically defined values which must be sent along with this action. **Note** There may be other actionParameters with dynamic values that are required by an action that are not contained in this list. The (third-party) master control system must still determine and send these actionParameters. Refer to the AGV Fact Sheet.
}			

The AGV Fact Sheet may define actions that can be taken nearly anywhere, such as triggering a series of beeps or activating a light on the vehicle. These types of general actions may or may not be defined on (most or all) nodes and edges in the LIF. Such actions must be discussed between the vehicle integrator and the (third-party) master control system.

8.3.7 ActionParameter

Object structure	Unit	Data type	Description
actionParameter {		JSON-object	Key/value based generic action parameter listing.
key		string	Key which must be unique among the collection of action parameters.
value		string	Value corresponding to the key.
}			

8.3.8 Edge

Object structure	Unit	Data type	Description
edge {		JSON-object	Refers to VDA 5050 edge definition. All properties that have the same name are meant to be semantically identical. The LIF only contains edges that can be used by at least one vehicle type. Therefore, the LIF does not contain any edges that are blocked.
edgeId		string	Unique identifier of the edge across all layouts within this LIF file. **Note** Different LIF files, especially from different vehicle integrators, may contain duplicate edgeIds. In this case, it is the responsibility of the (third-party) master control system to track whichever internal unique edgeId it wishes to use, and to map this to a vehicle integrator‘s edgeId for its specific LIF.
edgeName		string	Name of the edge. This should only be for visualization purposes. This attribute must not be used for any kind of identification or other logical purpose.
edgeDescription		string	Brief description of the edge. This should only be used for visualization or diagnostic purposes.
startNodeId		string	Id of the start node. The start node must always be part of the current layout.
endNodeId		string	Id of the end node. The end node can be located in another layout. This models a transition from one layout to another.
vehicleTypeEdgeProperties **[vehicleTypeEdgeProperty]**		array of JSON-object	Vehicle type specific properties for this edge. **Note** This attribute must not be empty. For each allowed vehicle type there must be an element.
}			

8.3.9 VehicleTypeEdgeProperty

Object structure	Unit	Data type	Description
vehicleTypeEdge Property {		JSON-object	
vehicleTypeId		string	Unique Id for the type of vehicle to which these properties apply on this edge. Only one vehicleTypeEdge Property can be declared per vehicle type per edge. **Note** It is suggested that this be a combination of [factsheet.manufacturer]. [factsheet.seriesName]
vehicleOrientation	rad	float64	Orientation of the vehicle on the edge. The value *orientationType* defines if it has to be interpreted relative to the global project specific map coordinate system or tangential to the edge. In case of interpreted tangential to the edge 0.0 = forwards and Pi = backwards. Example: orientation Pi/2 rad will lead to a rotation of 90 degrees. If the vehicle starts in a different orientation, rotate the vehicle on the edge to the desired orientation if *rotationAllowed* is set to „true“. If *rotationAllowed* is „false“, rotate before entering the edge (assuming the start node allows rotation). If no trajectory is defined, apply the orientation to the direct path between the two connecting nodes of the edge. If a trajectory is defined for the edge, apply the orientation to the trajectory.
orientationType		string	Enum {GLOBAL, TANGENTIAL}: „GLOBAL“: relative to the global project specific map coordinate system. „TANGENTIAL“: tangential to the edge. **Note** If not defined, the default value is „TANGENTIAL“
rotationAllowed		boolean	„true“: rotation is allowed on the edge. The (third-party) master control system must assume that the vehicle will rotate in any direction along the edge at any point. The (third-party) master control system is responsible for avoiding issuing commands which will result in invalid or conflicting commands to other vehicles also under its control (e.g. deadlocks, potential collision). „false“: rotation is not allowed on the edge.
rotationAtStart NodeAllowed		string	Enum {NONE, CCW, CW, BOTH} Allowed directions of rotation for the vehicle at the start node. „NONE“ - Rotation not allowed. „CCW“ - Counter-clockwise (positive). „CW“ - Clockwise (negative). „BOTH“ - Both directions. **Note** If not defined, the default value is „BOTH“. See section 8.3.9.1 for detailed description.

8.3.9 VehicleTypeEdgeProperty continued

Object structure	Unit	Data type	Description
rotationAtEnd Node Allowed		string	Enum {NONE, CCW, CW, BOTH} Allowed directions of rotation for the vehicle at the end node. „NONE“ - Rotation not allowed. „CCW“ - Counter-clockwise (positive). „CW“ - Clockwise (negative). „BOTH“ - Both directions. **Note** If not defined, the default value is „BOTH“. See section 8.3.9.1 for detailed description.
maxSpeed	m/s	float64	Permitted maximum speed on the edge. Speed is defined by the fastest measurement of the vehicle. **Note** If not defined, no limitation.
maxRotationSpeed	rad/s	float64	Maximum rotation speed **Note** If not defined, no limitation.
minHeight	metre	float64	Permitted minimal height of the load handling device on the edge. **Note** If not defined, no limitation.
maxHeight	metre	float64	Permitted maximum height of the vehicle, including the load, on edge. **Note** If not defined, no limitation.
loadRestriction		JSON-object	Describes the load restriction on this edge for a vehicle of the corresponding vehicleTypeId. **Note** If not defined, the edge can be used by both an unloaded and loaded vehicle with the corresponding vehicleTypeId.
*actions**[action]***		array of JSON-object	Holds actions that can be integrated into the order by the (third-party) master control system each time any vehicle with the corresponding vehicleTypeId is sent an order/order update that contains this edge. **Note** If no actions must be integrated, the attribute can be omitted.

8.3.9 VehicleTypeEdgeProperty continued

Object structure	Unit	Data type	Description
trajectory		JSON-object	Trajectory JSON-object for this edge as a NURBS. Defines the curve on which the vehicle should move between startNode and endNode. Can be omitted if the vehicle cannot process trajectories or if the vehicle plans its own trajectory. **Note** The trajectory is not required, but if it is not provided, the (third-party) master control system may not be able to determine whether different vehicles from the same or a different manufacturer are colliding.
reentryAllowed		boolean	„true“: Vehicles of the corresponding vehicleTypeId are allowed to enter into automatic management by the (third-party) master control system while on this edge. „false“: Vehicles of the corresponding vehicleTypeId are not allowed to enter into automatic management by the (third-party) master control system while on this edge. **Note** If not defined, the default is „true“.
}			

8.3.9.1 Rotation Allowed at Start and End

Two attributes, rotationAtEndNodeAllowed and rotationAtStartNodeAllowed, may contradict one another if they terminate and originate, respectively, at the same node. In such cases, these should be combined as per a boolean and. As an example, if the end node rotation is BOTH on the terminating edge, but NONE on the originating edge, this would be interpreted as NONE.

For directional rotation values of CW or CCW, they must also align exactly, or values of CW or CCW on the terminating edge but BOTH on the originating edge would also only allow CW or CCW rotation, respectively. If these two attributes do not align at such a node, some edges of the layout may be unnavigable depending upon how the vehicle arrived at the node (which may or may not be intentional).

8.3.10 LoadRestriction

Object structure	Unit	Data type	Description
loadRestriction {		JSON-object	
unloaded		boolean	„true“: This edge may be used by an unloaded vehicle with the corresponding vehicleTypeId. „false“: This edge must not be used by an unloaded vehicle with the corresponding vehicleTypeId.
loaded		boolean	„true“: This edge may be used by a loaded vehicle with the corresponding vehicleTypeId. „false“: This edge must not be used by a loaded vehicle with the corresponding vehicleTypeId.
loadSetNames [string]		array of string	List of load sets that may be transported by the vehicle on this edge. The (third-party) master control system must evaluate this attribute only if the attribute loaded is set to true. **Note** If not defined or the attribute is empty, it means that all load sets supported by the vehicle loadSets are allowed. The same names for the load sets must be used here as are given in the factsheet of the respective vehicle (Factsheet attribute: [loadSets.setName]).
}			

8.3.11 Trajectory

Object structure	Unit	Data type	Description
trajectory {		JSON-object	
degree		integer	Range: [1.0 ... integer.max] Defines the number of control points that influence any given point on the curve. Increasing the degree increases continuity. If not defined, the default value is 1.
knotVector [float64]		array of float64	Range: [0.0 ... 1.0] Sequence of parameter values that determines where and how the control points affect the NURBS curve. knotVector has size of number of control points + degree + 1.
controlPoints **[controlPoint]**		array of JSON-object	List of JSON controlPoint JSON-objects defining the control points of the NURBS, which includes the beginning and end points.
}			

8.3.12 ControlPoint

Object structure	Unit	Data type	Description
controlPoint {		JSON-object	
x		float64	X position on the layout in reference to the global origin.
y		float64	Y position on the layout in reference to the global origin.
weight		float64	Range: [0.0 ... float64.max] The weight with which this control point pulls on the curve. When not defined, the default is 1.0.
}			

8.3.13 Station

Object structure	Unit	Data type	Description
station {		JSON-object	Every point where a vehicle can explicitly interact with the environment, including but not limited to physical interactions.
stationId		string	Unique identifier of the station across all layouts within this LIF file. **Note** It is recommended that stationIds match and align between all LIFs from all vehicle integrators and other load handling systems such as WMSs, as well as physical visual labelling and the like.
interaction NodeIds[string]		array of string	List of nodeIds for this station. These are the nodes that represent the position at which interaction with this station takes place. Multiple nodes can be listed for stations which can be accessed in multiple ways (such as stations that can be approached from multiple directions, e.g.: a station which can receive a EUR pallet longitudinally or laterally). This attribute must not be empty; there must be at least one nodeId. **Note** The decision of which nodeId is used is the responsibility of the (third-party) master control system. Choosing the correct interaction node may require that the (third-party) master control system considers the list of load sets defined on the edge or edges leading to the interaction node.
stationName		string	Human-readable name for the station (e.g., for displaying).
stationDescription		string	Brief description of the station.
stationHeight	metre	float64	Range: [0 … float64.max] Absolute physical height of the station. **Note** If not defined, the station height is 0.
stationPosition {			Centre point and orientation of the station. **Note** Only for visualization purposes.
x	metre	float64	X position of the station in the layout in reference to the global origin.
y	metre	float64	Y position of the station in the layout in reference to the global origin.
theta	radians	float64	Range: [-Pi … Pi] Absolute orientation of the station on the node.
}			
}			

8.3.13.1 How the (Third-party) Master Control System Can Identify the Purpose of a Station

If the (third-party) master control system would need to graphically identify certain stations, or would need to filter on a list of stations for human interaction purposes, the purpose of a station is entirely defined by the actions available on its interaction nodes. Every station that represents a charging area, for instance, should have a corresponding charging action, as defined in the AGV Fact Sheet, on its interaction node. Stations that can have multiple purposes, such as both emergency evacuation and maintenance, could be represented by two overlapping stations, or one station with multiple actions on one or more interaction nodes, or one combined action defined in the AGV Fact Sheet, and so forth.

8.4 Complete Data Structure of LIF

The complete data structure of LIF is shown below:

LIF File:

```
{
  "metaInformation":{
    "projectIdentification":"string",
    "creator":"string",
    "exportTimestamp":"string",
    "lifVersion":"string"
  },
  "layouts":[
    {
      "layoutId":"string",
      "layoutName":"string",
      "layoutVersion":"string",
      "layoutLevelId":"string",
      "layoutDescription":"string",
      "nodes":[
        {
          "nodeId":"string",
          "nodeName":"string",
          "nodeDescription":"string",
          "mapId":"string",
          "nodePosition":{
            "x":"number",
            "y":"number"
          },
          "vehicleTypeNodeProperties":[
            {
              "vehicleTypeId":"string",
              "theta":"number",
              "actions":[
                {
                  "actionType":"string",
                  "actionDescription":"string",
                  "required":"boolean",
                  "blockingType":"string",
              "actionParameters":[
                {
                  "key":"string",
                  "value":"string"
                }
              ]
            }
          ]
        }
      ]
    }
  ],
  "edges":[
    {
      "edgeId":"string",
      "edgeName":"string",
      "edgeDescription":"string",
      "startNodeId":"string",
      "endNodeId":"string",
      "vehicleTypeEdgeProperties":[
        {
          "vehicleTypeId":"string",
          "vehicleOrientation":"number",
          "orientationType":"string",
          "rotationAllowed":"boolean",
          "rotationAtStartNodeAllowed":"string",
          "rotationAtEndNodeAllowed":"string",
          "maxSpeed":"number",
          "maxRotationSpeed":"number",
          "minHeight":"number",
          "maxHeight":"number",
          "loadRestriction":{
            "unloaded":"boolean",
            "loaded":"boolean",
            "loadSetNames":[
              "string"
            ]
          },
          "actions":[
            {
              "actionType":"string",
              "actionDescription":"string",
              "requirementType":"string",
              "blockingType":"string",
              "actionParameters":[
                {
                  "key":"string",
                  "value":"string"
                }
              ]
            }
          ],
          "trajectory":{
            "degree":"number",
            "knotVector":[
                  "number"
                ],
                "controlPoints":[
                  {
                    "x":"number",
                    "y":"number",
                    "weight":"number"
                  }
                ]
              },
              "reentryAllowed":"boolean"
            }
          ]
        }
      ],
      "stations":[
        {
          "stationId":"string",
          "interactionNodeIds":[
            "string"
          ],
          "stationName":"string",
          "stationDescription":"string",
          "stationHeight":"number",
          "stationPosition ":{
            "x":"number",
            "y":"number",
            "theta":"number"
          }
        }
      ]
    }
  ]
}
```

9 Additional Information That Should Be Exchanged Uniformly

In addition to the reference to the VDA 5050 interface definition, information about geometry, kinematics, lifting systems, "capabilities of the vehicle", and so forth are included in the AGV Fact Sheet.

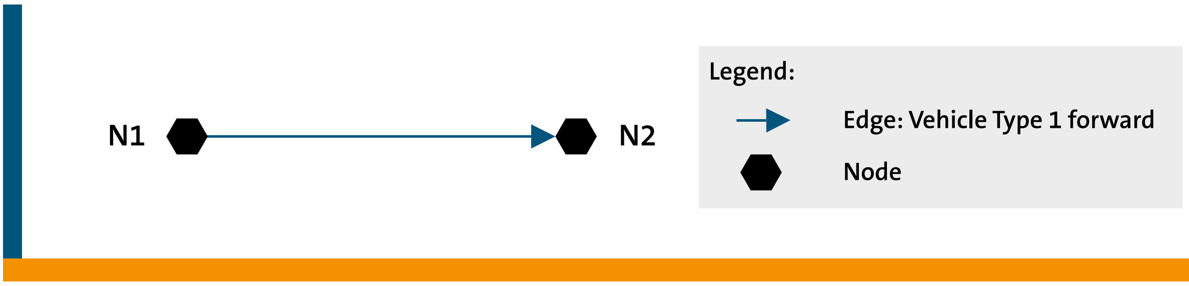
10 Examples

Note

The examples are kept simple, thus optional attributes (e.g. trajectory) are not defined for most.

10.1 Forward Edge

One edge (straight line) between two nodes. The vehicle may only move forward oriented on this edge.



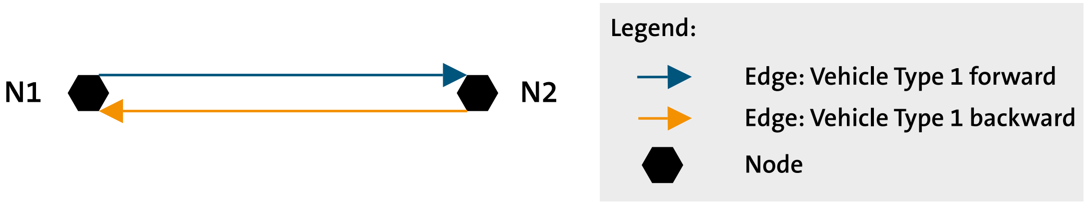
LIF File:

```
{
  "metaInformation":{
    "projectIdentification":"LIF Example 01",
    "creator":"VDMA",
    "exportTimestamp":"2023-09-28T10:00:00.00Z",
    "lifVersion":"1.0.0"
  },
  "layouts":[
    {
      "layoutId":"Layout_Ground_Level",
      "layoutName":"Name of Layout Ground Level",
      "layoutVersion":"1",
      "layoutDescription":"Ground level of Customer",
      "nodes":[
        {
          "nodeId":"N1",
          "mapId":"Map_Z-Level_1",
          "nodePosition":{
            "x":0.0,
            "y":0.0
          },
          "vehicleTypeNodeProperties":[
            {
              "vehicleTypeId":"Vehicle_Type_1"
            }
          ]
        },
        {
          "nodeId":"N2",
          "mapId":"Map_Z-Level_1",
          "nodePosition":{
            "x":11.0,
            "y":0.0
          },
          "vehicleTypeNodeProperties":[
            {
              "vehicleTypeId":"Vehicle_Type_1"
            }

          ]
        }
      ],
      "edges":[
        {
          "edgeId":"N1-N2",
          "startNodeId":"N1",
          "endNodeId":"N2",
          "vehicleTypeEdgeProperties":[
            {
              "vehicleTypeId":"Vehicle_Type_1",
              "vehicleOrientation":0.0,
              "orientationType":"TANGENTIAL",
              "rotationAllowed":false
            }
          ]
        }
      ]
    }
  ]
}
```

10.2 Bidirectional Edge

Two edges (straight line) between two nodes. The vehicle may only move forward oriented on one edge and backward oriented on the other edge.

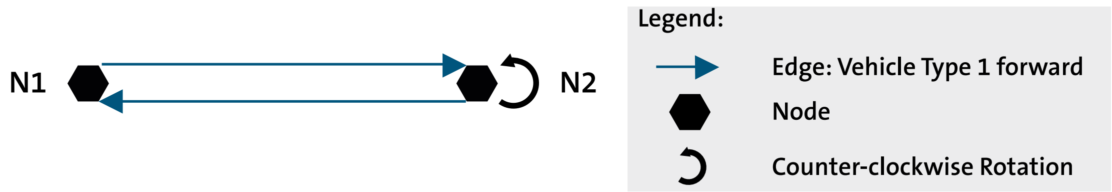


LIF File:

```
{
  "metaInformation":{
    "projectIdentification":"LIF Example 02",
    "creator":"VDMA",
    "exportTimestamp":"2023-09-28T10:00:00.00Z",
    "lifVersion":"1.0.0"
  },
  "layouts":[
    {
      "layoutId":"Layout_Ground_Level",
      "layoutName":"Name of Layout Ground Level",
      "layoutVersion":"1",
      "layoutDescription":"Ground level of Customer",
      "nodes":[
        {
          "nodeId":"N1",
          "mapId":"Map_Z-Level_1",
          "nodePosition":{
            "x":0.0,
            "y":0.0
          },
          "vehicleTypeNodeProperties":[
            {
              "vehicleTypeId":"Vehicle_Type_1"
            }
          ]
        },
       {
         "nodeId":"N2",
         "mapId":"Map_Z-Level_1",
         "nodePosition":{
           "x":11.0,
           "y":0.0
         },
         "vehicleTypeNodeProperties":[
           {
             "vehicleTypeId":"Vehicle_Type_1"
           }
         ]
       }
     ],
     "edges":[
       {
         "edgeId":"N1-N2",
         "startNodeId":"N1",
         "endNodeId":"N2",
         "vehicleTypeEdgeProperties":[
           {
             "vehicleTypeId":"Vehicle_Type_1",
             "vehicleOrientation":0.0,
             "orientationType":"TANGENTIAL",
             "rotationAllowed":false
           }
         ]
       },
       {
         "edgeId":"N2-N1",
         "startNodeId":"N2",
         "endNodeId":"N1",
         "vehicleTypeEdgeProperties":[
           {
             "vehicleTypeId":"Vehicle_Type_1",
             "vehicleOrientation":3.1415926535897931,
             "orientationType":"TANGENTIAL",
             "rotationAllowed":false
           }
         ]
       }
     ]
   }
  ]
}
```

10.3 Counter-clockwise Rotation on Node

Two edges (straight lines) between two nodes. The vehicle may only move forward oriented on both edges, rotation counter-clockwise allowed at node N1.

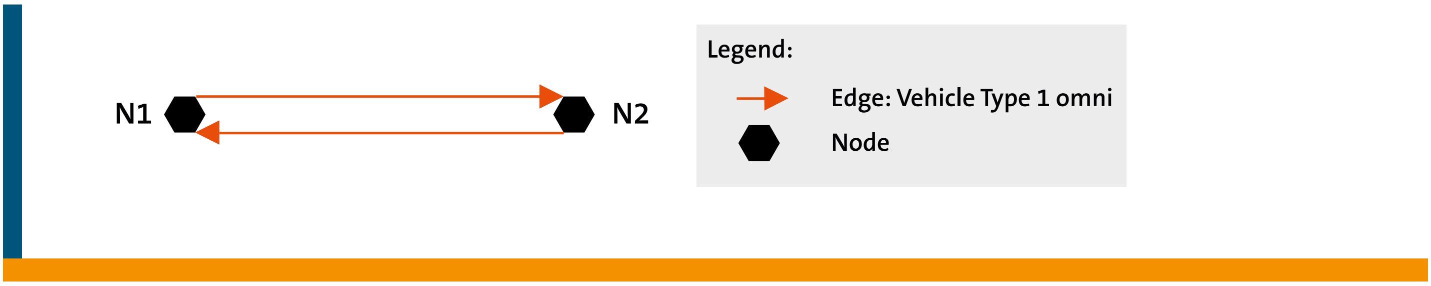


LIF File:

```
{
  "metaInformation":{
    "projectIdentification":"LIF Example 03",
    "creator":"VDMA",
    "exportTimestamp":"2023-09-28T10:00:00.00Z",
    "lifVersion":"1.0.0"
  },
  "layouts":[
    {
      "layoutId":"Layout_Ground_Level",
      "layoutName":"Name of Layout Ground Level",
      "layoutVersion":"1",
      "layoutDescription":"Ground level of Customer",
      "nodes":[
        {
          "nodeId":"N1",
          "mapId":"Map_Z-Level_1",
          "nodePosition":{
            "x":0.0,
            "y":0.0
          },
          "vehicleTypeNodeProperties":[
            {
              "vehicleTypeId":"Vehicle_Type_1"
            }
          ]
        },
        {
          "nodeId":"N2",
          "mapId":"Map_Z-Level_1",
          "nodePosition":{
            "x":11.0,
            "y":0.0
          },
          "vehicleTypeNodeProperties":[
            {
              "vehicleTypeId":"Vehicle_Type_1"
            }
          ]
        }
      ],
      "edges":[
        {
          "edgeId":"N1-N2",
          "startNodeId":"N1",
          "endNodeId":"N2",
          "vehicleTypeEdgeProperties":[
            {
              "vehicleTypeId":"Vehicle_Type_1",
              "vehicleOrientation":0.0,
              "orientationType":"TANGENTIAL",
              "rotationAllowed":false,
              "rotationAtStartNodeAllowed":"NONE",
              "rotationAtEndNodeAllowed":"CCW"
            }
          ]
        },
        {
          "edgeId":"N2-N1",
          "startNodeId":"N2",
          "endNodeId":"N1",
          "vehicleTypeEdgeProperties":[
            {
              "vehicleTypeId":"Vehicle_Type_1",
              "vehicleOrientation":0.0,
              "orientationType":"TANGENTIAL",
              "rotationAllowed":false,
              "rotationAtStartNodeAllowed":"CCW",
              "rotationAtEndNodeAllowed":"NONE"
            }
          ]
        }
      ]
    }
  ]
}
```

10.4 Omnidirectional Edge

Two edges (straight lines) between two nodes. The vehicle moves omnidirectionally to 90° on the edge from N1 to N2 and the vehicle moves omnidirectionally back to -90° on the edge from N2 back to N1.

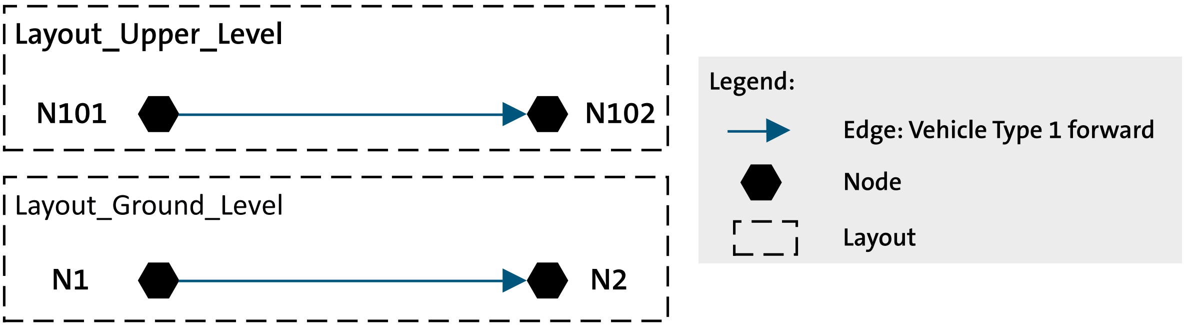


LIF File:

```
{
  "metaInformation":{
    "projectIdentification":"LIF Example 04",
    "creator":"VDMA",
    "exportTimestamp":"2023-09-28T10:00:00.00Z",
    "lifVersion":"1.0.0"
  },
  "layouts":[
    {
      "layoutId":"Layout_Ground_Level",
      "layoutName":"Name of Layout Ground Level",
      "layoutVersion":"1",
      "layoutDescription":"Ground level of Customer",
      "nodes":[
        {
          "nodeId":"N1",
          "mapId":"Map_Z-Level_1",
          "nodePosition":{
            "x":0.0,
            "y":0.0
          },
          "vehicleTypeNodeProperties":[
            {
              "vehicleTypeId":"Vehicle_Type_1"
            }
          ]
        },
        {
          "nodeId":"N2",
          "mapId":"Map_Z-Level_1",
          "nodePosition":{
            "x":11.0,
            "y":0.0
          },
          "vehicleTypeNodeProperties":[
            {
              "vehicleTypeId":"Vehicle_Type_1"
            }
          ]
        }
      ],
      "edges":[
        {
          "edgeId":"N1-N2",
          "startNodeId":"N1",
          "endNodeId":"N2",
          "vehicleTypeEdgeProperties":[
            {
              "vehicleTypeId":"Vehicle_Type_1",
              "vehicleOrientation":1.5707963267948966,
              "orientationType":"TANGENTIAL",
              "rotationAllowed":true,
              "rotationAtStartNodeAllowed":"NONE",
              "rotationAtEndNodeAllowed":"NONE"
            }
          ]
        },
        {
          "edgeId":"N2-N1",
          "startNodeId":"N2",
          "endNodeId":"N1",
          "vehicleTypeEdgeProperties":[
            {
              "vehicleTypeId":"Vehicle_Type_1",
              "vehicleOrientation":-1.5707963267948966,
              "orientationType":"TANGENTIAL",
              "rotationAllowed":true,
              "rotationAtStartNodeAllowed":"NONE",
              "rotationAtEndNodeAllowed":"NONE"
            }
          ]
        }
      ]
    }
  ]
}
```

10.5 Multiple Layouts in One LIF File

Two layouts in one LIF file, representing two different levels of the facility.

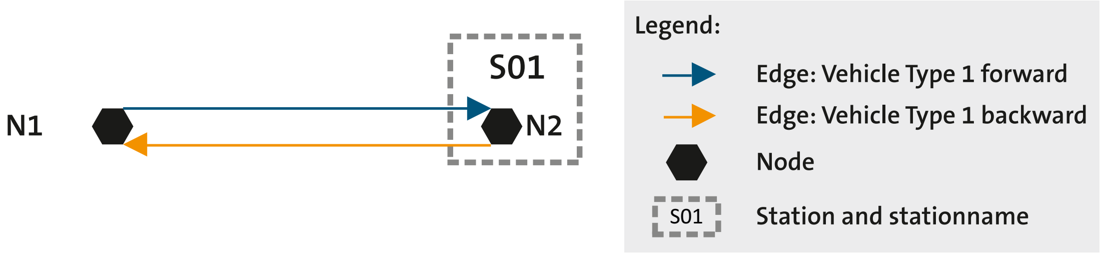


LIF File:

```
{
  "metaInformation":{
    "projectIdentification":"LIF Example 05",
    "creator":"VDMA",
    "exportTimestamp":"2023-09-28T10:00:00.00Z",
    "lifVersion":"1.0.0"
  },
  "layouts":[
    {
      "layoutId":"Layout_Ground_Level",
      "layoutName":"Name of Layout Ground Level",
      "layoutVersion":"1",
      "layoutLevelId":"0",
      "layoutDescription":"Ground level of Customer",
      "nodes":[
        {
          "nodeId":"N1",
          "mapId":"Map_Z-Level_1",
          "nodePosition":{
            "x":0.0,
            "y":0.0
          },
          "vehicleTypeNodeProperties":[
            {
              "vehicleTypeId":"Vehicle_Type_1"
            }
          ]
        },
        {
          "nodeId":"N2",
          "mapId":"Map_Z-Level_1",
          "nodePosition":{
            "x":11.0,
            "y":0.0
          },
          "vehicleTypeNodeProperties":[
            {
              "vehicleTypeId":"Vehicle_Type_1"
            }
          ]
        }
      ],
      "edges":[
        {
          "edgeId":"N1-N2",
          "startNodeId":"N1",
          "endNodeId":"N2",
          "vehicleTypeEdgeProperties":[
            {
              "vehicleTypeId":"Vehicle_Type_1",
              "vehicleOrientation":0.0,
              "orientationType":"TANGENTIAL",
              "rotationAllowed":false
            }
          ]
        }
      ]
    },
    {
      "layoutId":"Layout_Upper_Level",
      "layoutName":"Name of Layout Upper Level",
      "layoutVersion":"1",
      "layoutLevelId":"1",
      "layoutDescription":"Upper level of Customer",
      "nodes":[
        {
          "nodeId":"N101",
          "mapId":"Map_Z-Level_2",
          "nodePosition":{
            "x":12.4,
            "y":3.4
          },
          "vehicleTypeNodeProperties":[
            {
              "vehicleTypeId":"Vehicle_Type_1"
            }
          ]
        },
        {
          "nodeId":"N102",
          "mapId":"Map_Z-Level_2",
          "nodePosition":{
            "x":12.0,
            "y":3.4
          },
          "vehicleTypeNodeProperties":[
            {
              "vehicleTypeId":"Vehicle_Type_1"
            }
          ]
        }
      ],
      "edges":[
        {
          "edgeId":"N101-N102",
          "startNodeId":"N101",
          "endNodeId":"N102",
          "vehicleTypeEdgeProperties":[
            {
              "vehicleTypeId":"Vehicle_Type_1",
              "vehicleOrientation":0.0,
              "orientationType":"TANGENTIAL",
              "rotationAllowed":false
            }
          ]
        }
      ]
    }
  ]
}
```

10.6 Station with One Node

Two edges (straight lines) between two nodes. At one node there is a station for picking up pallets.
The vehicle may only move forward oriented on one edge and backward oriented on the other edge.

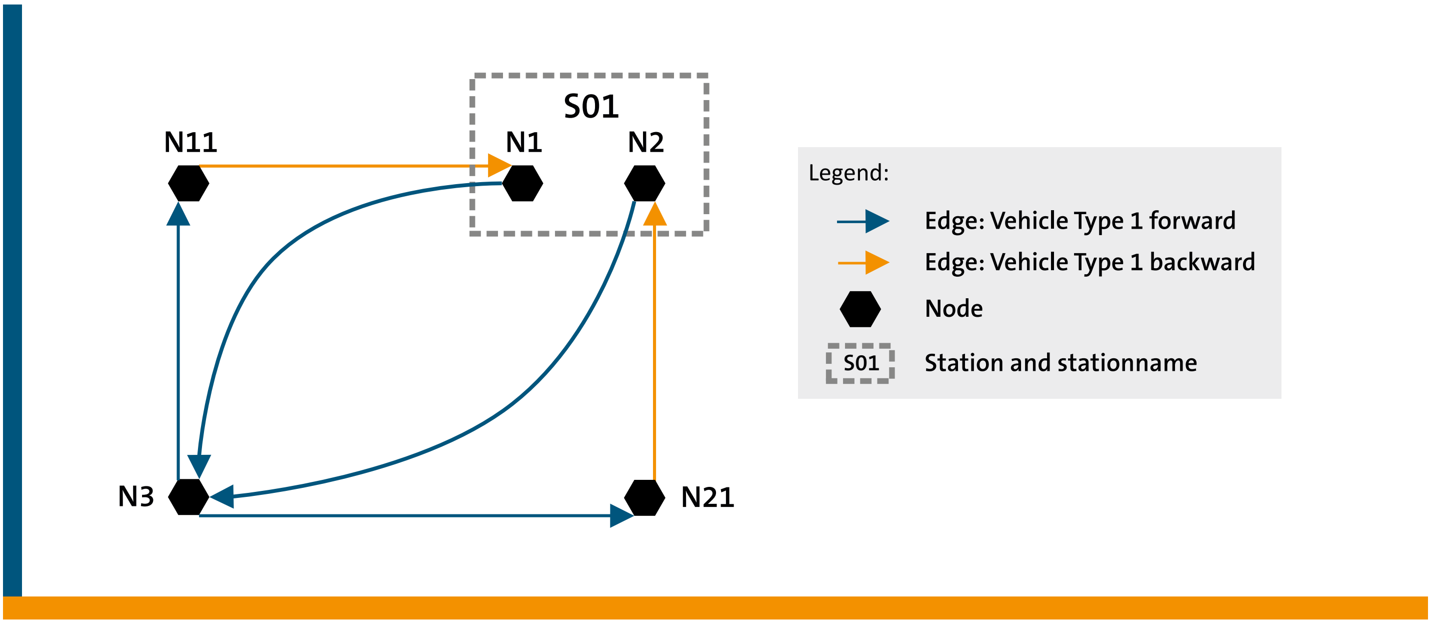


LIF File:

```
{
  "metaInformation":{
    "projectIdentification":"LIF Example 06",
    "creator":"VDMA",
    "exportTimestamp":"2023-09-28T10:00:00.00Z",
    "lifVersion":"1.0.0"
  },
  "layouts":[
    {
      "layoutId":"Layout_Ground_Level",
      "layoutName":"Name of Layout Ground Level",
      "layoutVersion":"1",
      "layoutDescription":"Ground level of Customer",
      "nodes":[
        {
          "nodeId":"N1",
          "mapId":"Map_Z-Level_1",
          "nodePosition":{
            "x":0.0,
            "y":0.0
          },
          "vehicleTypeNodeProperties":[
            {
              "vehicleTypeId":"Vehicle_Type_1"
            }
          ]
        },
    {
      "nodeId":"N2",
      "mapId":"Map_Z-Level_1",
      "nodePosition":{
        "x":11.0,
        "y":0.0
      },
      "vehicleTypeNodeProperties":[
        {
          "vehicleTypeId":"Vehicle_Type_1",
          "actions":[
            {
              "actionType":"pick",
              "requirementType":"CONDITIONAL",
              "blockingType":"HARD",
              "actionParameters":[
                {
                  "key":"loadType",
                  "value":"Example load type"
                }
              ]
            }
          ]
        }
      ]
    }
  ],
  "edges":[
    {
      "edgeId":"N1-N2",
      "startNodeId":"N1",
      "endNodeId":"N2",
      "vehicleTypeEdgeProperties":[
        {
          "vehicleTypeId":"Vehicle_Type_1",
          "vehicleOrientation":3.1415926535897931,
          "orientationType":"TANGENTIAL",
          "rotationAllowed":false
        }
      ]
    },
        {
          "edgeId":"N2-N1",
          "startNodeId":"N2",
          "endNodeId":"N1",
          "vehicleTypeEdgeProperties":[
            {
              "vehicleTypeId":"Vehicle_Type_1",
              "vehicleOrientation":0.0,
              "orientationType":"TANGENTIAL",
              "rotationAllowed":false
            }
          ]
        }
      ],
      "stations":[
        {
          "stationId":"S01",
          "interactionNodeIds":[
            "N2"
          ],
          "stationName":"SOURCE_01",
          "stationDescription":"Source to pick up pallet",
          "stationHeight":"0.55"
        }
      ]
    }
  ]
}
```

10.7 Station with Two Nodes

Modelling a station with two different nodes (e.g. rotation station for a pallet).



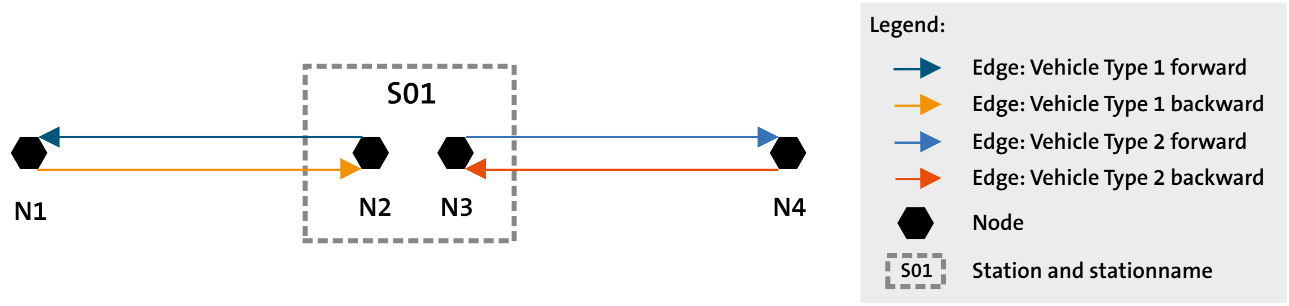
LIF File:

```
{
  "metaInformation":{
    "projectIdentification":"LIF Example 07",
    "creator":"VDMA",
    "exportTimestamp":"2023-09-28T10:00:00.00Z",
    "lifVersion":"1.0.0"
  },
  "layouts":[
    {
      "layoutId":"Layout_Ground_Level",
      "layoutName":"Name of Layout Ground Level",
      "layoutVersion":"1",
      "layoutDescription":"Ground level of Customer",
      "nodes":[
        {
          "nodeId":"N1",
          "mapId":"Map_Z-Level_1",
          "nodePosition":{
            "x":9.2,
            "y":3.4
          },
  "vehicleTypeNodeProperties":[
    {
      "vehicleTypeId":"Vehicle_Type_1",
      "actions":[
        {
          "actionType":"pick",
          "requirementType":"CONDITIONAL",
          "blockingType":"HARD",
          "actionParameters":[
            {
              "key":"loadType",
              "value":"Example load type"
            }
          ]
        },
        {
          "actionType":"drop",
          "requirementType":"CONDITIONAL",
          "blockingType":"HARD",
          "actionParameters":[
            {
              "key":"loadType",
              "value":"Example load type"
            }
          ]
        }
      ]
    }
  ]
},
{
  "nodeId":"N2",
  "mapId":"Map_Z-Level_1",
  "nodePosition":{
    "x":9.4,
    "y":3.2
  },
  "vehicleTypeNodeProperties":[
    {
      "vehicleTypeId":"Vehicle_Type_1",
      "actions":[
        {
          "actionType":"pick",
          "requirementType":"CONDITIONAL",
          "blockingType":"HARD"
        },
          {
            "actionType":"drop",
            "requirementType":"CONDITIONAL",
            "blockingType":"HARD"
          }
        ]
      }
    ]
  },
  {
    "nodeId":"N3",
    "mapId":"Map_Z-Level_1",
    "nodePosition":{
      "x":0.0,
      "y":0.0
    },
    "vehicleTypeNodeProperties":[
      {
        "vehicleTypeId":"Vehicle_Type_1"
      }
    ]
  },
  {
    "nodeId":"N11",
    "mapId":"Map_Z-Level_1",
    "nodePosition":{
      "x":0.0,
      "y":3.4
    },
    "vehicleTypeNodeProperties":[
      {
        "vehicleTypeId":"Vehicle_Type_1"
      }
    ]
  },
  {
    "nodeId":"N21",
    "mapId":"Map_Z-Level_1",
    "nodePosition":{
      "x":9.2,
      "y":0.0
    },
    "vehicleTypeNodeProperties":[
      {
        "vehicleTypeId":"Vehicle_Type_1"
      }
    ]
  }
],
"edges":[
  {
    "edgeId":"N11-N1",
    "startNodeId":"N11",
    "endNodeId":"N1",
    "vehicleTypeEdgeProperties":[
      {
        "vehicleTypeId":"Vehicle_Type_1",
        "vehicleOrientation":3.1415926535897931,
        "orientationType":"TANGENTIAL",
        "rotationAllowed":false
      }
    ]
  },
  {
    "edgeId":"N3-N11",
    "startNodeId":"N3",
    "endNodeId":"N11",
    "vehicleTypeEdgeProperties":[
      {
        "vehicleTypeId":"Vehicle_Type_1",
        "vehicleOrientation":0.0,
        "orientationType":"TANGENTIAL",
        "rotationAllowed":false
      }
    ]
  },
  {
    "edgeId":"N1-N3",
    "startNodeId":"N1",
    "endNodeId":"N3",
    "vehicleTypeEdgeProperties":[
      {
        "vehicleTypeId":"Vehicle_Type_1",
        "vehicleOrientation":0.0,
        "orientationType":"TANGENTIAL",
        "rotationAllowed":false
      }
    ]
  },
  {
    "edgeId":"N3-N21",
    "startNodeId":"N3",
    "endNodeId":"N21",
    "vehicleTypeEdgeProperties":[
      {
        "vehicleTypeId":"Vehicle_Type_1",
        "vehicleOrientation":0.0,
        "orientationType":"TANGENTIAL",
        "rotationAllowed":false
      }
```

```
          ]
        },
        {
          "edgeId":"N21-N2",
          "startNodeId":"N21",
          "endNodeId":"N2",
          "vehicleTypeEdgeProperties":[
            {
              "vehicleTypeId":"Vehicle_Type_1",
              "vehicleOrientation":3.1415926535897931,
              "orientationType":"TANGENTIAL",
              "rotationAllowed":false
            }
          ]
        },
        {
          "edgeId":"N2-N3",
          "startNodeId":"N2",
          "endNodeId":"N3",
          "vehicleTypeEdgeProperties":[
            {
              "vehicleTypeId":"Vehicle_Type_1",
              "vehicleOrientation":0.0,
              "orientationType":"TANGENTIAL",
              "rotationAllowed":false
            }
          ]
        }
      ],
      "stations":[
        {
          "stationId":"S01",
          "interactionNodeIds":[
            "N1",
            "N2"
          ],
          "stationName":"SOURCE_01",
          "stationDescription":"Pallet rotation station",
          "stationHeight":"0.55"
        }
      ]
    }
  ]
}
```

10.8 Station with Two Nodes, Restricted for Different Vehicle Types

Station with two nodes but restricted for different vehicle types.


S01
N1
N2
N3
N4
Legend:
Edge: Vehicle Type 1 forward
Edge: Vehicle Type 1 backward
Edge: Vehicle Type 2 forward
Edge: Vehicle Type 2 backward
Node
S01
Station and stationname

LIF File:

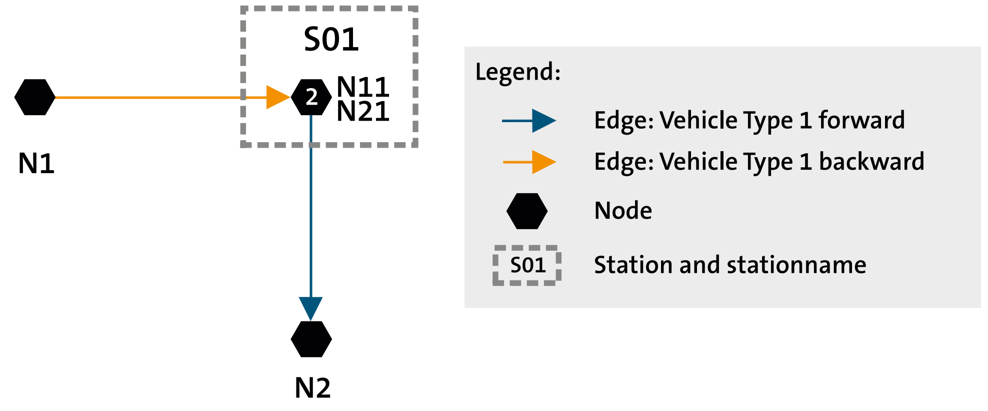
```
{
  "metaInformation":{
    "projectIdentification":"LIF Example 08",
    "creator":"VDMA",
    "exportTimestamp":"2023-09-28T10:00:00.00Z",
    "lifVersion":"1.0.0"
  },
  "layouts":[
    {
      "layoutId":"Layout_Ground_Level",
      "layoutName":"Name of Layout Ground Level",
      "layoutVersion":"1",
      "layoutDescription":"Ground level of Customer",
      "nodes":[
        {
          "nodeId":"N1",
          "mapId":"Map_Z-Level_1",
          "nodePosition":{
            "x":7.2,
            "y":0.0
          },
          "vehicleTypeNodeProperties":[
            {
```

```
      "vehicleTypeId":"Vehicle_Type_1"
    }
  ]
},
{
  "nodeId":"N2",
  "mapId":"Map_Z-Level_1",
  "nodePosition":{
    "x":9.2,
    "y":0.0
  },
  "vehicleTypeNodeProperties":[
    {
      "vehicleTypeId":"Vehicle_Type_1",
      "actions":[
        {
          "actionType":"drop",
          "requirementType":"CONDITIONAL",
          "blockingType":"HARD",
          "actionParameters":[
            {
              "key":"loadType",
              "value":"Example load type"
            }
          ]
        }
      ]
    }
  ]
},
{
  "nodeId":"N3",
  "mapId":"Map_Z-Level_1",
  "nodePosition":{
    "x":9.6,
    "y":0.0
  },
  "vehicleTypeNodeProperties":[
    {
      "vehicleTypeId":"Vehicle_Type_2",
      "actions":[
        {
          "actionType":"pick",
          "requirementType":"CONDITIONAL",
          "blockingType":"HARD",
          "actionParameters":[
            {
              "key":"loadType",
              "value":"Example load type"
            }
```

```
              ]
            }
          ]
        }
      ]
    },
    {
      "nodeId":"N4",
      "mapId":"Map_Z-Level_1",
      "nodePosition":{
        "x":14.8,
        "y":3.4
      },
      "vehicleTypeNodeProperties":[
        {
          "vehicleTypeId":"Vehicle_Type_2"
        }
      ]
    }
  ],
  "edges":[
    {
      "edgeId":"N1-N2",
      "startNodeId":"N1",
      "endNodeId":"N2",
      "vehicleTypeEdgeProperties":[
        {
          "vehicleTypeId":"Vehicle_Type_1",
          "vehicleOrientation":3.1415926535897931,
          "orientationType":"TANGENTIAL",
          "rotationAllowed":false
        }
      ]
    },
    {
      "edgeId":"N2-N1",
      "startNodeId":"N2",
      "endNodeId":"N1",
      "vehicleTypeEdgeProperties":[
        {
          "vehicleTypeId":"Vehicle_Type_1",
          "vehicleOrientation":0.0,
          "orientationType":"TANGENTIAL",
          "rotationAllowed":false
        }
      ]
    },
        {
          "edgeId":"N4-N3",
          "startNodeId":"N4",
          "endNodeId":"N3",
          "vehicleTypeEdgeProperties":[
            {
              "vehicleTypeId":"Vehicle_Type_2",
              "vehicleOrientation":3.1415926535897931,
              "orientationType":"TANGENTIAL",
              "rotationAllowed":false
            }
          ]
        },
        {
          "edgeId":"N3-N4",
          "startNodeId":"N3",
          "endNodeId":"N4",
          "vehicleTypeEdgeProperties":[
            {
              "vehicleTypeId":"Vehicle_Type_2",
              "vehicleOrientation":0.0,
              "orientationType":"TANGENTIAL",
              "rotationAllowed":false
            }
          ]
        }
      ],
      "stations":[
        {
          "stationId":"S01",
          "interactionNodeIds":[
            "N2",
            "N3"
          ],
          "stationName":"SOURCE_01",
          "stationDescription":"Handover station for pallet",
          "stationHeight":"0.0"
        }
      ]
    }
  ]
}
```

10.9 Rotation Station

Rotation station for pallet, on which a rectangular load can be dropped “short side leading” and then picked up “long side leading”.



LIF File:

```
{
  "metaInformation":{
    "projectIdentification":"LIF Example 09",
    "creator":"VDMA",
    "exportTimestamp":"2023-09-28T10:00:00.00Z",
    "lifVersion":"1.0.0"
  },
  "layouts":[
    {
      "layoutId":"Layout_Ground_Level",
      "layoutName":"Name of Layout Ground Level",
      "layoutVersion":"1",
      "layoutDescription":"Ground level of Customer",
      "nodes":[
        {
          "nodeId":"N1",
          "mapId":"Map_Z-Level_1",
          "nodePosition":{
            "x":7.2,
            "y":0.0
          },
  "vehicleTypeNodeProperties":[
    {
      "vehicleTypeId":"Vehicle_Type_1"
    }
  ]
},
{
  "nodeId":"N11",
  "mapId":"Map_Z-Level_1",
  "nodePosition":{
    "x":9.2,
    "y":0.0
  },
  "vehicleTypeNodeProperties":[
    {
      "vehicleTypeId":"Vehicle_Type_1",
      "actions":[
        {
          "actionType":"drop",
          "requirementType":"CONDITIONAL",
          "blockingType":"HARD"
        }
      ]
    }
  ]
},
{
  "nodeId":"N21",
  "mapId":"Map_Z-Level_1",
  "nodePosition":{
    "x":9.2,
    "y":0.0
  },
  "vehicleTypeNodeProperties":[
    {
      "vehicleTypeId":"Vehicle_Type_1",
        "theta":-1.5707963268,
        "actions":[
          {
            "actionType":"pick",
            "requirementType":"CONDITIONAL",
            "blockingType":"HARD",
            "actionParameters":[
              {
                "key":"loadType",
                "value":"Example load type"
              }
            ]
          }
        ]
      }
    ]
  },
  {
    "nodeId":"N2",
    "mapId":"Map_Z-Level_1",
    "nodePosition":{
      "x":9.2,
      "y":-5.0
    },
    "vehicleTypeNodeProperties":[
      {
        "vehicleTypeId":"Vehicle_Type_1"
      }
    ]
  }
],
"edges":[
  {
    "edgeId":"N1-N11",
    "startNodeId":"N1",
    "endNodeId":"N11",
    "vehicleTypeEdgeProperties":[
      {
        "vehicleTypeId":"Vehicle_Type_1",
        "vehicleOrientation":3.1415926535897931,
        "orientationType":"TANGENTIAL",
        "rotationAllowed":false
      }
    ]
  },
        {
          "edgeId":"N11-N21",
          "startNodeId":"N11",
          "endNodeId":"N21",
          "vehicleTypeEdgeProperties":[
            {
              "vehicleTypeId":"Vehicle_Type_1",
              "rotationAllowed":false,
              "rotationAtStartNodeAllowed":"CW"
            }
          ]
        },
        {
          "edgeId":"N21-N2",
          "startNodeId":"N21",
          "endNodeId":"N2",
          "vehicleTypeEdgeProperties":[
            {
              "vehicleTypeId":"Vehicle_Type_1",
              "vehicleOrientation":0.0,
              "orientationType":"TANGENTIAL",
              "rotationAllowed":false
            }
          ]
        }
      ],
      "stations":[
        {
          "stationId":"S01",
          "interactionNodeIds":[
            "N11",
            "N21"
          ],
          "stationName":"Rotation_01",
          "stationDescription":"Rotation station for palle
          "stationHeight":"0.75"
        }
      ]
    }
  ]
}
```

10.10 Station with Three Nodes, Restricted to Different Vehicle Types

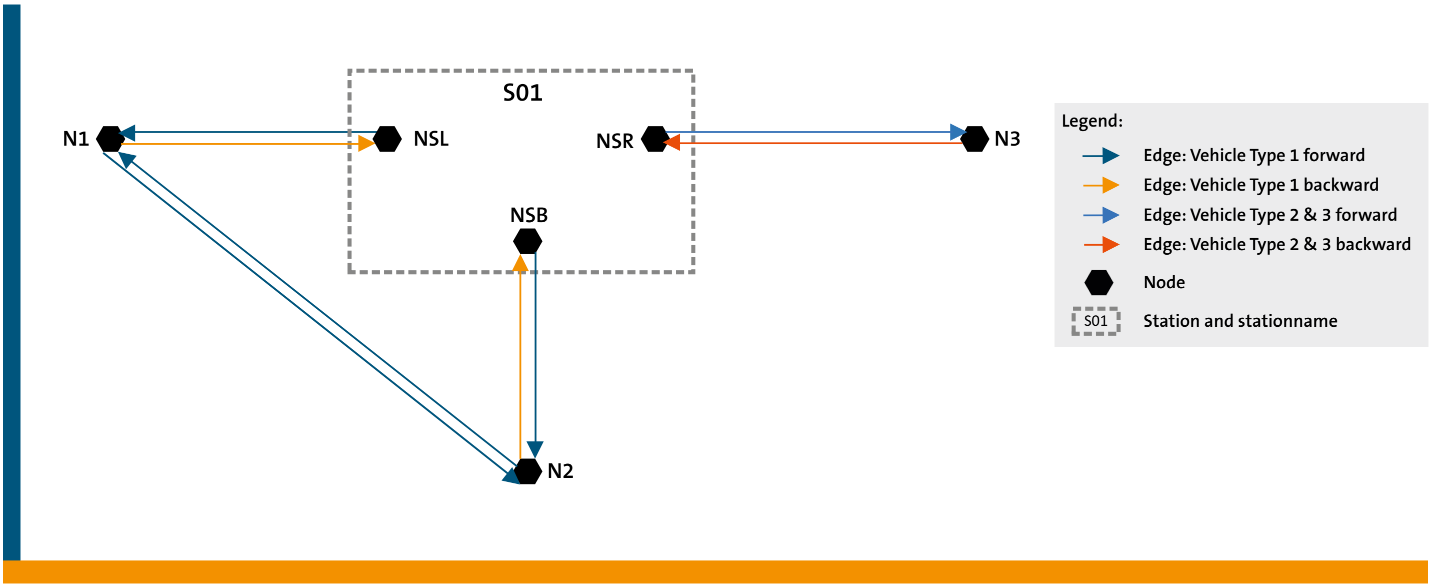
One station with three nodes, but restricted to different vehicle types.

Restriction on edges

- Vehicle Type 1 Forward
- Vehicle Type 2 Backward
- Vehicle Type 2 & 3 Forward
- Vehicle Type 2 & 3 Backward

Explanation

- NSL: Vehicle Type 1 pick and drop
- NSB: Vehicle Type 1 pick
- NSR: Vehicle Type 2 drop
- NSR: Vehicle Type 3 pick and drop


S01
N1
NSL
NSR
N3
NSB
N2
Legend:
Edge: Vehicle Type 1 forward
Edge: Vehicle Type 1 backward
Edge: Vehicle Type 2 & 3 forward
Edge: Vehicle Type 2 & 3 backward
Node
S01
Station and stationname

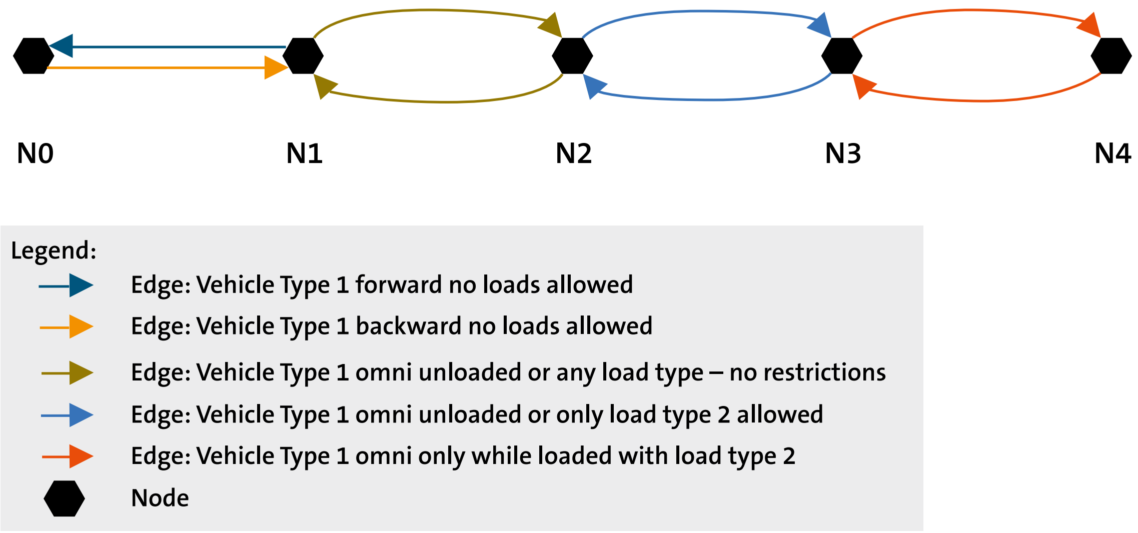
LIF File:

```
{
  "metaInformation":{
    "projectIdentification":"LIF Example 10",
    "creator":"VDMA",
    "exportTimestamp":"2023-09-28T10:00:00.00Z",
    "lifVersion":"1.0.0"
  },
  "layouts":[
    {
      "layoutId":"Layout_Ground_Level",
      "layoutName":"Name of Layout Ground Level",
      "layoutVersion":"1",
      "layoutDescription":"Ground level of Customer",
      "nodes":[
        {
```

```
      "nodeId":"N1",
      "mapId":"Map_Z-Level_1",
      "nodePosition":{
        "x":7.2,
        "y":0.0
      },
      "vehicleTypeNodeProperties":[
        {
          "vehicleTypeId":"Vehicle_Type_1"
        }
      ]
    },
    {
      "nodeId":"NSL",
      "mapId":"Map_Z-Level_1",
      "nodePosition":{
        "x":9.2,
        "y":0.0
      },
      "vehicleTypeNodeProperties":[
        {
          "vehicleTypeId":"Vehicle_Type_1",
          "actions":[
            {
              "actionType":"pick",
              "requirementType":"CONDITIONAL",
              "blockingType":"HARD"
            },
            {
              "actionType":"drop",
              "requirementType":"CONDITIONAL",
              "blockingType":"HARD"
            }
          ]
        }
      ]
    },
    {
      "nodeId":"NSR",
      "mapId":"Map_Z-Level_1",
      "nodePosition":{
        "x":10.2,
        "y":0.0
      },
  "vehicleTypeNodeProperties":[
    {
      "vehicleTypeId":"Vehicle_Type_2",
      "actions":[
        {
          "actionType":"drop",
          "requirementType":"CONDITIONAL",
          "blockingType":"HARD"
        }
      ]
    },
    {
      "vehicleTypeId":"Vehicle_Type_3",
      "actions":[
        {
          "actionType":"pick",
          "requirementType":"CONDITIONAL",
          "blockingType":"HARD"
        },
        {
          "actionType":"drop",
          "requirementType":"CONDITIONAL",
          "blockingType":"HARD"
        }
      ]
    }
  ]
},
{
  "nodeId":"NSB",
  "mapId":"Map_Z-Level_1",
  "nodePosition":{
    "x":9.7,
    "y":-0.5
  },
  "vehicleTypeNodeProperties":[
    {
      "vehicleTypeId":"Vehicle_Type_1",
      "actions":[
        {
          "actionType":"pick",
          "requirementType":"CONDITIONAL",
          "blockingType":"HARD"
        }
      ]
    }
  ]
},
    {
      "nodeId":"N2",
      "mapId":"Map_Z-Level_1",
      "nodePosition":{
        "x":9.7,
        "y":-5.0
      },
      "vehicleTypeNodeProperties":[
        {
          "vehicleTypeId":"Vehicle_Type_1"
        }
      ]
    },
    {
      "nodeId":"N3",
      "mapId":"Map_Z-Level_1",
      "nodePosition":{
        "x":13.2,
        "y":0.0
      },
      "vehicleTypeNodeProperties":[
        {
          "vehicleTypeId":"Vehicle_Type_2"
        },
        {
          "vehicleTypeId":"Vehicle_Type_3"
        }
      ]
    }
  ],
  "edges":[
    {
      "edgeId":"N1-NSL",
      "startNodeId":"N1",
      "endNodeId":"NSL",
      "vehicleTypeEdgeProperties":[
        {
          "vehicleTypeId":"Vehicle_Type_1",
          "vehicleOrientation":3.1415926535897931,
          "orientationType":"TANGENTIAL",
          "rotationAllowed":false
        }
      ]
    },
{
  "edgeId":"NSL-N1",
  "startNodeId":"NSL",
  "endNodeId":"N1",
  "vehicleTypeEdgeProperties":[
    {
      "vehicleTypeId":"Vehicle_Type_1",
      "vehicleOrientation":0.0,
      "orientationType":"TANGENTIAL",
      "rotationAllowed":false
    }
  ]
},
{
  "edgeId":"N2-NSB",
  "startNodeId":"N2",
  "endNodeId":"NSB",
  "vehicleTypeEdgeProperties":[
    {
      "vehicleTypeId":"Vehicle_Type_1",
      "vehicleOrientation":3.1415926535897931,
      "orientationType":"TANGENTIAL",
      "rotationAllowed":false
    }
  ]
},
{
  "edgeId":"NSB-N2",
  "startNodeId":"NSB",
  "endNodeId":"N2",
  "vehicleTypeEdgeProperties":[
    {
      "vehicleTypeId":"Vehicle_Type_1",
      "vehicleOrientation":0.0,
      "orientationType":"TANGENTIAL",
      "rotationAllowed":false
    }
  ]
},
{
  "edgeId":"N3-NSR",
  "startNodeId":"N3",
  "endNodeId":"NSR",
  "vehicleTypeEdgeProperties":[
    {
      "vehicleTypeId":"Vehicle_Type_2",
      "vehicleOrientation":3.1415926535897931,
      "orientationType":"TANGENTIAL",
      "rotationAllowed":false
    },
            {
              "vehicleTypeId":"Vehicle_Type_3",
              "vehicleOrientation":3.1415926535897931,
              "orientationType":"TANGENTIAL",
              "rotationAllowed":false
            }
          ]
        },
        {
          "edgeId":"NSR-N3",
          "startNodeId":"NSR",
          "endNodeId":"N3",
          "vehicleTypeEdgeProperties":[
            {
              "vehicleTypeId":"Vehicle_Type_2",
              "vehicleOrientation":0.0,
              "orientationType":"TANGENTIAL",
              "rotationAllowed":false
            },
            {
              "vehicleTypeId":"Vehicle_Type_3",
              "vehicleOrientation":0.0,
              "orientationType":"TANGENTIAL",
              "rotationAllowed":false
            }
          ]
        }
      ],
      "stations":[
        {
          "stationId":"NS",
          "interactionNodeIds":[
            "NSL",
            "NSB",
            "NSR"
          ],
          "stationName":"Complicated handover station",
          "stationDescription":"Handover station for multiple vehicle types with
different allowed actions",
          "stationHeight":"0.5"
        }
      ]
    }
  ]
}
```

10.11 Multiple Edges with Load Restrictions

Multiple edges with different load restrictions applied.


N0
N1
N2
N3
N4
Legend:
Edge: Vehicle Type 1 forward no loads allowed
Edge: Vehicle Type 1 backward no loads allowed
Edge: Vehicle Type 1 omni unloaded or any load type – no restrictions
Edge: Vehicle Type 1 omni unloaded or only load type 2 allowed
Edge: Vehicle Type 1 omni only while loaded with load type 2
Node

LIF File:

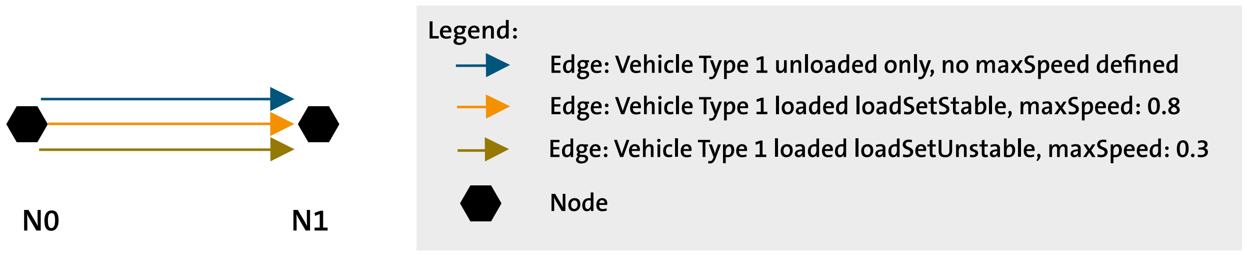
```
{
  "metaInformation":{
    "projectIdentification":"LIF Example 11",
    "creator":"VDMA",
    "exportTimestamp":"2023-09-28T10:00:00.00Z",
    "lifVersion":"1.0.0"
  },
  "layouts":[
    {
      "layoutId":"Layout_Ground_Level",
      "layoutName":"Name of Layout Ground Level",
      "layoutVersion":"1",
      "layoutDescription":"Ground level of Customer",
      "nodes":[
        {
          "nodeId":"N0",
          "mapId":"Map_Z-Level_1",
          "nodePosition":{
            "x":0.0,
            "y":0.0
          },
  "vehicleTypeNodeProperties":[
    {
      "vehicleTypeId":"Vehicle_Type_1",
      "actions":[
        {
          "actionType":"startCharging",
          "requirementType":"CONDITIONAL",
          "blockingType":"HARD"
        },
        {
          "actionType":"stopCharging",
          "requirementType":"CONDITIONAL",
          "blockingType":"HARD"
        }
      ]
    }
  ]
},
{
  "nodeId":"N1",
  "mapId":"Map_Z-Level_1",
  "nodePosition":{
    "x":5.0,
    "y":0.0
  },
  "vehicleTypeNodeProperties":[
    {
      "vehicleTypeId":"Vehicle_Type_1"
    }
  ]
},
{
  "nodeId":"N2",
  "mapId":"Map_Z-Level_1",
  "nodePosition":{
    "x":15.0,
    "y":0.0
  },
  "vehicleTypeNodeProperties":[
    {
      "vehicleTypeId":"Vehicle_Type_1"
    }
  ]
},
{
  "nodeId":"N3",
  "mapId":"Map_Z-Level_1",
  "nodePosition":{
    "x":25.0,
    "y":0.0
  },
    "vehicleTypeNodeProperties":[
      {
        "vehicleTypeId":"Vehicle_Type_1"
      }
    ]
  },
  {
    "nodeId":"N4",
    "mapId":"Map_Z-Level_1",
    "nodePosition":{
      "x":35.0,
      "y":0.0
    },
    "vehicleTypeNodeProperties":[
      {
        "vehicleTypeId":"Vehicle_Type_1"
      }
    ]
  }
],
"edges":[
  {
    "edgeId":"N0-N1",
    "startNodeId":"N0",
    "endNodeId":"N1",
    "vehicleTypeEdgeProperties":[
      {
        "vehicleTypeId":"Vehicle_Type_1",
        "vehicleOrientation":3.1415926535897931,
        "orientationType":"TANGENTIAL",
        "rotationAllowed":false,
        "rotationAtStartNodeAllowed":"NONE",
        "rotationAtEndNodeAllowed":"BOTH",
        "loadRestriction":{
          "unloaded":true,
          "loaded":false
        }
      }
    ]
  },
  {
    "edgeId":"N1-N0",
    "startNodeId":"N1",
    "endNodeId":"N0",
    "vehicleTypeEdgeProperties":[
      {
```

```
      "vehicleTypeId":"Vehicle_Type_1",
      "vehicleOrientation":0.0,
      "orientationType":"TANGENTIAL",
      "rotationAllowed":false,
      "rotationAtStartNodeAllowed":"BOTH",
      "rotationAtEndNodeAllowed":"NONE",
      "loadRestriction":{
        "unloaded":true,
        "loaded":false
      }
    }
  ]
},
{
  "edgeId":"N1-N2",
  "startNodeId":"N1",
  "endNodeId":"N2",
  "vehicleTypeEdgeProperties":[
    {
      "vehicleTypeId":"Vehicle_Type_1",
      "rotationAllowed":false,
      "rotationAtStartNodeAllowed":"BOTH",
      "rotationAtEndNodeAllowed":"BOTH"
    }
  ]
},
{
  "edgeId":"N2-N1",
  "startNodeId":"N2",
  "endNodeId":"N1",
  "vehicleTypeEdgeProperties":[
    {
      "vehicleTypeId":"Vehicle_Type_1",
      "rotationAllowed":false,
      "rotationAtStartNodeAllowed":"BOTH",
      "rotationAtEndNodeAllowed":"BOTH"
    }
  ]
},
{
  "edgeId":"N2-N3",
  "startNodeId":"N2",
  "endNodeId":"N3",
  "vehicleTypeEdgeProperties":[
    {
      "vehicleTypeId":"Vehicle_Type_1",
          "rotationAllowed":false,
          "rotationAtStartNodeAllowed":"BOTH",
          "rotationAtEndNodeAllowed":"BOTH",
          "loadRestriction":{
            "unloaded":true,
            "loaded":true,
            "loadSetNames":[
              "Load_Type_EUR"
            ]
          }
        }
      ]
    },
    {
      "edgeId":"N3-N2",
      "startNodeId":"N3",
      "endNodeId":"N2",
      "vehicleTypeEdgeProperties":[
        {
          "vehicleTypeId":"Vehicle_Type_1",
          "rotationAllowed":false,
          "rotationAtStartNodeAllowed":"BOTH",
          "rotationAtEndNodeAllowed":"BOTH",
          "loadRestriction":{
            "unloaded":true,
            "loaded":true,
            "loadSetNames":[
              "Load_Type_EUR"
            ]
          }
        }
      ]
    },
    {
      "edgeId":"N3-N4",
      "startNodeId":"N3",
      "endNodeId":"N4",
      "vehicleTypeEdgeProperties":[
        {
          "vehicleTypeId":"Vehicle_Type_1",
          "rotationAllowed":false,
          "rotationAtStartNodeAllowed":"BOTH",
          "rotationAtEndNodeAllowed":"BOTH",
          "loadRestriction":{
            "unloaded":false,
            "loaded":true,
            "loadSetNames":[
              "Load_Type_EUR"
            ]
```

```
              }
            }
          ]
        },
        {
          "edgeId":"N4-N3",
          "startNodeId":"N4",
          "endNodeId":"N3",
          "vehicleTypeEdgeProperties":[
            {
              "vehicleTypeId":"Vehicle_Type_1",
              "rotationAllowed":false,
              "rotationAtStartNodeAllowed":"BOTH",
              "rotationAtEndNodeAllowed":"BOTH",
              "loadRestriction":{
                "unloaded":false,
                "loaded":true,
                "loadSetNames":[
                  "Load_Type_EUR"
                ]
              }
            }
          ]
        }
      ]
    }
  ]
}
```

10.12 Multiple Edges Between Same Two Nodes for Different vehicleTypeEdge Property Constraints.

If, for example, a vehicle would be incapable of remembering the properties of the load it is carrying, and/or the (third-party) master control system would be asked to manage the vehicles' maxSpeed or other behaviour, multiple overlapping edges (or in other cases nodes) can accomplish this.

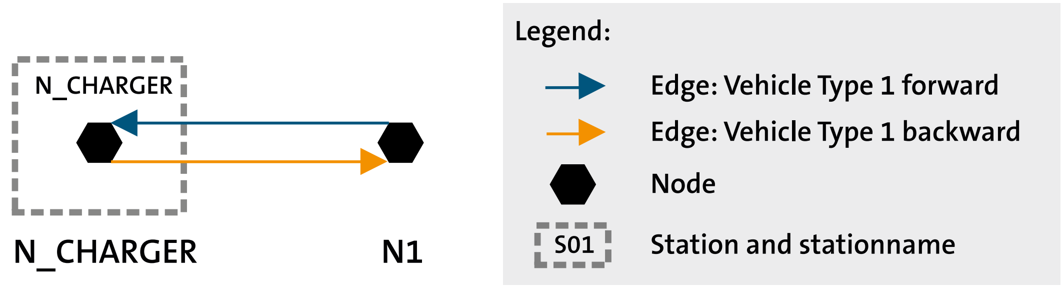

N0
N1
Legend:
Edge: Vehicle Type 1 unloaded only, no maxSpeed defined
Edge: Vehicle Type 1 loaded loadSetStable, maxSpeed: 0.8
Edge: Vehicle Type 1 loaded loadSetUnstable, maxSpeed: 0.3
Node

LIF File:

```
{
  "metaInformation":{
    "projectIdentification":"LIF Example 12",
    "creator":"VDMA",
    "exportTimestamp":"2023-09-28T10:00:00.00Z",
    "lifVersion":"1.0.0"
  },
  "layouts":[
    {
      "layoutId":"Layout_Ground_Level",
      "layoutName":"Name of Layout Ground Level",
      "layoutVersion":"1",
      "layoutDescription":"Ground level of Customer",
      "nodes":[
        {
          "nodeId":"N0",
          "mapId":"Map_Z-Level_1",
          "nodePosition":{
            "x":0.0,
            "y":0.0
          },
    "vehicleTypeNodeProperties":[
      {
        "vehicleTypeId":"Vehicle_Type_1",
        "actions":[
          {
            "actionType":"startCharging",
            "requirementType":"CONDITIONAL",
            "blockingType":"HARD"
          },
          {
            "actionType":"stopCharging",
            "requirementType":"CONDITIONAL",
            "blockingType":"HARD"
          }
        ]
      }
    ]
  },
  {
    "nodeId":"N1",
    "mapId":"Map_Z-Level_1",
    "nodePosition":{
      "x":5.0,
      "y":0.0
    },
    "vehicleTypeNodeProperties":[
      {
        "vehicleTypeId":"Vehicle_Type_1"
      }
    ]
  },
  {
    "nodeId":"N2",
    "mapId":"Map_Z-Level_1",
    "nodePosition":{
      "x":15.0,
      "y":0.0
    },
    "vehicleTypeNodeProperties":[
      {
        "vehicleTypeId":"Vehicle_Type_1"
      }
    ]
  }
],
"edges":[
{
  "edgeId":"N0-N1_Unloaded",
  "startNodeId":"N0",
  "endNodeId":"N1",
  "vehicleTypeEdgeProperties":[
    {
      "vehicleTypeId":"Vehicle_Type_1",
      "vehicleOrientation":3.1415926535897931,
      "orientationType":"TANGENTIAL",
      "rotationAllowed":false,
      "rotationAtStartNodeAllowed":"NONE",
      "rotationAtEndNodeAllowed":"BOTH",
      "loadRestriction":{
        "unloaded":true,
        "loaded":false
      }
    }
  ]
},
{
  "edgeId":"N1-N0_Stable_Load",
  "startNodeId":"N1",
  "endNodeId":"N0",
  "vehicleTypeEdgeProperties":[
    {
      "vehicleTypeId":"Vehicle_Type_1",
      "vehicleOrientation":0.0,
      "orientationType":"TANGENTIAL",
      "rotationAllowed":false,
      "rotationAtStartNodeAllowed":"BOTH",
      "rotationAtEndNodeAllowed":"NONE",
      "maxSpeed":0.8,
      "loadRestriction":{
        "unloaded":false,
        "loaded":true,
        "loadSetNames":[
          "Stable_Load_Unit"
        ]
      }
    }
  ]
},
{
  "edgeId":"N1-N0_Unstable_Load",
  "startNodeId":"N1",
  "endNodeId":"N0",
  "vehicleTypeEdgeProperties":[
            {
              "vehicleTypeId":"Vehicle_Type_1",
              "vehicleOrientation":0.0,
              "orientationType":"TANGENTIAL",
              "rotationAllowed":false,
              "rotationAtStartNodeAllowed":"BOTH",
              "rotationAtEndNodeAllowed":"NONE",
              "maxSpeed":0.3,
              "loadRestriction":{
                "unloaded":false,
                "loaded":true,
                "loadSetNames":[
                  "Unstable_Load_Unit"
                ]
              }
            }
          ]
        }
      ]
    }
  ]
}
```

10.13 Battery Charging Station

Modelling of a battery charging station.



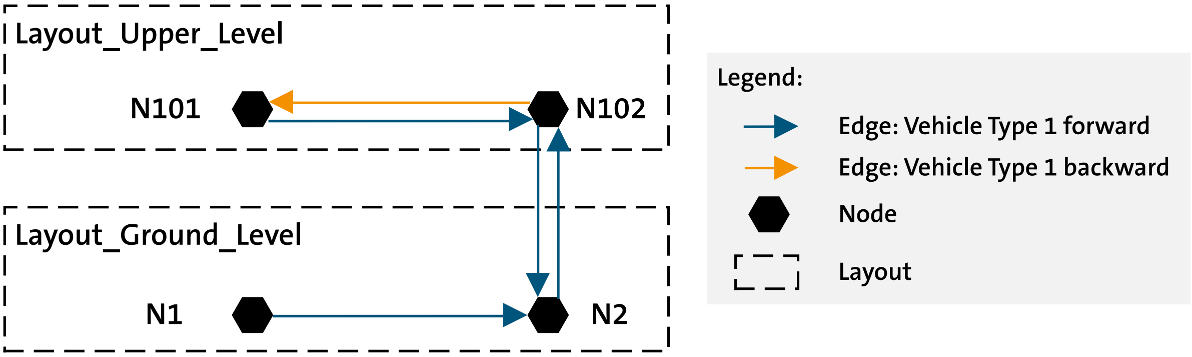
LIF File:

```
{
  "metaInformation":{
    "projectIdentification":"LIF Example 13",
    "creator":"VDMA",
    "exportTimestamp":"2023-09-28T10:00:00.00Z",
    "lifVersion":"1.0.0"
  },
  "layouts":[
    {
      "layoutId":"Layout_Ground_Level",
      "layoutName":"Name of Layout Ground Level",
      "layoutVersion":"1",
      "layoutDescription":"Ground level of Customer",
      "nodes":[
        {
          "nodeId":"N_CHARGER",
          "mapId":"Map_Z-Level_1",
          "nodePosition":{
            "x":0.0,
            "y":0.0
          },
          "vehicleTypeNodeProperties":[
            {
```

```
        "vehicleTypeId":"Vehicle_Type_1",
        "actions":[
          {
            "actionType":"startCharging",
            "requirementType":"CONDITIONAL",
            "blockingType":"HARD"
          },
          {
            "actionType":"stopCharging",
            "requirementType":"CONDITIONAL",
            "blockingType":"HARD"
          }
        ]
      }
    ]
  },
  {
    "nodeId":"N1",
    "mapId":"Map_Z-Level_1",
    "nodePosition":{
      "x":5.0,
      "y":0.0
    },
    "vehicleTypeNodeProperties":[
      {
        "vehicleTypeId":"Vehicle_Type_1"
      }
    ]
  }
],
"edges":[
  {
    "edgeId":"N_CHARGER-N1",
    "startNodeId":"N_CHARGER",
    "endNodeId":"N1",
    "vehicleTypeEdgeProperties":[
      {
        "vehicleTypeId":"Vehicle_Type_1",
        "vehicleOrientation":3.1415926535897931,
        "orientationType":"TANGENTIAL",
        "rotationAllowed":false,
        "rotationAtStartNodeAllowed":"NONE",
        "rotationAtEndNodeAllowed":"BOTH",
        "loadRestriction":{
          "unloaded":true,
          "loaded":false
        }
      }
    ]
  },
        {
          "edgeId":"N1-N_CHARGER",
          "startNodeId":"N1",
          "endNodeId":"N_CHARGER",
          "vehicleTypeEdgeProperties":[
            {
              "vehicleTypeId":"Vehicle_Type_1",
              "vehicleOrientation":0.0,
              "orientationType":"TANGENTIAL",
              "rotationAllowed":false,
              "rotationAtStartNodeAllowed":"BOTH",
              "rotationAtEndNodeAllowed":"NONE",
              "loadRestriction":{
                "unloaded":true,
                "loaded":false
              }
            }
          ]
        }
      ],
      "stations":[
        {
          "stationId":"N_CHARGER",
          "interactionNodeIds":[
            "N_CHARGER"
          ],
          "stationName":"Battery Charging Station",
          "stationDescription":"Station to charge the battery or park the
vehicle",
          "stationHeight":"0.0"
        }
      ]
    }
  ]
}
```

10.14 Two Levels of a Facility in One LIF File

Two layouts in one LIF file, representing two different levels of the facility. Modelling of a transition between two levels.

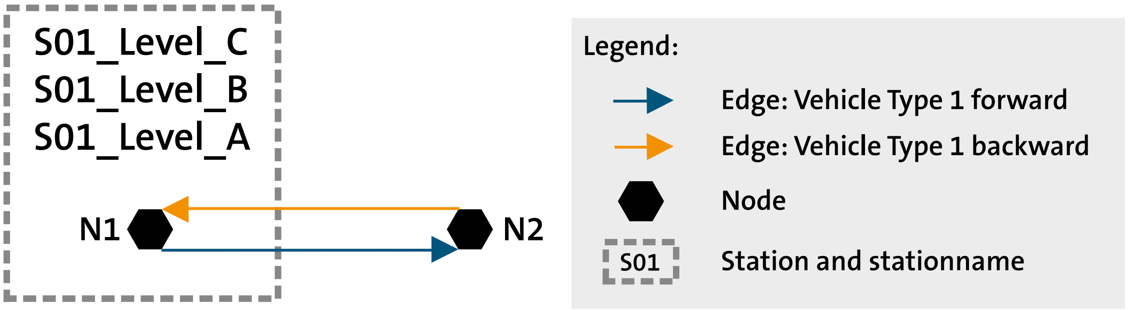


LIF File:

```
{
  "metaInformation":{
    "projectIdentification":"LIF Example 14",
    "creator":"VDMA",
    "exportTimestamp":"2023-09-28T10:00:00.00Z",
    "lifVersion":"1.0.0"
  },
  "layouts":[
    {
      "layoutId":"Layout_Ground_Level",
      "layoutName":"Name of Layout Ground Level",
      "layoutVersion":"1",
      "layoutLevelId":"0",
      "layoutDescription":"Ground level of Customer",
      "nodes":[
        {
          "nodeId":"N1",
          "mapId":"Map_Z-Level_1",
          "nodePosition":{
            "x":0.0,
            "y":0.0
      },
      "vehicleTypeNodeProperties":[
        {
          "vehicleTypeId":"Vehicle_Type_1"
        }
      ]
    },
    {
      "nodeId":"N2",
      "mapId":"Map_Z-Level_1",
      "nodePosition":{
        "x":11.0,
        "y":0.0
      },
      "vehicleTypeNodeProperties":[
        {
          "vehicleTypeId":"Vehicle_Type_1"
        }
      ]
    }
  ],
  "edges":[
    {
      "edgeId":"N1-N2",
      "startNodeId":"N1",
      "endNodeId":"N2",
      "vehicleTypeEdgeProperties":[
        {
          "vehicleTypeId":"Vehicle_Type_1",
          "vehicleOrientation":0.0,
          "orientationType":"TANGENTIAL",
          "rotationAllowed":false
        }
      ]
    },
    {
      "edgeId":"N2-N102",
      "startNodeId":"N2",
      "endNodeId":"N102",
      "vehicleTypeEdgeProperties":[
        {
          "vehicleTypeId":"Vehicle_Type_1",
          "vehicleOrientation":0.0,
          "orientationType":"TANGENTIAL",
          "rotationAllowed":false
        }
      ]
    }
  ]
},
{
  "layoutId":"Layout_Upper_Level",
  "layoutName":"Name of Layout Upper Level",
  "layoutVersion":"1",
  "layoutLevelId":"1",
  "layoutDescription":"Upper level of Customer",
  "nodes":[
    {
      "nodeId":"N102",
      "mapId":"Map_Z-Level_2",
      "nodePosition":{
        "x":12.4,
        "y":3.4
      },
      "vehicleTypeNodeProperties":[
        {
          "vehicleTypeId":"Vehicle_Type_1"
        }
      ]
    },
    {
      "nodeId":"N101",
      "mapId":"Map_Z-Level_2",
      "nodePosition":{
        "x":12.0,
        "y":3.4
      },
      "vehicleTypeNodeProperties":[
        {
          "vehicleTypeId":"Vehicle_Type_1"
        }
      ]
    }
  ],
  "edges":[
    {
      "edgeId":"N102-N101",
      "startNodeId":"N102",
      "endNodeId":"N101",
      "vehicleTypeEdgeProperties":[
        {
              "vehicleTypeId":"Vehicle_Type_1",
              "vehicleOrientation":3.1415926535897931,
              "orientationType":"TANGENTIAL",
              "rotationAllowed":false
            }
          ]
        },
        {
          "edgeId":"N101-N102",
          "startNodeId":"N101",
          "endNodeId":"N102",
          "vehicleTypeEdgeProperties":[
            {
              "vehicleTypeId":"Vehicle_Type_1",
              "vehicleOrientation":0.0,
              "orientationType":"TANGENTIAL",
              "rotationAllowed":false
            }
          ]
        },
        {
          "edgeId":"N102-N2",
          "startNodeId":"N102",
          "endNodeId":"N2",
          "vehicleTypeEdgeProperties":[
            {
              "vehicleTypeId":"Vehicle_Type_1",
              "vehicleOrientation":0.0,
              "orientationType":"TANGENTIAL",
              "rotationAllowed":false
            }
          ]
        }
      ]
    }
  ]
}
```

10.15 Rack Station Modelled by Three Stations

Rack station with three levels modelled by three individual stations.



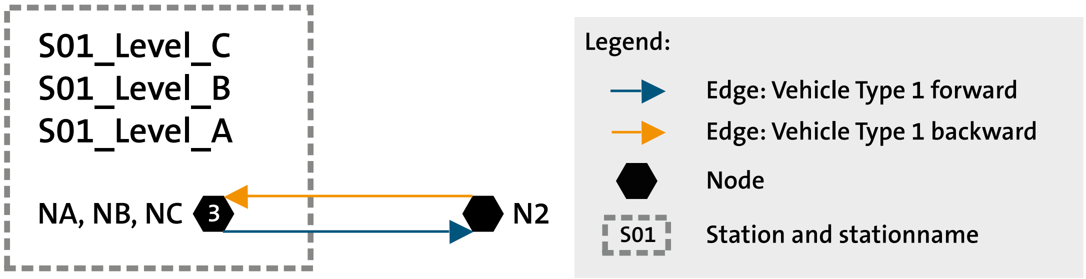
LIF File:

```
{
  "metaInformation":{
    "projectIdentification":"LIF Example 15",
    "creator":"VDMA",
    "exportTimestamp":"2023-09-28T10:00:00.00Z",
    "lifVersion":"1.0.0"
  },
  "layouts":[
    {
      "layoutId":"Layout_Ground_Level",
      "layoutName":"Name of Layout Ground Level",
      "layoutVersion":"1",
      "layoutDescription":"Ground level of Customer",
      "nodes":[
        {
          "nodeId":"N1",
          "mapId":"Map_Z-Level_1",
          "nodePosition":{
            "x":7.2,
            "y":0.0
          },
      "vehicleTypeNodeProperties":[
        {
          "vehicleTypeId":"Vehicle_Type_1",
          "actions":[
            {
              "actionType":"pick",
              "requirementType":"CONDITIONAL",
              "blockingType":"HARD"
            },
            {
              "actionType":"drop",
              "requirementType":"CONDITIONAL",
              "blockingType":"HARD"
            }
          ]
        }
      ]
    },
    {
      "nodeId":"N2",
      "mapId":"Map_Z-Level_1",
      "nodePosition":{
        "x":9.2,
        "y":0.0
      },
      "vehicleTypeNodeProperties":[
        {
          "vehicleTypeId":"Vehicle_Type_1"
        }
      ]
    }
  ],
  "edges":[
    {
      "edgeId":"N1-N2",
      "startNodeId":"N1",
      "endNodeId":"N2",
      "vehicleTypeEdgeProperties":[
        {
          "vehicleTypeId":"Vehicle_Type_1",
          "vehicleOrientation":3.1415926535897931,
          "orientationType":"TANGENTIAL",
          "rotationAllowed":false
        }
      ]
    },
        {
          "edgeId":"N2-N1",
          "startNodeId":"N2",
          "endNodeId":"N1",
          "vehicleTypeEdgeProperties":[
            {
              "vehicleTypeId":"Vehicle_Type_1",
              "vehicleOrientation":0.0,
              "orientationType":"TANGENTIAL",
              "rotationAllowed":false
            }
          ]
        }
      ],
      "stations":[
        {
          "stationId":"S01_Level_A",
          "interactionNodeIds":[
            "N1"
          ],
          "stationName":"Shelf on Level A",
          "stationDescription":"Shelf on level A",
          "stationHeight":"0.0"
        },
        {
          "stationId":"S01_Level_B",
          "interactionNodeIds":[
            "N1"
          ],
          "stationName":"Shelf on Level B",
          "stationDescription":"Shelf on level B",
          "stationHeight":"2.5"
        },
        {
          "stationId":"S01_Level_2",
          "interactionNodeIds":[
            "N1"
          ],
          "stationName":"Shelf on Level C",
          "stationDescription":"Shelf on level C",
          "stationHeight":"5.0"
        }
      ]
    }
  ]
}
```

10.16 Rack Station Modelled by Three Nodes

Rack station with three levels modelled by three different nodes:

- Node NA is only for picking a load.
- Node NB is only for dropping a load.
- Node NC is for picking and dropping a load.



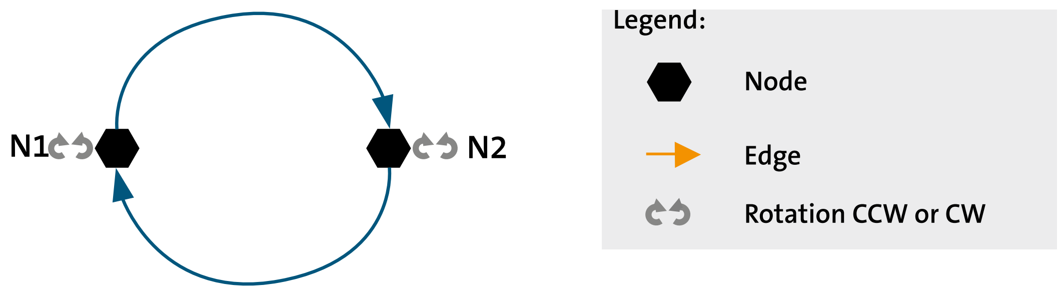
LIF File:

```
{
  "metaInformation":{
    "projectIdentification":"LIF Example 16",
    "creator":"VDMA",
    "exportTimestamp":"2023-09-28T10:00:00.00Z",
    "lifVersion":"1.0.0"
  },
  "layouts":[
    {
      "layoutId":"Layout_Ground_Level",
      "layoutName":"Name of Layout Ground Level",
      "layoutVersion":"1",
      "layoutDescription":"Ground level of Customer",
      "nodes":[
        {
          "nodeId":"NA",
          "mapId":"Map_Z-Level_1",
          "nodePosition":{
            "x":7.2,
            "y":0.0
          },
          "vehicleTypeNodeProperties":[
            {
              "vehicleTypeId":"Vehicle_Type_1",
              "actions":[
        {
          "actionType":"pick",
          "requirementType":"CONDITIONAL",
          "blockingType":"HARD"
        }
      ]
    }
  ]
},
{
  "nodeId":"NB",
  "mapId":"Map_Z-Level_1",
  "nodePosition":{
    "x":7.2,
    "y":0.0
  },
  "vehicleTypeNodeProperties":[
    {
      "vehicleTypeId":"Vehicle_Type_1",
      "actions":[
        {
          "actionType":"drop",
          "requirementType":"CONDITIONAL",
          "blockingType":"HARD"
        }
      ]
    }
  ]
},
{
  "nodeId":"NC",
  "mapId":"Map_Z-Level_1",
  "nodePosition":{
    "x":7.2,
    "y":0.0
  },
  "vehicleTypeNodeProperties":[
    {
      "vehicleTypeId":"Vehicle_Type_1",
      "actions":[
        {
          "actionType":"pick",
          "requirementType":"CONDITIONAL",
          "blockingType":"HARD"
        },
            {
              "actionType":"drop",
              "requirementType":"CONDITIONAL",
              "blockingType":"HARD"
            }
          ]
        }
      ]
    },
    {
      "nodeId":"N2",
      "mapId":"Map_Z-Level_1",
      "nodePosition":{
        "x":9.2,
        "y":0.0
      },
      "vehicleTypeNodeProperties":[
        {
          "vehicleTypeId":"Vehicle_Type_1"
        }
      ]
    }
  ],
  "edges":[
    {
      "edgeId":"NA-N2",
      "startNodeId":"NA",
      "endNodeId":"N2",
      "vehicleTypeEdgeProperties":[
        {
          "vehicleTypeId":"Vehicle_Type_1",
          "vehicleOrientation":3.1415926535897931,
          "orientationType":"TANGENTIAL",
          "rotationAllowed":false
        }
      ]
    },
    {
      "edgeId":"N2-NA",
      "startNodeId":"N2",
      "endNodeId":"NA",
      "vehicleTypeEdgeProperties":[
        {
          "vehicleTypeId":"Vehicle_Type_1",
          "vehicleOrientation":0.0,
          "orientationType":"TANGENTIAL",
          "rotationAllowed":false
        }
      ]
    },
{
  "edgeId":"NB-N2",
  "startNodeId":"NA",
  "endNodeId":"N2",
  "vehicleTypeEdgeProperties":[
    {
      "vehicleTypeId":"Vehicle_Type_1",
      "vehicleOrientation":3.1415926535897931,
      "orientationType":"TANGENTIAL",
      "rotationAllowed":false
    }
  ]
},
{
  "edgeId":"N2-NB",
  "startNodeId":"N2",
  "endNodeId":"NB",
  "vehicleTypeEdgeProperties":[
    {
      "vehicleTypeId":"Vehicle_Type_1",
      "vehicleOrientation":0.0,
      "orientationType":"TANGENTIAL",
      "rotationAllowed":false
    }
  ]
},
{
  "edgeId":"NC-N2",
  "startNodeId":"NC",
  "endNodeId":"N2",
  "vehicleTypeEdgeProperties":[
    {
      "vehicleTypeId":"Vehicle_Type_1",
      "vehicleOrientation":3.1415926535897931,
      "orientationType":"TANGENTIAL",
      "rotationAllowed":false
    }
  ]
},
{
  "edgeId":"N2-NC",
  "startNodeId":"N2",
  "endNodeId":"NC",
  "vehicleTypeEdgeProperties":[
    {
      "vehicleTypeId":"Vehicle_Type_1",
      "vehicleOrientation":0.0,
      "orientationType":"TANGENTIAL",
      "rotationAllowed":false
```

```
            }
          ]
        }
      ],
      "stations":[
        {
          "stationId":"S01_Level_A",
          "interactionNodeIds":[
            "NA"
          ],
          "stationName":"Shelf on Level A",
          "stationDescription":"Handover shelf from manual trucks. Inbound
toward AGV system only.",
          "stationHeight":"0.0"
        },
        {
          "stationId":"S01_Level_B",
          "interactionNodeIds":[
            "NB"
          ],
          "stationName":"Shelf on Level B",
          "stationDescription":"Handover shelf toward manual trucks. Outbound
away from AGV system only.",
          "stationHeight":"2.5"
        },
        {
          "stationId":"S01_Level_C",
          "interactionNodeIds":[
            "NC"
          ],
          "stationName":"Shelf on Level C",
          "stationDescription":"Special bi-directional handover shelf.",
          "stationHeight":"5.0"
        }
      ]
    }
  ]
}
```

10.17 Edge with Trajectory Definition

Two edges between node N1 and N2 with a half circle trajectory. Before entering the edge N1 to N2 the vehicle needs to rotate on N1 to -90°. The vehicle will maintain the -90° while moving on the edge. Before entering the edge N2 to N1 the vehicle needs to rotate to 90°. The vehicle will maintain the 90° while moving on the edge.

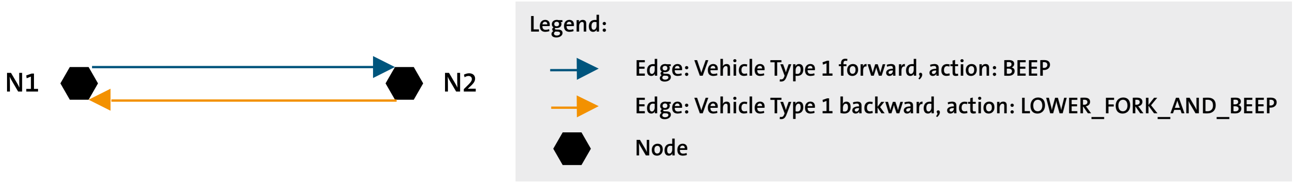


LIF File:

```
{
  "metaInformation":{
    "projectIdentification":"LIF Example 17",
    "creator":"VDMA",
    "exportTimestamp":"2023-09-28T10:00:00.00Z",
    "lifVersion":"1.0.0"
  },
  "layouts":[
    {
      "layoutId":"Layout_Ground_Level",
      "layoutName":"Name of Layout Ground Level",
      "layoutVersion":"1",
      "layoutDescription":"Ground level of Customer",
      "nodes":[
        {
          "nodeId":"N1",
          "mapId":"Map_Z-Level_1",
          "nodePosition":{
            "x":5.0,
            "y":0.0
          },
    "vehicleTypeNodeProperties":[
      {
        "vehicleTypeId":"Vehicle_Type_1"
      }
    ]
  },
  {
    "nodeId":"N2",
    "mapId":"Map_Z-Level_1",
    "nodePosition":{
      "x":15.0,
      "y":0.0
    },
    "vehicleTypeNodeProperties":[
      {
        "vehicleTypeId":"Vehicle_Type_1"
      }
    ]
  }
],
"edges":[
  {
    "edgeId":"N1-N2",
    "startNodeId":"N1",
    "endNodeId":"N2",
    "vehicleTypeEdgeProperties":[
      {
        "vehicleTypeId":"Vehicle_Type_1",
        "vehicleOrientation":-1.5707963267948966,
        "orientationType":"TANGENTIAL",
        "rotationAllowed":false,
        "rotationAtStartNodeAllowed":"BOTH",
        "rotationAtEndNodeAllowed":"BOTH",
        "trajectory":{
          "degree":2,
          "knotVector":[
            0,
            0,
            0,
            0.5,
            1,
            1,
            1
          ],
        "controlPoints":[
          {
            "x":0,
            "y":0
          },
          {
            "x":0,
            "y":1.8
          },
          {
            "x":3.6,
            "y":1.8
          },
          {
            "x":3.6,
            "y":0
          }
        ]
      }
    }
  ]
},
{
  "edgeId":"N2-N1",
  "startNodeId":"N2",
  "endNodeId":"N1",
  "vehicleTypeEdgeProperties":[
    {
      "vehicleTypeId":"Vehicle_Type_1",
      "vehicleOrientation":1.5707963267948966,
      "orientationType":"TANGENTIAL",
      "rotationAllowed":false,
      "rotationAtStartNodeAllowed":"BOTH",
      "rotationAtEndNodeAllowed":"BOTH",
      "trajectory":{
        "degree":2,
        "knotVector":[
          0,
          0,
          0,
          0.5,
          1,
          1,
          1
        ],
        "controlPoints":[
          {
            "x":3.6,
            "y":0
          },
                  {
                    "x":3.6,
                    "y":-1.8
                  },
                  {
                    "x":0,
                    "y":-1.8
                  },
                  {
                    "x":0,
                    "y":0
                  }
                ]
              }
            }
          ]
        }
      ]
    }
  ]
}
```

10.18 Manufacturer Specific Action on an Edge

Manufacturer specific action on an edge.

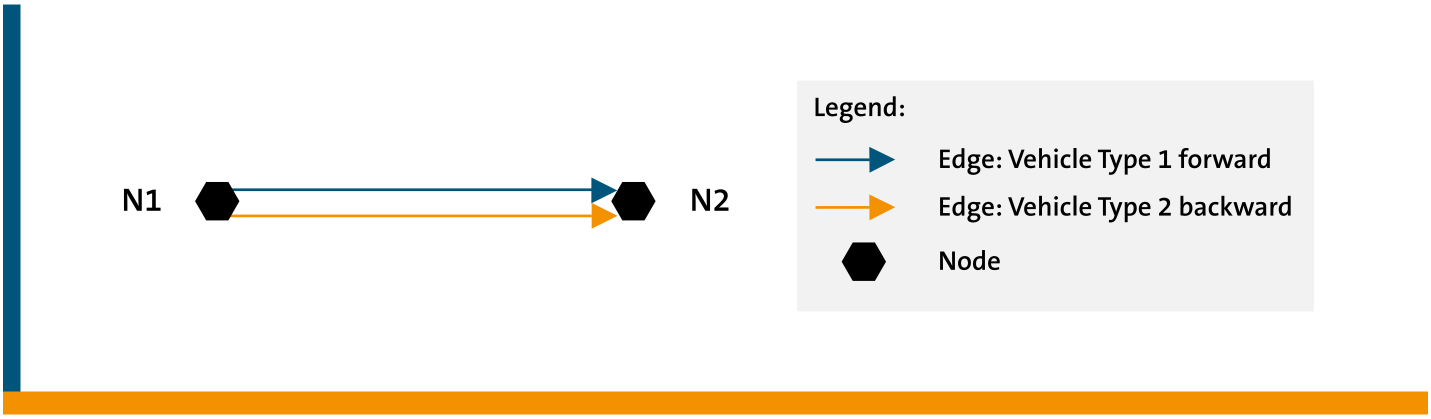

N1
N2
Legend:
Edge: Vehicle Type 1 forward, action: BEEP
Edge: Vehicle Type 1 backward, action: LOWER_FORK_AND_BEEP
Node

LIF File:

```
{
  "metaInformation":{
    "projectIdentification":"LIF Example 18",
    "creator":"VDMA",
    "exportTimestamp":"2023-09-28T10:00:00.00Z",
    "lifVersion":"1.0.0"
  },
  "layouts":[
    {
      "layoutId":"Layout_Ground_Level",
      "layoutName":"Name of Layout Ground Level",
      "layoutVersion":"1",
      "layoutDescription":"Ground level of Customer",
      "nodes":[
        {
          "nodeId":"N1",
          "mapId":"Map_Z-Level_1",
          "nodePosition":{
            "x":0.0,
            "y":0.0
          },
          "vehicleTypeNodeProperties":[
            {
              "vehicleTypeId":"Vehicle_Type_1"
            }
          ]
        },
        {
          "nodeId":"N2",
          "mapId":"Map_Z-Level_1",
          "nodePosition":{
            "x":11.0,
            "y":0.0
          },
          "vehicleTypeNodeProperties":[
            {
              "vehicleTypeId":"Vehicle_Type_1"
            }
          ]
        }
      ],
      "edges":[
        {
          "edgeId":"N1-N2",
          "startNodeId":"N1",
          "endNodeId":"N2",
          "vehicleTypeEdgeProperties":[
            {
              "vehicleTypeId":"Vehicle_Type_1",
              "vehicleOrientation":0.0,
              "orientationType":"TANGENTIAL",
              "rotationAllowed":false,
              "actions":[
                {
                  "actionType":"BEEP",
                  "actionDescription":"Section where the (third-party) master
control system could instruct the vehicle to beep",
                  "requirementType":"OPTIONAL",
                  "blockingType":"SOFT"
                }
              ]
            }
          ]
        },
          {
            "edgeId":"N2-N1",
            "startNodeId":"N2",
            "endNodeId":"N1",
            "vehicleTypeEdgeProperties":[
              {
                "vehicleTypeId":"Vehicle_Type_1",
                "vehicleOrientation":3.1415926535897931,
                "orientationType":"TANGENTIAL",
                "rotationAllowed":false,
                "actions":[
                  {
                    "actionType":"LOWER_FORK_AND_BEEP",
                    "actionDescription":"Section where the (third-party) master
  control system must tell the AGV to lower forks and beep",
                    "requirementType":"REQUIRED",
                    "blockingType":"SOFT"
                  }
                ]
              }
            ]
          }
        ]
      }
    ]
  }
```

10.19 Forward Edge with Two Vehicle Types with Differing Orientation

One edge (straight line) between two nodes, where two different vehicle types from the same vehicle integrator must adopt different orientations.



LIF File:

```
{
  "metaInformation":{
    "projectIdentification":"LIF Example 19",
    "creator":"VDMA",
    "exportTimestamp":"2023-09-28T10:00:00.00Z",
    "lifVersion":"1.0.0"
  },
  "layouts":[
    {
      "layoutId":"Layout_Ground_Level",
      "layoutName":"Name of Layout Ground Level",
      "layoutVersion":"1",
      "layoutDescription":"Ground level of Customer",
      "nodes":[
        {
          "nodeId":"N1",
          "mapId":"Map_Z-Level_1",
          "nodePosition":{
            "x":0.0,
            "y":0.0
          },
          "vehicleTypeNodeProperties":[
            {
              "vehicleTypeId":"Vehicle_Type_1"
            },
            {
              "vehicleTypeId":"Vehicle_Type_2"
            }
          ]
        },
        {
          "nodeId":"N2",
          "mapId":"Map_Z-Level_1",
          "nodePosition":{
            "x":11.0,
            "y":0.0
          },
          "vehicleTypeNodeProperties":[
            {
              "vehicleTypeId":"Vehicle_Type_1"
            },
            {
              "vehicleTypeId":"Vehicle_Type_2"
            }
          ]
        }
      ],
      "edges":[
        {
          "edgeId":"N1-N2",
          "startNodeId":"N1",
          "endNodeId":"N2",
          "vehicleTypeEdgeProperties":[
            {
              "vehicleTypeId":"Vehicle_Type_1",
              "vehicleOrientation":0.0,
              "orientationType":"TANGENTIAL",
              "rotationAllowed":false,
              "rotationAtStartNodeAllowed":"NONE",
              "rotationAtEndNodeAllowed":"NONE"
            },
            {
              "vehicleTypeId":"Vehicle_Type_2",
              "vehicleOrientation":1.5707963267948966,
              "orientationType":"TANGENTIAL",
              "rotationAllowed":false,
              "rotationAtStartNodeAllowed":"NONE",
              "rotationAtEndNodeAllowed":"NONE"
            }
          ]
        }
      ]
    }
  ]
}
```

Imprint

Imprint

VDMA
Materials Handling and Intralogistics

Lyoner Str. 18
60528 Frankfurt am Main
Germany

Contact
Andreas Scherb
Phone +49 69 6603-1724
Fax +49 69 6603-2724
Internet foerd.vdma.org

Editor
Andreas Scherb

Picture Credits
Cover Page shutterstock

Stand
March 2024

VDMA
Materials Handling and Intralogistics

Lyoner Str. 18
60528 Frankfurt am Main
Germany

Contact
Andreas Scherb
Phone +49 69 6603-1724
E-Mail foerd@vdma.org
Internet foerd.vdma.org

foerd.vdma.org

DesignStudio