# Opturion Dynamic Transport Optimiser Data Formats

Opturion

16 June 2025

# Contents

1	Ove	rview	1
2	Dat	a Model	2
	2.1	Scenarios	2
	2.2	Solutions	2
	2.3	Identifiers	2
	2.4	Attributes	2
	2.5	Weights and Volumes	2
	2.6	Time	3
	2.7	Costs and Hidden Costs	3
	2.8	Global Parameters	4
		2.8.1 Iterations	4
		2.8.2 Batched Loads	4
		2.8.3 Colocated Pickups	4
		2.8.4 Sticky Deliveries	4
		2.8.5 Scaling Factors	4
		2.8.6 Geofencing	4
		2.8.7 Maximum Neighbourhood Size	5
		2.8.8 Preferred-Vehicle Rewards	5
	2.9	Replanning	6
	2.10	Locations	7
	2.11	Vehicles	9
		2.11.1 Vehicle Costs	9
		2.11.2 Vehicle Locations	11
		2.11.3 Vehicle Times	11
		2.11.4 Vehicle Capacities	12
		2.11.5 Vehicle Compartments	$12^{$
		2.11.6 Maximum Loads	$12^{$
		2.11.7 Maximum Drops per Load	12
		2.11.8 Fatigue Management Scheme	$12^{$
		2.11.9 Loaded Breaks	$12^{$
		2.11.10 Vehicle Speed Scale	$12^{$
		2.11.11 Vehicle Types	$\frac{12}{12}$
	2.12	Orders	13
	2.13	Fatigue Management	$15^{-10}$
	2.14	Iteration Schemes	16
	2.15	Time Windows	17
		2.15.1 Arrival Only in Time Window	17
	2 16	Soft Time Windows	18
	2.10 2.17	Compatibility	19
	2.18	Stops	$\frac{10}{21}$
	2.10	Loads	$\frac{21}{21}$
	2.13	Solution Times	21 21
	2.20 2.21	Unassigned Orders	21 99
	2.21 2.21	Pickup Segments	⊿⊿ ງຈ
	2.22	Delivery Segments	20 92
	2.20	Finish Soments	20 92
	2.24		ചാ

ABN: 13 146 662 053

i

10 June 2020	16	June	2025
--------------	----	------	------

	2.25	Loade	1 Distance Penalties    24
3	Mic	rosoft	Excel Input 25
	3.1	Identif	iers in Excel
	3.2	Times	in Excel
	3.3	Sheet:	General
		3.3.1	Column: Parameter
		3.3.2	Column: Value
		3.3.3	Parameter: Scenario Name
		3.3.4	Parameter: Iterations
		3.3.5	Parameter: Rush Hour
		3.3.6	Parameter: Use Miles
		3.3.7	Parameter: Iteration Scheme
		3.3.8	Parameter: Colocated Pickups
		3.3.9	Parameter: Geofence North
		3.3.10	Parameter: Geofence East
		3.3.11	Parameter: Geofence South
		3.3.12	Parameter: Geofence West 27
		3 3 13	Parameter: Hard Geofence Errors 27
		3 3 14	Parameter: Batched Loads 28
		3 3 15	Parameter: Sticky Deliveries 28
		3 3 16	Parameter: Cost Scale 28
		3317	Parameter: Weight Scale 28
		3 3 18	Parameter: Volume Scale 28
		3 3 10	Parameter: Maximum Neighbourhood Size
		2 2 2 20	Parameter: Disable Replanning
		3.3.20 3.2.21	Parameter: Disable Replaining
		0.0.21	Parameter: Previous-venicle Fixed Reward
		0.0.22	Parameter: Previous-Vehicle Weight Reward
		3.3.23	Parameter: Previous-venicie volume Reward
		3.3.24	Parameter: Forbid Unrecognised Columns
	9.4	3.3.25	Parameter: Arrival Only in Time Window
	3.4	Sheet:	Locations
		3.4.1	
		3.4.2	Column: Name
		3.4.3	Column: Attributes
		3.4.4	Column: Address
		3.4.5	Column: Longitude
		3.4.6	Column: Latitude
		3.4.7	Column: Site Time
		3.4.8	Column: Load Time
		3.4.9	Column: Unload Time    31
		3.4.10	Column: Opening Time    31
		3.4.11	Column: Closing Time    31
		3.4.12	Column: Operating Time Windows    31
		3.4.13	Column: Cost per Visit
		3.4.14	Column: Hidden Cost per Visit
	3.5	Sheet:	<b>O</b> rders
		3.5.1	Column: Id
		3.5.2	Column: Name         32

16	June	2025
----	------	------

iii

	3.5.3	Column:	Pickup Location
	3.5.4	Column:	Pickup Service Time
	3.5.5	Column:	Earliest Pickup Time
	3.5.6	Column:	Latest Pickup Time    32
	3.5.7	Column:	Pickup Time Windows 32
	3.5.8	Column:	Pickup Soft Time Windows
	3.5.9	Column:	Soft Pickup Constant
	3.5.10	Column:	Soft Pickup Variable
	3.5.11	Column:	Delivery Location
	3.5.12	Column:	Delivery Service Time
	3.5.13	Column:	Earliest Delivery Time
	3.5.14	Column:	Latest Delivery Time
	3.5.15	Column:	Delivery Time Windows
	3.5.16	Column:	Delivery Soft Time Windows
	3.5.17	Column:	Soft Delivery Constant
	3.5.18	Column:	Soft Delivery Variable
	3.5.19	Column:	Maximum Duration
	3.5.20	Column:	Weight
	3.5.21	Column:	Volume
	3.5.22	Column:	Assign Cost
	3.5.23	Column:	Hidden Assign Cost
	3.5.24	Column:	Attributes
3.6	Sheet:	Vehicle T	Vpe
	3.6.1	Column:	Id
	3.6.2	Column:	Start Location
	3.6.3	Column:	Finish Location
	3.6.4	Column:	Maximum Weight
	3.6.5	Column:	Maximum Volume
	3.6.6	Column:	Earliest Start Time
	367	Column	Latest Start Time 37
	368	Column	Latest Finish Time 37
	369	Column	Minimum Paid Time 37
	3 6 10	Column	Maximum Drive Time 37
	3 6 11	Column:	Maximum Work Time 37
	3 6 12	Column:	Load Time 37
	3 6 13	Column:	Unload Time 38
	3614	Column	Speed Scale 38
	3615	Column:	Cost per Use 38
	3 6 16	Column:	Cost per Hour 38
	3617	Column:	Cost per Kilometre 38
	3 6 18	Column:	Cost per Mile 38
	3 6 10	Column:	Cost per load 30
	3 6 20	Column:	Hidden Cest per Use
	3.6.20	Column:	Hidden Cost per Ose
	3699	Column	Hidden Cost per Hour
	3692	Column	Hidden Cost per Mile 20
	0.0.20 2.6.94	Column:	Hidden Cost per Load 20
	3.0.24	Column:	Finish Corment Distance Denalty 40
	0.0.20 2.6.96	Column:	Finish Segment Der Hour Denaltz
	5.0.20	Column:	r mish beginent per hour penalty 40

16 June 2025

	3.6.27	Column:	Pickup Segment Fixed Penalty	
	3.6.28	Column:	Pickup Segment Distance Penalty 40	
	3.6.29	Column:	Delivery Segment Fixed Penalty 40	
	3.6.30	Column:	Delivery Segment Distance Penalty 40	
	3.6.31	Column:	Loaded Weight Distance Penalty	
	3.6.32	Column:	Loaded Volume Distance Penalty 41	
	3.6.33	Column:	Attributes	
	3.6.34	Column:	Break Scheme	
	3.6.35	Column:	Forbid Loaded Breaks	
	3.6.36	Column:	Maximum Loads	
	3.6.37	Column:	Compartments	
3.7	Sheet:	Fleet		
	3.7.1	Column:	Id	
	3.7.2	Column:	Vehicle Type	
	3.7.3	Column:	Name	
	3.7.4	Column:	Start Location	
	3.7.5	Column:	Finish Location	
	3.7.6	Column:	Maximum Weight	
	3.7.7	Column:	Maximum Volume	
	3.7.8	Column:	Earliest Start Time	
	3.7.9	Column:	Latest Start Time	
	3.7.10	Column:	Latest Finish Time	
	3.7.11	Column:	Minimum Paid Time	
	3.7.12	Column:	Maximum Drive Time	
	3.7.13	Column:	Maximum Work Time	
	3.7.14	Column:	Load Time	
	3.7.15	Column:	<b>Unload Time</b>	
	3.7.16	Column:	Speed Scale	
	3.7.17	Column:	Cost per Use	
	3.7.18	Column:	Cost per Hour	
	3.7.19	Column:	Cost per Kilometre	
	3.7.20	Column:	Cost per Mile 45	
	3.7.21	Column:	Cost per Load	
	3.7.22	Column:	Hidden Cost per Use 45	
	3.7.23	Column:	Hidden Cost per Hour	
	3.7.24	Column:	Hidden Cost per Kilometre	
	3.7.25	Column:	Hidden Cost per Mile	
	3.7.26	Column:	Hidden Cost per Load	
	3.7.27	Column:	Finish Segment Distance Penalty46	
	3.7.28	Column:	Finish Segment Per Hour Penalty 46	
	3.7.29	Column:	Delivery Segment Fixed Penalty	
	3.7.30	Column:	Delivery Segment Distance Penalty 46	
	3.7.31	Column:	Loaded Weight Distance Penalty	
	3.7.32	Column:	Loaded Volume Distance Penalty 46	
	3.7.33	Column:	Attributes	
	3.7.34	Column:	Break Scheme	
	3.7.35	Column:	Forbid Loaded Breaks	
	3.7.36	Column:	Maximum Loads	
	3.7.37	Column:	Maximum Drops per Load 47	

REGISTERED OFFICE: LEVEL 1, 18 KAVANAGH ST, SOUTHBANK, VIC, 3006

16 June 2025

	3.7.38	Column: Compartments
3.8	Sheet:	Fatigue Management    48
	3.8.1	Column: Id
	382	Column: Number 48
	383	Column: Duration 48
	9.0.9 9.0.4	Column. Internal 40
2.0	3.8.4	
3.9	Sneet:	1 me windows
	3.9.1	Column: Id
	3.9.2	Column: Start
	3.9.3	Column: End
3.10	Sheet:	Compartments
	3.10.1	Column: Scheme Id 50
	3.10.2	Column: Id
	3 10 3	Column: Name 50
	3 10 /	Column: Maximum Weight 50
	9.10.4	Column: Maximum Volumo
	3.10.3	
~	3.10.6	Column: Attributes
3.11	Sheet:	Rush Hour
	3.11.1	Column: Start
	3.11.2	Column: End
	3.11.3	Column: Speed Scale
3.12	Sheet:	Iteration Schemes
	3.12.1	Column: Id
	3 12 2	Column: Iterations 52
	3 1 9 3	Column: Maximum Naighbourhood Size
9 1 9	5.12.5 Choot.	Orden Vehiele Competibility
0.10	3121	
	3.13.1	Column A: Order Attribute
	3.13.2	Row 1: Vehicle Attribute
	3.13.3	All other cells (except $A1$ )
3.14	Sheet:	Order-Compartment Compatibility
	3.14.1	Column A: Order Attribute
	3.14.2	Row 1: Compartment Attribute
	3.14.3	All other cells (except A1)
3 15	Sheet.	Order-Order Compatibility 55
0.10	3 15 1	Column A: Order Attribute 55
	3 15 9	Bow 1: Order Attribute 55
	9.15.2	All other cells (except 11)
9.10	0.10.0	
3.10	Sneet:	
	3.16.1	Cell A1
	3.16.2	Column A: Location Attribute
	3.16.3	Row 1: Vehicle Attribute
	3.16.4	All other cells
3.17	Sheet:	Order-Vehicle Costs
	3.17.1	Column: Order
	3.17.2	Column: Vehicle
	3 17 3	Column: Cost 57
	3 17 /	Column: Hidden Cost 57
9 10	Sheet	Orden Vehiele Dreferences
J.18	Sheet:	Orliner-venicie rieleiences
	3.18.1	Column:         Order         58

REGISTERED OFFICE: LEVEL 1, 18 KAVANAGH ST, SOUTHBANK, VIC, 3006

16 June	2025
---------	------

		3.18.2	Column: Vehicle
		3.18.3	Column: Fixed Reward 58
		3.18.4	Column: Weight Reward 58
		3.18.5	Column: Volume Reward 58
	3.19	Sheet:	Order Precedences
		3.19.1	Column: Before         59
		3.19.2	Column: After
		3.19.3	Column: Gap
	3.20	Sheet:	Previous Solution
		3.20.1	Column: Order
		3.20.2	Column: Vehicle
		3.20.3	Column: Stop
	3.21	Sheet:	Time Matrix   61
		3.21.1	The Matrix
	3.22	Sheet:	Distance Matrix
		3.22.1	The Matrix
	3.23	Note (	On Matrices
4	Mic	$\mathbf{rosoft}$	Excel Output 62
	4.1	Sheet:	Runsheet
		4.1.1	Column: Vehicle
		4.1.2	Column: Load
		4.1.3	Column: Stop Number
		4.1.4	Column: ETA
		4.1.5	Column: ETD
		4.1.6	$Column: Stop \dots \dots$
		4.1.7	Column: Order
		4.1.8	Column: Compartment
		4.1.9	Column: Address
		4.1.10	Column: Volume
		4.1.11	Column: Delta Volume
		4.1.12	Column: Weight
		4.1.13	Column: Delta Weight
		4.1.14	Column: Site Time
		4.1.15	Column: Load Time
		4.1.16	Column: Unload Time
		4.1.17	Column: Service Time
		4.1.18	Column: Transit Time
		4.1.19	Column: Transit Distance
		4.1.20	Column: Soft Time Window Start
		4.1.21	Column: Soft Time Window End
		4.1.22	Column: Soft Time Window Delta
	4.2	Sheet:	Vehicle Summary
		4.2.1	Column: Vehicle
		4.2.2	Column: Used
		4.2.3	Column: Cost
		4.2.4	Column: Peak Volume
		4.2.5	Column: Total Volume
		4.2.6	Column: Volume Capacity

16 June 2025

vii

	4.2.7	Column:	Peak Weight
	4.2.8	Column:	Total Weight
	4.2.9	Column:	Weight Capacity
	4.2.10	Column:	Number Of Loads   66
	4.2.11	Column:	Transit Distance
	4.2.12	Column:	Loaded Distance
	4.2.13	Column:	Empty Distance
	4.2.14	Column:	Actual Start Time
	4.2.15	Column:	Actual Finish Time
	4.2.16	Column:	Work Time
	4.2.17	Column:	Allowed Work Time
	4.2.18	Column:	Transit Time
	4.2.19	Column:	Allowed Transit Time
	4.2.20	Column:	Site Time
	4.2.21	Column:	Load Time
	4.2.22	Column:	Unload Time
	4.2.23	Column:	Service Time
	4.2.24	Column:	Idle Time
	4.2.25	Column:	Break Time
	4.2.26	Column:	Number Of Drops
	4 2 27	Column	Number Of Orders 67
	4 2 28	Column	Average Transit Speed 67
	4 2 29	Column	Pickup Soft Time Window Early 67
	4 2 30	Column	Pickup Soft Time Window Late 68
	4 2 31	Column	Delivery Soft Time Window Early 68
	4 2 32	Column	Delivery Soft Time Window Late 68
43	Sheet:	Load Sur	nmary 69
1.0	4 3 1	Column	Vehicle 69
	432	Column	Load Number 69
	4.3.3	Column:	Peak Volume 69
	4.0.0	Column:	Volume Capacity 69
	435	Column:	Peak Weight 69
	4.3.5	Column:	Weight Capacity 60
	4.3.0	Column:	Loaded Distance 60
	4.3.1	Column:	Empty Distance Refore
	4.3.0	Column:	Empty Distance Defore
	4.3.3	Column:	Work Time 70
	4.3.10	Column:	Transit Time 70
	4.3.11	Column:	Site Time 70
	4.5.12	Column:	Load Time 70
	4.3.13	Column:	Unload Time 70
	4 3 15	Column:	Sorvice Time 70
	4.0.10 / 2.16	Column	Idle Time 70
	4.3.10	Column	Break Time 70
	4.J.17	Column	Number Of Drops 70
	4.0.10	Column	Average Transit Speed 71
	4.3.13	Column	Pickup Soft Time Window Farly 71
	4.J.2U / 2.91	Column	Pickup Soft Time Window Late 71
	4.0.21	Column:	Delivery Soft Time Window Farly 71
	4.3.22	Corumn:	

10 June 202	16	e 202	June	25
-------------	----	-------	------	----

viii

		4.3.23	Column: Delivery Soft Time Window Late    71
	4.4	Sheet:	Solution Summary
		4.4.1	Scenario Name 72
		4.4.2	Total Iterations
		4.4.3	Total Cost
		4.4.4	Total Distance
		4.4.5	Loaded Distance
		4.4.6	Empty Distance
		4.4.7	Average Transit Speed (km/h)
		4.4.8	Average Transit Speed (mph)
		4.4.9	Loads
		4.4.10	Km Per Load
		4.4.11	Miles Per Load
		4 4 12	Hours Per Load 73
		4 4 13	Drops Per Load 73
		4.4.10	Work Time 73
		4 4 15	Transit Time 73
		<i>A A</i> 16	Break Time 73
		4.4.17	Idle Time 73
		4.4.17	Total Weight 73
		4.4.10	Total Velume 73
		4.4.19	Aggigned Orderg 74
		4.4.20	Floot Usage 74
		4.4.21	Column: Didum Soft Time Window Forly 74
		4.4.22	Columni. Fickup Soft Time Window Larry
		4.4.23	Columni: Pickup Soft Time Window Late
		4.4.24	Column: Delivery Soft Time Window Early
	4 5	4.4.20	Column: Delivery Soft Time window Late
	4.5	Sneet:	Unassigned Orders
		4.5.1	Column: Order
		4.5.2	Column: Pickup Location
		4.5.3	Column: Delivery Location
		4.5.4	Column: Reason
	4.6	Sheet:	Replan Changes         76
		4.6.1	Column: Order
		4.6.2	Column: Weight
		4.6.3	Column: Volume
		4.6.4	Column: Previous Vehicle
		4.6.5	Column: New Vehicle
		4.6.6	Column: Previous Pickup Time76
		4.6.7	Column: New Pickup Time76
		4.6.8	Column: Previous Delivery Time76
		4.6.9	Column: New Delivery Time    77
_	TOO		
5	JSC	ON Inp	ut 78
	5.1	Scenar	70 Object
	5.2	Genera	al Object $\ldots \ldots \ldots$
	5.3	Locati	on Object
	5.4	Order	Object
	5.5	Vehicle	e Object

Registered Office: Level 1, 18 Kavanagh St, Southbank, VIC, 3006

### 16 June 2025

	5.6	Break Object	96			
	5.7	Compartment Object	97			
	5.8	Time Window Object	98			
	5.9	Soft Time Window Object	99			
	5.10	Attribute Pair Object	100			
	5.11	Rush Hour Object	101			
	5.12	Iteration Batch Object	102			
	5.13	Vehicle Cost Object	103			
	5.14	Vehicle Preference Object	104			
	5.15	Order Precedence Object	105			
6	JSO	N Output	106			
Ŭ	6.1	Solution Object	106			
	6.2	Route Object	110			
	6.3	Stop Object	112			
	6.4	Stop Type Codes	114			
	6.5	Route KPI Object	115			
	6.6	Load KPI Object	117			
	6.7	Unassigned Reason Object	119			
Appendices 120						
A	JSO	N Input Example	120			
в	3 JSON Output Example 124					
С	$\mathbf{Uns}$	upported Excel Functions	128			

# 1 Overview

This document specifies the input and output formats of the Opturion Dynamic Transport Optimiser (DTO). The input and output to the DTO can be given either as a Microsoft Excel workbook (DTO-Excel, see Sections 3 and 4) or as a JSON document (DTO-JSON, see Sections 5 and 6).

The DTO-Excel format is intended for use as a standalone modelling interface.

The DTO-JSON interface is intended for use when interacting with the DTO via a web service.

Examples of data input and output in JSON format are provided in Appendices A and B.

# 2 Data Model

This section provides an overview of the major concepts in the DTO data model.

# 2.1 Scenarios

An input to the DTO takes the form of a set of parameters, locations, vehicles and orders, called a *scenario*. A scenario is a representation of the transport optimisation problem you wish to solve. It includes a description of all the orders that require servicing, all available vehicles and their properties, location data and any order-vehicle restrictions as well as other optional data which is described in the following sections.

# 2.2 Solutions

A *solution* is a plan created by the optimiser. It consists of a schedule that assigns orders to vehicles.

# 2.3 Identifiers

Entities such as orders, locations and vehicles within a scenario must have a unique identifier (Id) that is used to refer to them.

Identifiers must contain at least one character and must not contain the semicolon (;) character. They are not case or "white space" sensitive.

For example, "DEPOT2", "depot2" and "Depot 2" will all be treated as identifying the same entity by the optimiser because they only differ in case or the amount of white space they have.

Even though they may be composed of digits, identifiers are not numeric. For example, the optimiser will treat "1" and "001" as being distinct identifiers.

(Users of the DTO-Excel format should be aware that there are Excel-specific restrictions on the specification of identifiers; see Section 3.1 for details.)

All references to an entity are made using its identifier.

# 2.4 Attributes

Attributes are labels that are attached to entities like orders and vehicles to describe their properties. Attribute names, like identifiers, must not contain semicolons and are not case or "white space" sensitive.

Attributes may be used to define relationships between the entities in a scenario. (See Section 2.17 for further details).

Note that an entity's identifier is automatically considered to be one of its attributes.

# 2.5 Weights and Volumes

Weights and volumes in a scenario can be specified in arbitrary units, however the selected units must be used consistently throughout the scenario. Mixing units, giving order weights in kilograms and vehicle capacities in tons for example, will lead to errors.

# 2.6 Time

Time is specified in *HH:MM* format.

The hour component of a time may extend beyond 23, allowing times later than a single day to be specified. For example, 36:00 would be midday on the second day.

# 2.7 Costs and Hidden Costs

The optimiser deals with two kinds of cost: actual (i.e. monetary) costs and hidden costs.

Vehicles, orders and locations can have both actual and hidden costs associated with them.

Hidden costs are considered by the optimiser when making decisions, but they do *not* contribute to the actual costs reported by the optimiser. Hidden costs are nominally measured in the same units as actual costs.

Hidden costs have many uses, for example they can be used to cause the optimiser to prefer the use of a particular subset of the fleet or used to affect the "shape" of routes.

# 2.8 Global Parameters

## 2.8.1 Iterations

The number of iterations the optimiser will run for. Each iteration may result in an improved solution and hence a higher number of iterations increases the probability of finding a better solution. This comes at the cost of increasing the duration the optimiser will run for.

# 2.8.2 Batched Loads

Batched loads is a scheduling constraint that enforces a strict "Pickup - Delivery" cycle where once a vehicle begins making deliveries, it cannot pick up new orders until it has delivered all of its current orders. (See Section 2.19 for further information.)

## 2.8.3 Colocated Pickups

Colocated pickups refers to a scheduling constraint that all orders in a load must have come from the same location.

# 2.8.4 Sticky Deliveries

If *sticky deliveries* are enabled then the optimiser will attempt to assign all of the orders being delivered to an individual location to the *same* vehicle. The intention of this is to reduce the number of different vehicles that make deliveries at a location.

# 2.8.5 Scaling Factors

A scaling factor can be applied to speeds, costs, weights and volumes in the scenario. This allows the user to increase the effect that a particular variable has on the optimiser. A value of 0 means that no scaling is applied. The scaling factors include speed scale, cost scale, weight scale and volume scale.

# 2.8.6 Geofencing

The optimiser can be restricted to only consider orders and vehicles that are located within a specific geographic area. This area is called a *geofence* and it takes the form of a bounding box that is defined by the latitudes of its southern and northern boundaries and by the longitudes of its western and eastern boundaries.

The geofence can operate in one of two modes: *hard* and *soft*.

If the geofence is in *hard* mode then the optimiser will report an error for any order that has pickup or delivery locations outside the geofence. It will also report an error for any vehicle that has start or finish locations outside of the geofence. In the event of such errors being detected, optimisation will *not* proceed.

If the geofence is in *soft* mode then the optimiser will not assign any order whose pickup or delivery location is outside the geofence. It will not use any vehicle whose start or finish location is outside the geofence. In this mode optimisation will proceed using orders and vehicles that are within the geofence.

## 2.8.7 Maximum Neighbourhood Size

The maximum size of the neighbourhood for the optimiser. At each iteration of the optimiser, the solution is unpacked and changed to find a new solution with the neighbourhood being the amount to unpack. Smaller neighbourhoods are quicker to change but may result in worse quality solutions, while larger neighbourhoods require a longer running time but may lead to better quality solutions. Note that the maximum neighbourhood size must be at least ten.

# 2.8.8 Preferred-Vehicle Rewards

If **preferred vehicles** are specified for an order, then there are three types of rewards that can be specified.

# 1. Preferred Vehicle Fixed Reward

This is a fixed reward for assigning the order to one of its preferred vehicles.

## 2. Preferred-Vehicle Weight Reward

This reward is multiplied by the weight of the order if it is assigned to one of its preferred vehicles. It can be used to give a higher priority to assigning heavier orders to their preferred vehicles.

# 3. Preferred-Vehicle Volume Reward

This reward is multiplied by the volume of the order if it is assigned to one of its preferred vehicles. It can be used to give a higher priority to assigning larger orders to their preferred vehicles.

# 2.9 Replanning

Sometimes it is necessary to update or extend a solution generated by the optimiser. Reasons for this include:

- Orders being added.
- Orders being cancelled.
- Order details (e.g. weights, volumes, time windows etc.) changing.
- Vehicles being added to the fleet.
- Vehicles being removed from the fleet.
- Vehicle / driver availability changing.

When this happens, it is often desirable to maintain as much of the existing solution as practicable. For example, when loads are being assembled at the loading docks of a warehouse, altering the original plan too much may unduly affect warehouse operations.

This usage of the optimiser is called *replanning*. When replanning, an updated scenario is extended with information about the assignments and sequencing in a previous solution. The optimiser can be configured to minimise the differences between the solution it generates and the previous solution while incorporating any changes.

The following global parameters control the degree and manner to which deviations from the existing solution can occur.

## 1. Previous-Vehicle Fixed Reward

This parameter gives the strength of the preference for orders to remain on their previously assigned vehicles.

## 2. Previous-Vehicle Weight Reward

This parameter gives the strength of the preference for heavier orders to remain on their previously assigned vehicles.

## 3. Previous-Vehicle Volume Reward

This parameter gives the strength of the preference for larger orders to remain on the their previously assigned vehicles.

Each of the above rewards is a form of hidden cost and their effect is cumulative.

# 2.10 Locations

#### 1. Identifier

Each location must have a unique identifier. This identifier acts as a reference to the location when referred to from other data (e.g. orders).

## 2. Name

Each location may have a name associated with it, for example a store or warehouse name.

#### 3. Address

Each location may have a street address associated with it.

#### 4. Coordinates

Each location must be provided with its geographic coordinates specified as latitude and longitude in decimal degrees.

#### 5. Site Time

Site time is the time incurred by the vehicle when it arrives at the location. It is *only* incurred when the vehicle arrives at a location. Multiple consecutive pickups or deliveries at the *same* location without any intervening travel do not individually incur site time.

#### 6. Load Time

Load time is the time incurred by the vehicle when it visits the location for a pickup or pickups. It is incurred only once per sequence of consecutive pickups at the *same* location. If the vehicle also has a positive load time, then the vehicle's load time will override this value.

#### 7. Unload Time

Unload time is the time incurred by the vehicle when it visits the location for a delivery or deliveries. It is incurred only once per sequence of consecutive deliveries at the *same* location. If the vehicle also has a positive **unload time**, then the vehicle's unload time will override this value.

#### 8. Opening Time

The earliest time a vehicle can pickup or deliver at the location.

#### 9. Closing Time

The latest time a vehicle can pickup or deliver at the location.

#### $10. \ {\bf Time \ Windows}$

Time windows define spans of time when a vehicle can pickup or deliver at the location. Time window may be a single span of time, such as 9:00-17:00 or multiple spans, such as 9:00-12:00 and 14:00-17:00.

#### 11. Operating Times

The *operating times* are times at which a vehicle can pickup or deliver at the location. Operating times are defined by the opening time, closing time and any time windows.

#### 12. Cost per Visit

This is a cost that is incurred whenever a vehicle visits the location.

# 13. Hidden Cost per Visit

This is a hidden cost that is incurred whenever a vehicle visits the location. This is used in the computation of a solution, but will not be seen in the final output.

# 2.11 Vehicles

# 2.11.1 Vehicle Costs

Vehicle costs are specified in arbitrary units. The unit used must be consistent across the scenario.

## 1. Cost per Use

This is a cost incurred whenever the vehicle is used (i.e. has orders assigned to it) in a solution.

## 2. Hidden Cost per Use

The hidden version of the cost per use. This is used in the computation of a solution, but it will not be seen in the final output.

#### 3. Cost per Hour

This is the cost-per-hour for the vehicle.

## 4. Hidden Cost per Hour

The hidden version of the cost per hour. This is used in the computation of a solution, but it will not be seen in the final output.

#### 5. Cost per Kilometre

This is the cost-per-kilometre for the vehicle.

#### 6. Hidden Cost per Kilometre

The hidden version of the cost per km.

#### 7. Cost per Mile

This is the cost-per-mile for the vehicle.

## 8. Hidden Cost per Mile

The hidden version of the cost per mile.

## 9. Cost per Load

This is a cost that is incurred for each load a vehicle has in the solution.

## 10. Hidden Cost per Load

The hidden version of the cost per load. This is used in the computation of a solution, but it will not be seen in the final output.

16 June 2025

10

Costs and hidden costs per distance may be specified using either kilometres or miles, but  $\mathit{not}$  both.

There are additional hidden costs (penalties) for vehicles that are useful for controlling the "shape" of a vehicle's route.

- 1. Pickup segment penalties.
- 2. Delivery segment penalties.
- 3. Finish segment penalties.
- 4. Loaded volume distance penalties.
- 5. Loaded weight distance penalties.

#### 2.11.2 Vehicle Locations

#### 1. Starting Location

This is the location of the vehicle at the start of its route.

If this location differs from the first pickup location, then the vehicle is required to travel from the start location to the first pickup location. In such a case, the travel time and distance to the first pickup location are included in the solution.

If no start location is specified, then the vehicle begins its route at the location of its first pickup.

#### 2. Finishing Location

This is a location to which the vehicle must return at the end of its route. The travel time and distance to return to this location are considered in the solution.

If no finish location is specified, then the vehicle ends its route at the location of its last delivery.

#### 2.11.3 Vehicle Times

#### 1. Earliest Start

This is the earliest time that a vehicle can start its route.

#### 2. Latest Start

This is the latest time that a vehicle can start its route.

#### 3. Latest Finish

This is the latest time that the vehicle can finish its route. It includes the time taken for the vehicle to return to its finish location.

#### 4. Maximum Drive Time

This is the maximum time that a vehicle can spend in transit during a route.

#### 5. Maximum Work Time

This is the maximum time allowed between the start of the route and its completion. This includes the time needed for a vehicle to return to its finish location at the end of its route.

#### 6. Load Time

Load time is the time incurred by the vehicle when it begins a sequence of pickups at a location. A vehicle's load time will override a location's load time if *both* the vehicle's and location's load time are greater than zero.

#### 7. Unload Time

Unload time is the time incurred by the vehicle when it begins a sequence of deliveries at a location. A vehicle's unload time will override a location's unload time if *both* the vehicle's and location's unload times are greater than zero.

12

# 2.11.4 Vehicle Capacities

Vehicles may have both a maximum weight capacity and maximum volume capacity. These are upper bounds on how much the vehicle can carry.

# 2.11.5 Vehicle Compartments

Vehicles may be divided into multiple *compartments*. Each compartment has its own capacities (e.g. maximum weight, maximum volume) and attributes. A *compartment scheme* is a list of compartments identified using a unique identifier. An individual compartment scheme can be applied to multiple vehicles.

## 2.11.6 Maximum Loads

If **batched loads** is enabled, then this is an upper bound on the number of loads the vehicle is allowed to do.

## 2.11.7 Maximum Drops per Load

If **batched loads** is enabled, then this is an upper bound on the number of locations a vehicle can deliver to per load.

## 2.11.8 Fatigue Management Scheme

This is an identifier that specifies the fatigue management rules that apply to the vehicle. Driver breaks inserted by the optimiser will satisfy all rules in the fatigue management scheme.

## 2.11.9 Loaded Breaks

A *loaded break* is a driver break that is scheduled while one or more orders are loaded on the vehicle. The optimiser can optionally prevent loaded breaks from being scheduled (i.e. driver breaks will only occur while the vehicle is not loaded).

## 2.11.10 Vehicle Speed Scale

As all vehicles use the same time and distance matrices in computation of transit times and distance, applying a scaling factor allows the user to account for vehicles with varying speed. For example, if a vehicle was known to be 30% slower than the average vehicle in the fleet, a scaling factor of 0.7 (70%) could be applied. Similarly, if a vehicle was known to be 10% faster, a scaling factor of 1.10 (110%) could be applied.

# 2.11.11 Vehicle Types

To avoid unnecessary repetition when defining vehicles, DTO-Excel (only) allows for the definition of vehicle types. All vehicles of the same type share the properties defined for that type, such as maximum weight, speed scale etc. Note that defining both a vehicle type and vehicle properties in the Fleet sheet is not allowed.

# 2.12 Orders

## 1. Locations

The pickup and delivery locations must be given for each order and must match a location identifier defined in the set of locations. (See Section 2.10.)

#### 2. Times

#### (a) Earliest and Latest Times

The earliest pickup time is the earliest time that the order will be available at its pickup location. The earliest delivery time is the earliest time the same order is able to be delivered. This is analogous for latest times.

#### (b) Time Windows

The times when an order can be picked up and delivered can also be specified by time windows. If both earliest and latest times and time windows are specified, then *both* must be satisfied for the order to be picked up or delivered.

#### (c) Service Time

Each pickup and delivery may have an amount of service time associated with it. This is the amount of time incurred by the vehicle when the order is picked up or delivered. This time is *in addition* to any site, load or unload time incurred because of the location or vehicle.

#### 3. Soft Time Window Penalties

If an order has soft time windows then both fixed and variable penalties are incurred if these time windows are missed.

#### (a) Fixed Penalty

A constant penalty that is incurred if the soft time windows are missed. Note that if multiple soft time windows are defined, this penalty is still only incurred once.

## (b) Variable Penalty

A penalty incurred per minute the time window is missed. If multiple time windows are present, it is calculated based on the nearest soft time window boundary.

#### 4. Maximum Duration

Orders can optionally have a *maximum duration* specified. The optimiser will not generate solutions that assign such orders in a way that the time needed to service them exceeds their maximum duration. (The *duration* of an order is defined to be the time that elapses from when its pickup commences up until the time when its delivery is complete.)

Note that if the minimum service time needed by an order exceeds its maximum duration then the order cannot be assigned by the optimiser.

#### 5. Weight and Volume

The weight and volume of the order are given in arbitrary units which must be consistent with the respective vehicle capacities previously defined. For example, the volume may be measured in cubic metres, number of pallets or roll cages.

16 June 2025

## 6. Assign Cost

This is a (monetary) cost associated with servicing the order. This can be used to prefer certain orders as the optimiser will attempt to find the solution with lowest cost. Negative costs can also be used.

#### 7. Hidden Assign Cost

The hidden version of the assign cost. While this is used in the computation of a solution, but it will not be seen in the final output.

#### 8. Vehicle-Specific Assign Cost

An order can have costs associated with it that are only incurred when the order is assigned to a specific vehicle or set of vehicles. These costs are incurred in addition to any assign cost for the order.

#### 9. Vehicle-Specific Hidden Assign Cost

An order can have hidden costs associated with it that are only incurred when the order is assigned to a specific vehicle or set of vehicles. These hidden costs are incurred in addition to any hidden assign cost for the order.

#### 10. **Preferred Vehicles**

An order can have a list of preferred vehicles. A reward is obtained on assigning the order to one of its preferred vehicles. This reward is calculated using a fixed reward, a volume dependent reward and a weight dependent reward. See Global Parameters for an explanation of these rewards.

#### 11. Order Precedences

Two orders can be associated with a precedence. This requires that the *delivery* of the first order must be completed before the *pickup* of the second. An optional duration that must elapse between the delivery and the pickup can also be specified. This duration is called the *gap*.

Precedences are often used to model cross-docking, where the first leg of the order must be delivered before the second leg can commence.

# 2.13 Fatigue Management

A fatigue management scheme is a set of rules that specify the number of breaks of a certain duration that must occur within a given time interval (period). Each rule consists of the following:

## 1. Number of Breaks

The number of breaks required by the rule.

## 2. Duration of Break

The duration of the break.

#### 3. Interval of Break

The time interval (period) in which the break must occur.

**Example** A fatigue management scheme may have the rule: A single 15 minute break every 2 hours. Here, the number of breaks is 1, the duration of the break is 15 minutes and the interval in which the break must occur is 2 hours.

16

# 2.14 Iteration Schemes

Iteration schemes provide finer-grained control over how the optimiser explores its solution space. An iteration scheme allows a single invocation of the optimiser to be split into a sequence of "batches", where each batch runs for a specified number of iterations with a specified maximum neighbourhood size. In the absence of an iteration scheme the optimiser will run with a single fixed maximum neighbourhood size for all of the iterations.

An example iteration scheme is:

- 1000 iterations with a maximum neighbourhood size of 200.
- 2000 iterations with a maximum neighbourhood size of 150.
- $\bullet$  4000 iterations with a maximum neighbourhood size of 100.

In this scheme, the optimiser will run for a total of 7000 iterations. For the first 1000 iterations the optimiser will use a maximum neighbourhood size of 200; for the next 2000 it will use a maximum neighbourhood size of 150; for the last 4000 it will use a maximum neighbourhood size of 100.

Using an iteration scheme may improve performance as the optimiser is allowed a greater degree of freedom in the choices it makes at the beginning of a batch. Having multiple batches effectively increases the amount of time that the optimiser is afforded this greater degree of freedom.

Reducing the maximum neighbourhood size in later batches (as in the above example) reduces the likelihood of the optimiser straying too far from a good solution and failing to find any further improvements.

17

# 2.15 Time Windows

Time windows represent a period of time within which either an order can be serviced or a location is open depending on the context.

**Example**: a location is open between 6:00-12:00, closed for lunch and open again between 13:00-17:00. Figure 1 shows how to represent this using time windows.

```
Γ
        {
           "start": "06:00",
           "end": "12:00"
        },
        {
           "start": "13:00",
           "end": "17:00"
        }
      ]
                     В
                                 С
        A
        ID
1
                    Start
                                End
2
  DAY
                       06:00
                                 12:00
3
  DAY
                       13:00
                                 17:00
```

Figure 1: Example Time Window in JSON format (top) and Excel format (bottom).

#### 2.15.1 Arrival Only in Time Window

Time windows have two modes of operation. The first ensures that every pickup or delivery occurs within the specified time window. This is the default behaviour. The second is called "Arrival Only in Time Window". This mode of operation make it so that the time window is satisfied for each pickup (or delivery) if the time window was satisfied upon the arrival of the vehicle at the location. This second mode of operation is only enabled when Batched Loads is also enabled.

	Α	В	С
1	Stop	Time	Time Window
2	START	9:00	
3	PICKUP	9:20	9:00 - 10:00
4	PICKUP	9:25	9:00 - 9:20
5	PICKUP	9:30	8:30 - 9:15
6	DELIVERY	11:00	9:00 - 12:00
7	BREAK	11:05	
8	DELIVERY	11:35	9:00 - 11:30
9	DELIVERY	11:40	9:00 - 11:30
10	FINISH	12:00	

Figure 2: Example route with time windows to illustrate "Arrival Only in Time Window" behaviour. All pickups and all deliveries are at the same location. Time windows shaded green are satisfied normally. If "Arrival Only in Time Window" is enabled, then time windows shaded yellow are also satisfied, since the arrival stop is within the time window. The time window shaded red is violated with both schemes, since neither the stop or the arrival stop are within the time window.

# 2.16 Soft Time Windows

Soft time windows apply only to orders. They represent a preferred time of service (i.e. pickup or delivery). The optimiser may schedule pickups or deliveries outside of any soft time windows but will incur a penalty for doing so.

A soft time window penalty has two components: a *fixed* component that is incurred if the soft time window is missed and a *variable* component that is a per-minute rate based on the number of minutes from the nearest time window boundary. The soft time window penalty acts as an additional form of hidden cost.

**Example** Assume the time windows in Figure 1 are delivery soft time windows and we have 200 and 5 as the fixed and variable penalty components respectively. In this case the hard time window exists from 6:00-17:00. If the order was delivered at 12:20, we would incur a fixed penalty of 200 and a variable penalty of  $5 \times 20 = 100$  for a total penalty of 300.

# 2.17 Compatibility

Compatibility relationships are used to restrict what decisions the optimiser may make when generating a solution. The following kinds of compatibility relationships are supported:

- **Order-vehicle compatibility**: an order may only be assigned to a vehicle with which it is compatible.
- **Order-compartment compatibility**: an order may only be loaded in a compartment with which it is compatible.
- **Order-order compatibility**: orders may only be simultaneously loaded into a compartment if they are compatible.
- Load Order-order compatibility: orders may only be simultaneously loaded on to a vehicle if they are compatible.
- Location-vehicle compatibility: a vehicle must only pickup or deliver orders to locations with which it is compatible.

Compatibility relationships are specified using attributes. As an example, consider a fleet that consists of both refrigerated and non-refrigerated vehicles. To restrict an order that requires refrigeration to be assigned to only a refrigerated vehicle, we can label the order with the attribute REFRIGERATED and also label all of the refrigerated vehicles with the attribute REFRIGERATED. We can then specify an *order-vehicle* compatibility relationship such that orders with the attribute REFRIGERATED must only be assigned to vehicles with the attribute REFRIGERATED.

Order\Vehicle	REFRIGERATED	TALL	LONG	VEHICLE1
REFRIGERATED	true			
TALL		true	true	
ORDER7				true

Table 1: Order-Vehicle Compatibility.

Table 1 is an example of an *order-vehicle* compatibility relationship, where **true** means that the attribute pair is compatible and an empty cell means that the attribute pair is *not* compatible.

Table 1 is interpreted as follows: for orders with the attribute REFRIGERATED, the compatible vehicles must have the attribute REFRIGERATED; for orders with the attribute TALL, the compatible vehicles must have **either** the attribute TALL or the attribute LONG; for the orders with the attribute ORDER7, the compatible vehicles must have the attribute VEHICLE1. For the last case, if ORDER7 is one of the order identifiers and VEHICLE1 is one of the vehicle identifiers then this will restrict ORDER7 to only be compatible with VEHICLE1. If ORDER7 has other attributes the VEHICLE1 is not compatible with then this will mean this order cannot be assigned to any vehicle. (See **Conjunction Rule** below).

There are three rules applied when interpreting attributes and their compatibility. Here we use *order-vehicle* compatibility to illustrate them.

1. Null Rule: An order attribute that is not part of a *order-vehicle* compatibility relationship imposes no restriction on which vehicle the order can be assigned to. We can ignore this attribute or, equivalently, consider it compatible with all vehicle attributes.

20

- 2. **Disjunction Rule**: If an order attribute has multiple compatible vehicle attributes, then any vehicle having *one* of the attributes is a candidate compatible vehicle.
- 3. **Conjunction Rule**: If an order has multiple attributes, a compatible vehicle for the order must have a compatible attribute for each of the order's attributes.

These rules are applied in only one direction: attributes are interpreted as requirements for orders and features for vehicles. An order does not need to have a compatible attribute for each of a vehicle's attributes to be compatible with that vehicle. For example, in Table 1, an order that has only one attribute, REFRIGERATED, is compatible with a vehicle that has attributes REFRIGERATED and LONG. These rules apply similarly to *order-compartment* and *location-vehicle* compatibility.

For *order-order* and *load order-order* compatibility, these rules must be applied in *both* directions, that is: for two compatible orders, each of them must have a compatible attribute for each attribute of the other order.

# 2.18 Stops

In a solution, a vehicle's route consists of a sequence of stops, each with a time-of-arrival and time-of-departure. The different stop types are:

- 1. **START**: Denotes the start of the route.
- 2. FINISH: Denotes the end of the route.
- 3. PICKUP: Denotes an order pickup.
- 4. DELIVERY: Denotes an order delivery.
- 5. BREAK: Denotes a break for the vehicle.

A route must begin with a START stop and end with a FINISH stop. These are the only positions in the route where START and FINISH stops can appear. Every PICKUP stop in the route must have a matching DELIVERY stop.

# 2.19 Loads

A *load* is defined to be a sequence of PICKUP stops followed by a sequence of DELIVERY stops such that once the first DELIVERY stop occurs, no further PICKUP stops can occur until the vehicle is empty.

#### Example:

Stop Number	Load	Not a Load
1	PICKUP	PICKUP
2	PICKUP	PICKUP
3	PICKUP	DELIVERY
4	DELIVERY	PICKUP
5	DELIVERY	DELIVERY
6	DELIVERY	DELIVERY

# 2.20 Solution Times

In the solution generated by the optimiser, we have the following notions of times:

#### 1. Work Time

In the vehicle summary, this is the time between leaving the START stop and arriving at the FINISH stop in a vehicle's route. A vehicle's work time is the total time taken to complete the route, while a load's work time is the total time taken to complete the load.

2. Site Time

A vehicle's site time is the sum of all site time incurred over the entire route, while a load's site time is the sum of all site time incurred during the load.

#### 3. Load Time

A vehicle's load time is the sum of all load times incurred over the entire route. If **batched** loads is enabled, this measure is analogous for each of the vehicle's loads.

# 4. Unload Time

A vehicle's unload time is the sum of all unload time incurred over the entire route. If **batched loads** is enabled, this measure is analogous for each of the vehicle's loads.

## 5. Service Time

A vehicle's service time is the sum of all service time incurred over the entire route. If **batched loads** is enabled, this measure is analogous for each of the vehicle's loads.

## 6. Idle Time

Idle time (wait time) is incurred by a vehicle if it arrives at a location outside of the location's time windows or if it arrives outside of the order's time windows. A vehicle's idle time is the sum of all idle time incurred over the entire route. If **batched loads** is enabled, this measure is analogous for each of the vehicle's loads.

#### 7. Break Time

Break time is incurred if the vehicle has a fatigue management scheme specified (See Section 2.13). A vehicle's break time is the sum of all break time incurred over the entire route. If batched loads is enabled, this measure is analogous for each of the vehicle's loads.

# 2.21 Unassigned Orders

If the solution generated by the optimiser contains any orders that are not assigned to a vehicle, then the optimiser will carry out an analysis to try to determine the reasons for those orders not being assigned.

Examples of such reasons include:

- Orders not being compatible with any of the vehicles in the fleet.
- Orders have a weight or volume demand that exceeds the maximum vehicle weight or volume capacity of any vehicle in the fleet.
- Orders that need to be picked up or delivered *before* the start time of any of the vehicles in the fleet.
- Orders that require more time to complete than the maximum working time of any vehicle in the fleet.

Note that the focus of the analysis is on finding reasons that are likely to indicate errors or omissions in the input data. It is possible that the analysis will not find any definite reason for an order being unassigned.

# 2.22 Pickup Segments

A *pickup segment* is defined to be any two consecutive PICKUP stops in a route. (A route or load with only a single PICKUP stop has no pickup segments.)

The optimiser can optionally apply a penalty based on the pickup segments in a route. This penalty has two components: a *fixed* component based on the number of pickup segments in the route and a *variable* component proportional to the transit distance along the pickup segments. These penalties take the form of an additional hidden cost.

The pickup segment penalty is useful for "shaping" the route. The fixed component of the penalty may be used to reduce the number of pickup stops per load, while the variable component may be used to encourage the optimiser to cluster pickups more closely together.

The pickup segment penalty is only applied if the **Batched Loads** parameter is enabled.

# 2.23 Delivery Segments

A *delivery segment* is defined to be any two consecutive DELIVERY stops in a route. (A route or load with only a single DELIVERY stop has no delivery segments.)

The optimiser can optionally apply a penalty based on the delivery segments in a route. This penalty has two components: a *fixed* component based on the number of delivery segments in the route and a *variable* component proportional to the transit distance along the delivery segments. These penalties take the form of an additional hidden cost.

The delivery segment penalty is useful for "shaping" the route. The fixed component of the penalty may be used to reduce the number of delivery stops per load, while the variable component may be used to encourage the optimiser to cluster deliveries more closely together.

The delivery segment penalty is only applied if the **Batched loads** parameter is enabled.

# 2.24 Finish Segments

The *finish segment* of a route is defined as the part of the route from the last DELIVERY stop to the FINISH stop. There may be intervening BREAK stops between these two stops.

The optimiser can optionally apply a penalty based on the transit time or distance of the finish segment of a route. These penalties take the form of an additional hidden cost.

Penalties applied to the finish segment are used to tell the optimiser to prefer solutions in which vehicles end their routes "close" to their finish locations.

24

# 2.25 Loaded Distance Penalties

A *loaded distance penalty* is a hidden cost that is scaled by both the distance travelled by a vehicle and the amount loaded on the vehicle while travelling.

More precisely, the loaded distance penalty for a vehicle over a single travel segment is defined as:

 $TransitDistance \times LoadedQuantity \times PenaltyFactor$ 

where *TransitDistance* is the distance travelled in that segment (in either kilometres or miles), *LoadedQuantity* is the amount loaded on the vehicle over that segment and *PenaltyFactor* is a user-supplied value. The loaded distance penalty of a vehicle's route is the sum of the loaded distance penalties over all of its individual travel segments.

Loaded distance penalties can be applied to either the weight loaded on the vehicle, the volume loaded on the vehicle or both.

# 3 Microsoft Excel Input

This section describes the Microsoft Excel input format for the optimiser.

Excel input must be either an Office Open XML workbook (.xlsx) or macro-enabled workbook (.xlsm). Office Open XML binary workbooks (.xlsb) are *not* currently supported. Workbooks in the old Excel binary file format (.xls) are *not* supported.

The workbook should contain the sheets described in the following sections. Those sheets that are specified as "required" must be present in the workbook. The workbook may contain sheets other than those recognised by the DTO-Excel format. Such extra sheets are ignored by the optimiser.

Unless otherwise noted, sheets recognised by the optimiser may contain columns other than those required by the DTO-Excel format. Such extra columns are ignored by the optimiser.

Columns recognised by the optimiser may appear in any order.

All sheets must have a non-empty first row.

The optimiser will evaluate any formulas used in the workbook when it is loaded. Formulas must not refer to sheets in other workbooks.

Not all Excel functions are supported by the optimiser; appendix C lists the unsupported Excel functions.

# 3.1 Identifiers in Excel

Identifiers and attributes in DTO-Excel may be provided as either text or numeric data. Identifiers given in text cells are taken *as-is* subject only to the rules concerning case and "white space" insensitivity. Identifiers given in numeric cells are *only* accepted by the optimiser if they are integer (i.e. their fractional part is zero). Formulas that evaluate to either text or (integer) numeric data may also be used to specify identifiers in Excel.

# 3.2 Times in Excel

In Excel, times are *stored* as decimal numbers, where 0 is 00:00:00 and 1 is 24:00:00.

The integer component of the decimal represents a 24 hour period of time and is used to represent times greater than 23:59:59. When using this representation, it is useful to think of the integer component as the day on which the time occurs, with times less than 24:00:00 occurring on *day zero*. For example, 1.50 would be 36:00, or 12:00 on the second day.

In Excel, cells containing times may be formatted in various ways, for example, hh:mm, d hh:mm or h:mm AM/PM. Times may be given to the optimiser in any of the Excel time formats.

	hh:mm	[hh];mm	dd/mm/yyy hh:mm	d hh:mm
0.5	12:00	12:00	00/01/1900 12:00	0 12:00
1.5	12:00	36:00	01/01/1900 12:00	1 12:00

Table 2: Excel time formatting examples.

Note here that if using the dd/mm/yyy hh:mm format, day zero occurs on 00/01/1900, which is Excel's earliest recognised date.

# 3.3 Sheet: General

This sheet is required.

# 3.3.1 Column: Parameter

**Required:** Yes. **Type:** Text. **Description:** Parameters that control the settings of scenario.

# 3.3.2 Column: Value

Required: Yes.Type: Text or Number.Description: Values of the parameters. Can be a number or text depending on the parameter.

## 3.3.3 Parameter: Scenario Name

Required: No. Value Type: Text. Default: No name. Description: A name for the scenario.

## 3.3.4 Parameter: Iterations

Required: No.
Value Type: Number (positive integer).
Default: 3000.
Description: The number of iterations the optimiser will run for. (This parameter is ignored if the Iteration Scheme parameter is also specified.)

## 3.3.5 Parameter: Rush Hour

Required: No. Value Type: Text (Boolean). Default: FALSE. Description: If TRUE, then use the rush hour periods defined in the Rush Hour sheet.

## 3.3.6 Parameter: Use Miles

Required: No. Value Type: Text (Boolean). Default: FALSE. Description: If TRUE, then distance values in the solution will be reported in miles instead of kilometres.
#### 3.3.7 Parameter: Iteration Scheme

Required: No.Value Type: Text.Default: No iterations scheme used.Description: The identifier of an iteration scheme defined in Iteration Schemes sheet that the optimiser should use.

#### 3.3.8 Parameter: Colocated Pickups

Required: No. Value Type: Text (Boolean). Default: FALSE. Description: If TRUE, then enforce the Colocated Pickups constraint (Section 2.8).

## 3.3.9 Parameter: Geofence North

Required: No. Value Type: Number in [-90, 90]. Default: none. Description: Latitude of the northern boundary for locations in scenario.

## 3.3.10 Parameter: Geofence East

Required: No. Value Type: Number in [-180, 180]. Default: none. Description: Longitude of the eastern boundary for locations in scenario.

## 3.3.11 Parameter: Geofence South

Required: No. Value Type: Number in [-90, 90]. Default: none. Description: Latitude of the southern boundary for locations in scenario.

#### 3.3.12 Parameter: Geofence West

Required: No. Value Type: Number in [-180, 180]. Default: none. Description: Longitude of the western boundary for locations in scenario.

## 3.3.13 Parameter: Hard Geofence Errors

Required: No. Value Type: Boolean. Default: FALSE. Description: If TRUE, then treat any errors produced by geofence violations as being hard.

#### 3.3.14 Parameter: Batched Loads

Required: No. Value Type: Text (Boolean). Default: TRUE. Description: If TRUE, then enforce the batched loads constraint.

## 3.3.15 Parameter: Sticky Deliveries

Required: No. Value Type: Text (Boolean). Default: FALSE. Description: If TRUE, then enable the Sticky Deliveries constraint.

#### 3.3.16 Parameter: Cost Scale

Required: No.
Value Type: Number (positive).
Default: 0 (no scaling).
Description: Multiply all cost values by this number before solving. (See Scaling Factors in Section 2.8.)

#### 3.3.17 Parameter: Weight Scale

Required: No.
Value Type: Number (positive).
Default: 0 (no scaling).
Description: Multiply all weight values by this number before solving. (See Scaling Factors in Section 2.8.)

#### 3.3.18 Parameter: Volume Scale

Required: No.
Value Type: Number (positive).
Default: 0 (no scaling).
Description: Multiply all volume values by this number before solving. (See Scaling Factors in Section 2.8.)

#### 3.3.19 Parameter: Maximum Neighbourhood Size

Required: No. Value Type: Number (positive integer). Default: 200. Description: The value of the maximum neighbourhood size for the optimiser. (See Maximum Neighbourhood Size in Section 2.8.)

#### 3.3.20 Parameter: Disable Replanning

Required: No. Value Type: Text (Boolean). Default: FALSE. Description: If TRUE, ignore the Previous Solution sheet if it is present.

REGISTERED OFFICE: LEVEL 1, 18 KAVANAGH ST, SOUTHBANK, VIC, 3006

ABN: 13 146 662 053

## 3.3.21 Parameter: Previous-Vehicle Fixed Reward

Required: No. Value Type: Number (nonnegative). Default: 0. Description: the strength of the preference for orders to remain on their previously assigned vehicles when replanning.

#### 3.3.22 Parameter: Previous-Vehicle Weight Reward

Required: No.Value Type: Number (nonnegative).Default: 0.Description: the strength of preference for heavier orders to remain on their previously assigned vehicles when replanning.

## 3.3.23 Parameter: Previous-Vehicle Volume Reward

Required: No. Value Type: Number (nonnegative). Default: 0. Description: the strength of preference for larger orders to remain on their previously assigned vehicles when replanning.

## 3.3.24 Parameter: Forbid Unrecognised Columns

Required: No. Value Type: Boolean. Default: FALSE. Description: If TRUE, then treat any unrecognised column as an error.

#### 3.3.25 Parameter: Arrival Only in Time Window

Required: No. Value Type: Boolean. Default: FALSE. Description: If TRUE, the optimiser will enable the Arrival Only in Time Window behaviour.

# **3.4 Sheet: Locations**

This sheet must be defined and contain at least one location.

## 3.4.1 Column: Id

Required: Yes. Type: Text or (integer) number. Description: A unique identifier for the location.

# 3.4.2 Column: Name

Required: No. Type: Text. Description: A name for the location (e.g. a store or facility name).

# 3.4.3 Column: Attributes

Required: No. Type: Text or (integer) number. Description: Multiple attributes should be separated by semicolons. (Multiple attributes can only be specified in a text cell.)

# 3.4.4 Column: Address

Required: No. Type: Any. Description: The full street address of the location.

## 3.4.5 Column: Longitude

Required: Yes.Type: Number.Description: Longitude of the location in decimal degrees between -180 and 180.

## 3.4.6 Column: Latitude

Required: Yes. Type: Number. Description: Latitude of the location in decimal degrees between -90 and 90.

## 3.4.7 Column: Site Time

Required: No. Type: Text (*HH:MM*). Default: 00:00. Description: Minimum time spent at the location. (See Site Time in Section 2.10.)

30

## 3.4.8 Column: Load Time

Required: No. Type: Text (*HH:MM*). Default: 00:00. Description: Time spent for loading at the location. (See Load Time in Section 2.10.)

## 3.4.9 Column: Unload Time

Required: No. Type: Text (*HH:MM*). Default: 00:00. Description: Time spent for unloading at the location. (See Unload Time in Section 2.10.)

#### 3.4.10 Column: Opening Time

Required: No. Type: Excel time. Default: 00:00. Description: Earliest opening time for the location.

## 3.4.11 Column: Closing Time

Required: No. Type: Excel time. Default: No closing time. Description: Latest closing time for the location.

#### 3.4.12 Column: Operating Time Windows

Required: No.
Type: Text or (integer) number.
Default: No time windows.
Description: The identifier of the time window as defined in the Time Windows sheet. (See Location Time Windows in Section 2.10).

#### 3.4.13 Column: Cost per Visit

Required: No. Type: Number. Default: 0. Description: The cost incurred when a vehicle visits this location. This may be negative to represent gains associated with visiting the location. (See Cost per Visit in Section 2.10.)

#### 3.4.14 Column: Hidden Cost per Visit

Required: No.
Type: Number.
Default: 0.
Description: An additional cost incurred when a vehicle visits this location. This may be negative and is not included in the cost calculation for the solution. (See Hidden Cost per Visit in Section 2.10.)

# 3.5 Sheet: Orders

This sheet must be defined and contain at least one order.

## 3.5.1 Column: Id

Required: Yes.Type: Text or (integer) number.Description: A unique identifier for the order. An order's identifier also acts as an attribute name.

## 3.5.2 Column: Name

Required: No. Type: Text. Description: A descriptive name for the order (e.g. a purchase order).

## 3.5.3 Column: Pickup Location

Required: Yes. Type: Text or (integer) number. Description: An identifier from the Locations sheet.

## 3.5.4 Column: Pickup Service Time

Required: No. Type: Excel time. Default: 00:00. Description: A nonnegative time. (See Service Time in Section 2.12.)

## 3.5.5 Column: Earliest Pickup Time

Required: No. Type: Excel time. Default: 00:00. Description: A nonnegative time.

#### 3.5.6 Column: Latest Pickup Time

Required: No. Type: Excel time. Default: Inferred from data. Description: A nonnegative time.

#### 3.5.7 Column: Pickup Time Windows

Required: No. Type: Text. Default: Can occur at any time subject to the pickup location's and vehicles' time windows. Description: The identifier of the time window as defined in the Time Windows sheet. (See Order Time Windows in section 2.12.)

## 3.5.8 Column: Pickup Soft Time Windows

Required: No.
Type: Text.
Default: Can occur at any time subject to the pickup location's, order's and vehicles' time windows.
Description: The identifier of the time window as defined in the Time Windows sheet.

## 3.5.9 Column: Soft Pickup Constant

Required: No. Type: Number. Default: 0. Description: The fixed penalty for the order's pickup soft time windows.

## 3.5.10 Column: Soft Pickup Variable

Required: No. Type: Number. Default: 0. Description: The variable penalty for the order's pickup soft time windows.

## 3.5.11 Column: Delivery Location

Required: Yes. Type: Text or (integer) number. Description: A location identifier from the Locations sheet.

#### 3.5.12 Column: Delivery Service Time

Required: No. Type: Excel time. Default: 00:00. Description: A nonnegative time. (See Service Time in Section 2.12.)

#### 3.5.13 Column: Earliest Delivery Time

Required: No. Type: Excel time. Default: 00:00. Description: A nonnegative time.

#### 3.5.14 Column: Latest Delivery Time

Required: No. Type: Excel time. Default: No latest delivery time. Description: A nonnegative time.

## 3.5.15 Column: Delivery Time Windows

Required: No. Type: Text. Default: Can occur at any time subject to locations' and vehicles' time windows. Description: The identifier of the time window as defined in the Time Windows sheet. (See Order Time Windows in section 2.12.)

## 3.5.16 Column: Delivery Soft Time Windows

Required: No.Type: Text.Default: Can occur at any time subject to the delivery location's, order's and vehicles' time windows.Description: The identifier of the time window as defined in the Time Windows sheet.

## 3.5.17 Column: Soft Delivery Constant

Required: No. Type: Number. Default: 0. Description: The fixed penalty for any pickup soft time windows associated with the order.

## 3.5.18 Column: Soft Delivery Variable

Required: No. Type: Number. Default: 0. Description: The variable penalty for any pickup soft time windows associated with the order.

#### 3.5.19 Column: Maximum Duration

Required: No. Type: Excel time. Default: No maximum duration. Description: The maximum duration for the order.

#### 3.5.20 Column: Weight

Required: No, but if omitted then the Volume column *must* be present.Type: Number.Default: 0.Description: Weight of the order with units consistent with vehicles' weight capacity.

#### 3.5.21 Column: Volume

Required: No, but if omitted then the Weight column *must* be present.Type: Number.Default: 0.Description: Volume of the order with units consistent with vehicles' volume capacity.

## 3.5.22 Column: Assign Cost

Required: No.
Type: Number.
Default: 0.
Description: The cost incurred when servicing the order. This can be negative to represent gains associated with servicing the order. (See Assign Cost in Section 2.12.)

## 3.5.23 Column: Hidden Assign Cost

Required: No.
Type: Number.
Default: 0.
Description: Additional fixed cost to service the order. This can be negative and is not included in the cost calculation for the solution and may be used to prioritise certain orders. (See Hidden Assign Cost in Section 2.12.)

## 3.5.24 Column: Attributes

**Required:** No. **Type:** Text or (integer) number.

**Description:** Multiple attributes should be separated by semicolons. (Multiple attributes can only be specified in a text cell.)

# 3.6 Sheet: Vehicle Type

The **Vehicle Type** sheet may be used to define a **type** of vehicle It may be omitted if vehicle types are not used.

#### 3.6.1 Column: Id

**Required:** Yes. **Type:** Text or (integer) number. **Description:** Each vehicle type must be have a unique identifier.

## 3.6.2 Column: Start Location

Required: No.Type: Text or (integer) number.Default: The vehicle begins its route at its first pickup location.Description: The identifier of the location in the Locations sheet where the vehicle starts its route.

## 3.6.3 Column: Finish Location

Required: No. Type: Text or (integer) number.

**Default:** The vehicle finishes its route at its last delivery location.

**Description:** The identifier of the location in the Locations sheet where the vehicle *must* end its route. The transit time and distance associated with returning to this location after the last delivery are considered by the solution.

#### 3.6.4 Column: Maximum Weight

Required: Yes. Type: Number. Default: 0. Description: The maximum weight the vehicle can carry. Units are unspecified but must be consistent with orders' weights. Must be nonnegative.

## 3.6.5 Column: Maximum Volume

Required: Yes.Type: Number.Default: 0.Description: The maximum volume the vehicle can carry. Units are unspecified but must be consistent with orders' volumes. Must be nonnegative.

#### 3.6.6 Column: Earliest Start Time

Required: No. Type: Text (*HH:MM*). Default: 00:00. Description: Earliest time a vehicle can start at the start location.

16 June 2025

37

## 3.6.7 Column: Latest Start Time

Required: No. Type: Excel time. Default: No latest start time. Description: Latest time a vehicle can start at the start location.

## 3.6.8 Column: Latest Finish Time

Required: No. Type: Excel time. Default: No latest finish time. Description: Latest time a vehicle must arrive at the finish location.

## 3.6.9 Column: Minimum Paid Time

Required: No. Type: Excel time. Default: 00:00. Description: A minimum amount of time that this vehicle's cost-per-hour will be incurred. If the vehicle's actual working time is less than this, then the hourly cost will be this value multiplied by the cost-per-hour instead.

## 3.6.10 Column: Maximum Drive Time

Required: No. Type: Excel time. Default: Inferred from data. Description: Maximum time a vehicle can drive for. Must be nonnegative.

## 3.6.11 Column: Maximum Work Time

Required: No.Type: Excel time.Default: Inferred from data.Description: Maximum time a vehicle can work for, calculated from the start time to finish time.Must be nonnegative.

## 3.6.12 Column: Load Time

Required: No. Type: Excel time. Default: 00:00. Description: Time needed to load vehicle. (See Load Time in Section 2.11.) Must be nonnegative.

## 3.6.13 Column: Unload Time

Required: No. Type: Excel time. Default: 00:00. Description: Time needed to unload the vehicle. (See Unload Time in Section 2.11.) Must be nonnegative.

#### 3.6.14 Column: Speed Scale

Required: No. Type: Number. Default: 0 (means no speed scale). Description: Nonnegative speed scale multiplier for the vehicle. (See Speed Scale in Section 2.11.)

#### 3.6.15 Column: Cost per Use

Required: No. Type: Number. Default: 0. Description: A fixed cost for using the vehicle.

## 3.6.16 Column: Cost per Hour

Required: No. Type: Number. Default: 0. Description: Cost incurred per hour of vehicle's work time.

## 3.6.17 Column: Cost per Kilometre

Required: No.
Type: Number.
Default: 0.
Description: Cost incurred per kilometre of transit distance.
Note: If the parameter Use Miles is TRUE in the General sheet then this column must *not* be present.

#### 3.6.18 Column: Cost per Mile

Required: No.
Type: Number.
Default: 0.
Description: Cost incurred per mile of transit distance.
Note: If the parameter Use Miles is FALSE in the General sheet then this column must *not* be present.

## 3.6.19 Column: Cost per Load

Required: No. Type: Number. Default: 0. Description: Cost incurred for each load. Only applied if batched loads is enabled.

## 3.6.20 Column: Hidden Cost per Use

Required: No.
Type: Number.
Default: 0.
Description: Hidden "cost" for using the vehicle. This is not included in the cost calculation for the solution and could be used to prioritise certain vehicles.

#### 3.6.21 Column: Hidden Cost per Hour

Required: No.Type: Number.Default: 0.Description: Additional cost incurred per hour of vehicle use. This is not included in the cost calculation for the solution and could be used to prioritise vehicles.

#### 3.6.22 Column: Hidden Cost per Kilometre

Required: No. Type: Number. Default: 0.

**Description:** Additional cost incurred per km of vehicle use. This is not included in the cost calculation for the solution and could be used to prioritise certain vehicles.

**Note:** If the parameter Use Miles is TRUE in the General sheet then this column must *not* be present.

#### 3.6.23 Column: Hidden Cost per Mile

Required: No. Type: Number.

Default: 0.

**Description:** Additional cost incurred per mile of vehicle use. This is not included in the cost calculation for the solution and could be used to prioritise certain vehicles.

Note: If the parameter Use Miles is FALSE in the General sheet then this column cannot be present.

#### 3.6.24 Column: Hidden Cost per Load

Required: No.Type: Number.Default: 0.Description: Additional cost incurred for each load. This is not included in the cost calculation for the solution. Only applied if batched loads is enabled.

## 3.6.25 Column: Finish Segment Distance Penalty

Required: No. Type: Number. Default: 0. Description: The per-unit distance component of the finish segment penalty for this vehicle. Must be nonnegative.

#### 3.6.26 Column: Finish Segment Per Hour Penalty

Required: No.Type: Number.Default: 0.Description: The per hour component of the finish segment penalty for this vehicle. Must be nonnegative.

## 3.6.27 Column: Pickup Segment Fixed Penalty

Required: No. Type: Number. Default: 0. Description: The fixed component of the pickup segment penalty for this vehicle type. Must be nonnegative.

#### 3.6.28 Column: Pickup Segment Distance Penalty

Required: No. Type: Number. Default: 0. Description: The per-unit distance component of the pickup segment penalty for this vehicle type. Must be nonnegative.

#### 3.6.29 Column: Delivery Segment Fixed Penalty

Required: No. Type: Number. Default: 0. Description: The fixed component of the delivery segment penalty for this vehicle type. Must be nonnegative.

#### 3.6.30 Column: Delivery Segment Distance Penalty

Required: No. Type: Number. Default: 0. Description: The per-unit distance component of the delivery segment penalty for this vehicle type. Must be nonnegative.

## 3.6.31 Column: Loaded Weight Distance Penalty

Required: No. Type: Number. Default: 0. Description: The weight scaled loaded distance penalty factor for this vehicle type. Must be nonnegative.

#### 3.6.32 Column: Loaded Volume Distance Penalty

Required: No. Type: Number. Default: 0. Description: The volume scaled loaded distance penalty factor for this vehicle type. Must be nonnegative.

## 3.6.33 Column: Attributes

**Required:** No. **Type:** Text or (integer) number. **Description:** Multiple attributes should be separated by semicolons.

### 3.6.34 Column: Break Scheme

Required: No. Type: Text or (integer) number. Description: The break scheme requirements for the vehicle. Must match a break scheme identifier defined in the Fatigue Management sheet.

#### 3.6.35 Column: Forbid Loaded Breaks

Required: No. Type: Boolean. Description: If set to TRUE, then loaded breaks will not be scheduled for this vehicle type.

#### 3.6.36 Column: Maximum Loads

Required: No.
Type: Number (positive integer).
Default: 0 (means no constraint on the maximum number of loads).
Description: The maximum number of loads this vehicle type can do. Must be nonnegative. This field is only applicable if batched loads has been enabled.

## 3.6.37 Column: Compartments

Required: No. Type: Text or (integer) number. Description: The compartment scheme for the vehicle type. If given, must match the identifier of a compartment scheme defined in the Compartments sheet.

# 3.7 Sheet: Fleet

This sheet must be defined and contain at least one vehicle.

## 3.7.1 Column: Id

Required: Yes. Type: Text or (integer) number. Description: Each vehicle must be have a unique identifier. Note that a vehicle's identifier can also function as an attribute name.

## 3.7.2 Column: Vehicle Type

Required: No.Type: Text or (integer) number.Description: The identifier of a vehicle type defined in the Vehicle Type sheet.

## 3.7.3 Column: Name

Required: No. Type: Text. Default: No name. Description: A descriptive name for the vehicle.

## 3.7.4 Column: Start Location

Required: No.Type: Text or (integer) number.Default: The vehicle begins its route at its first pickup location.Description: The identifier of the location in the Locations sheet where the vehicle starts its route.

#### 3.7.5 Column: Finish Location

Required: No.
Type: Text or (integer) number.
Default: The vehicle finishes its route at its last delivery location.
Description: The identifier of the location in the Locations sheet where the vehicle *must* end its route. The transit time and distance associated with returning to this location after the last delivery are considered by the solution.

#### 3.7.6 Column: Maximum Weight

Required: Yes.
Type: Number.
Default: 0.
Description: Maximum weight the vehicle can carry. Units are unspecified but must be consistent with orders' weights. Must be nonnegative.

## 3.7.7 Column: Maximum Volume

Required: Yes. Type: Number. Default: 0. Description: Maximum volume the vehicle can carry. Units are unspecified but must be consistent with orders' volumes. Must be nonnegative.

## 3.7.8 Column: Earliest Start Time

Required: No. Type: Text (*HH:MM*). Default: 00:00. Description: Earliest time a vehicle can start at the start location.

## 3.7.9 Column: Latest Start Time

Required: No. Type: Excel time. Default: No latest start time. Description: Latest time a vehicle can start at the start location.

## 3.7.10 Column: Latest Finish Time

Required: No. Type: Excel time. Default: No latest finish time. Description: Latest time a vehicle must arrive at the finish location.

#### 3.7.11 Column: Minimum Paid Time

Required: No. Type: Excel time. Default: 00:00. Description: A minimum amount of time that this vehicle's cost-per-hour will be incurred. If the vehicle's actual working time is less than this, then the hourly cost will be this value multiplied by the cost-per-hour instead.

#### 3.7.12 Column: Maximum Drive Time

Required: No. Type: Excel time. Default: Inferred from data. Description: Maximum time a vehicle can drive for. Must be nonnegative.

#### 3.7.13 Column: Maximum Work Time

Required: No.Type: Excel time.Default: Inferred from data.Description: Maximum time a vehicle can work for, calculated from the start time to finish time.Must be nonnegative.

REGISTERED OFFICE: LEVEL 1, 18 KAVANAGH ST, SOUTHBANK, VIC, 3006

ABN: 13 146 662 053

# 3.7.14 Column: Load Time

Required: No. Type: Excel time. Default: 00:00. Description: Time needed to load vehicle. (See Load Time in Section 2.11.) Must be nonnegative.

## 3.7.15 Column: Unload Time

Required: No. Type: Excel time. Default: 00:00. Description: Time needed to unload the vehicle. (See Unload Time in Section 2.11.) Must be nonnegative.

# 3.7.16 Column: Speed Scale

Required: No. Type: Number. Default: 0 (means no speed scale). Description: Nonnegative speed scale multiplier for the vehicle. (See Speed Scale in Section 2.11.)

## 3.7.17 Column: Cost per Use

Required: No. Type: Number. Default: 0. Description: A fixed cost for using the vehicle.

# 3.7.18 Column: Cost per Hour

Required: No. Type: Number. Default: 0. Description: Cost incurred per hour of vehicle's work time.

## 3.7.19 Column: Cost per Kilometre

Required: No.
Type: Number.
Default: 0.
Description: Cost incurred per kilometre of transit distance.
Note: If the parameter Use Miles is TRUE in the General sheet then this column must *not* be present.

## 3.7.20 Column: Cost per Mile

Required: No.
Type: Number.
Default: 0.
Description: Cost incurred per mile of transit distance.
Note: If the parameter Use Miles is FALSE in the General sheet then this column must *not* be present.

#### 3.7.21 Column: Cost per Load

Required: No. Type: Number. Default: 0. Description: Cost incurred for each load. Only applied if batched loads is enabled.

## 3.7.22 Column: Hidden Cost per Use

Required: No.Type: Number.Default: 0.Description: Hidden "cost" for using the vehicle. This is not included in the cost calculation for the solution and could be used to prioritise certain vehicles.

## 3.7.23 Column: Hidden Cost per Hour

Required: No.Type: Number.Default: 0.Description: Additional cost incurred per hour of vehicle use. This is not included in the cost calculation for the solution and could be used to prioritise vehicles.

#### 3.7.24 Column: Hidden Cost per Kilometre

Required: No. Type: Number. Default: 0.

**Description:** Additional cost incurred per km of vehicle use. This is not included in the cost calculation for the solution and could be used to prioritise certain vehicles.

Note: If the parameter Use Miles is TRUE in the General sheet then this column must *not* be present.

#### 3.7.25 Column: Hidden Cost per Mile

Required: No.Type: Number.Default: 0.Description: Additional cost incurred per mile of vehicle use. This is not included in the cost calculation for the solution and could be used to prioritise certain vehicles.

Note: If the parameter Use Miles is FALSE in the General sheet then this column cannot be present.

## 3.7.26 Column: Hidden Cost per Load

Required: No.Type: Number.Default: 0.Description: Additional cost incurred for each load. This is not included in the cost calculation for the solution. Only applied if batched loads is enabled.

## 3.7.27 Column: Finish Segment Distance Penalty

Required: No. Type: Number. Default: 0. Description: The per-unit distance component of the finish segment penalty for this vehicle. Must be nonnegative.

#### 3.7.28 Column: Finish Segment Per Hour Penalty

Required: No. Type: Number. Default: 0. Description: The per hour component of the finish segment penalty for this vehicle. Must be nonnegative.

## 3.7.29 Column: Delivery Segment Fixed Penalty

Required: No. Type: Number. Default: 0. Description: The fixed component of the delivery segment penalty for this vehicle. Must be nonnegative.

#### 3.7.30 Column: Delivery Segment Distance Penalty

Required: No. Type: Number. Default: 0. Description: The per-unit distance component of the delivery segment penalty for this vehicle. Must be nonnegative.

## 3.7.31 Column: Loaded Weight Distance Penalty

Required: No. Type: Number. Default: 0. Description: The weight scaled loaded distance penalty factor for this vehicle. Must be nonnegative.

#### 3.7.32 Column: Loaded Volume Distance Penalty

Required: No. Type: Number.

## Default: 0.

**Description:** The volume scaled loaded distance penalty factor for this vehicle. Must be nonnegative.

## 3.7.33 Column: Attributes

**Required:** No. **Type:** Text or (integer) number. **Description:** Multiple attributes must be separated by semicolons.

## 3.7.34 Column: Break Scheme

Required: No. Type: Text or (integer) number. Description: The break scheme requirements for the vehicle. Must match a break scheme identifier defined in the Fatigue Management sheet.

## 3.7.35 Column: Forbid Loaded Breaks

Required: No. Type: Boolean. Description: If set to TRUE, then loaded breaks will not be scheduled for the vehicle.

## 3.7.36 Column: Maximum Loads

**Required:** No. **Type:** Number (positive integer).

**Default:** 0 (means no constraint on the maximum number of loads).

**Description:** The maximum number of loads the vehicle can do. Must be nonnegative. This field is only applicable if **batched loads** has been enabled.

#### 3.7.37 Column: Maximum Drops per Load

Required: No.
Type: Number (positive integer).
Default: 0 (means no constraint on the maximum number of drops per load).
Description: The maximum number of delivery locations the vehicle can visit on a given load.
Must be nonnegative. This field is only applicable if batched loads has been enabled.

## 3.7.38 Column: Compartments

Required: No.

Type: Text or (integer) number.

**Description:** The compartment scheme for the vehicle. If given, must match the identifier of a compartment scheme defined in the Compartments sheet.

# 3.8 Sheet: Fatigue Management

This sheet may be omitted if no fatigue management schemes are required. (See Section 2.13 for further information.)

## 3.8.1 Column: Id

**Required:** Yes. **Type:** Text or (integer) number. **Description:** Each fatigue management scheme must be have a unique identifier.

## 3.8.2 Column: Number

Required: Yes.Type: Number (integer).Description: The number of breaks required by the scheme in the given interval in the Interval column. Must be nonnegative.

## 3.8.3 Column: Duration

Required: Yes. Type: Excel time. Description: The duration of the break. Must be nonnegative.

## 3.8.4 Column: Interval

Required: Yes. Type: Excel time. Description: The interval in which the number of blocks specified in Duration must occur as breaks. Must be nonnegative.

# 3.9 Sheet: Time Windows

The **Time Windows** sheet may be used to define groups of time windows that can be associated with pickups, deliveries and locations.

This sheet can be omitted if no named groups of time windows are required.

## 3.9.1 Column: Id

Required: Yes.Type: Text or (integer) number.Description: An identifier giving the named group of time windows that this window is part of.

# 3.9.2 Column: Start

Required: Yes.Type: Excel time.Description: Starting time of the window. Must occur before the End time and be nonnegative.

## 3.9.3 Column: End

Required: Yes. Type: Excel time. Description: Finishing time of the window. Must occur after the Start time and be nonnegative.

49

# 3.10 Sheet: Compartments

This sheet may be omitted if no vehicle has compartments.

## 3.10.1 Column: Scheme Id

Required: Yes. Type: Text or (integer) number. Description: A unique identifier for the compartment scheme.

## 3.10.2 Column: Id

Required: Yes.Type: Text or (integer) number.Description: A unique identifier for the compartment. Must be unique within the compartment scheme.

## 3.10.3 Column: Name

Required: No. Type: Text. Description: A descriptive name for the compartment. Must be unique within the compartment scheme.

#### 3.10.4 Column: Maximum Weight

Required: Yes.Type: Number.Description: The maximum weight that the compartment can carry. Units are unspecified but must be consistent with the orders' volumes. Must be nonnegative.

#### 3.10.5 Column: Maximum Volume

Required: Yes. Type: Number. Description: The maximum volume that the compartment can carry. Units are unspecified but must be consistent with the orders' volumes. Must be nonnegative.

## 3.10.6 Column: Attributes

**Required:** No. **Type:** Text or (integer) number. **Description:** Multiple attributes must be separated by semicolons.

# 3.11 Sheet: Rush Hour

The **Rush Hour** sheet is used to define time periods where speed scaling should take effect. Note that this feature must be enabled by setting the value of the **Rush Hour** parameter to **TRUE**.

This sheet can be omitted if rush hour periods are not required.

# 3.11.1 Column: Start

**Required:** Yes. **Type:** Excel time. **Description:** Starting time of the rush hour period.

# 3.11.2 Column: End

Required: Yes. Type: Excel time. Description: Finishing time of the rush hour period. Must occur after Start Time and be nonnegative.

## 3.11.3 Column: Speed Scale

Required: Yes.Type: Number.Description: The speed scaling factor to apply to transit times within the rush hour period.

# 3.12 Sheet: Iteration Schemes

The Iteration Schemes sheet is used to define iteration schemes.

Each row in this sheet defines an iteration scheme batch. The order of the batches in each scheme is defined by the order of the rows. Note that this sheet can define multiple iteration schemes, although the optimiser can only use one at a time.

This sheet can be omitted.

### 3.12.1 Column: Id

**Required:** Yes. **Type:** Text or (integer) number. **Description:** The identifier of the iteration scheme to which this batch belongs.

## 3.12.2 Column: Iterations

Required: Yes. Type: Number (integer). Description: The nonnegative number of iterations in the batch.

## 3.12.3 Column: Maximum Neighbourhood Size

Required: Yes. Type: Number (integer). Description: The maximum size of the neighbourhood of the batch. Must be at least 10.

# 3.13 Sheet: Order-Vehicle Compatibility

This sheet may be omitted if *order-vehicle* compatibility is not needed. It takes the form of a matrix where the first column specifies the order attributes and the first row specifies the vehicle attributes. The other cells define the compatibility: they should be TRUE or empty (i.e. FALSE). The upper-left corner of the matrix (i.e. cell A1) should be empty.

Note that this sheet may not contain extra columns that are *not* recognised by the optimiser.

# 3.13.1 Column A: Order Attribute

## Required: Yes.

Type: Text or (integer) number.

**Description:** Order attributes that are needed in the *order-vehicle* compatibility are listed on the column A, starting from the row 2.

## 3.13.2 Row 1: Vehicle Attribute

Required: Yes.

**Type:** Text or (integer) number. **Description:** Vehicle attributes that are needed in the *order-vehicle* compatibility are listed on the row 1, starting from the column B.

## 3.13.3 All other cells (except A1)

**Required:** Yes (only for TRUE).

Type: Boolean.

**Description:** If the corresponding order attribute (on column A) and vehicle attribute (on row 1) are compatible, the cell value is TRUE; otherwise, empty.

# 3.14 Sheet: Order-Compartment Compatibility

This sheet may be omitted if *order-compartment* compatibility is not needed. It is in matrix format with the first column for order attributes, the first row for compartment attributes, the upper-left corner being empty and all other cells being TRUE or empty.

Note that this sheet may not contain extra columns that are *not* recognised by the optimiser.

## 3.14.1 Column A: Order Attribute

## Required: Yes.

Type: Text or (integer) Number.

**Description:** Order attributes that are needed in the *order-compartment* compatibility are listed on the column A, starting from row 2.

## 3.14.2 Row 1: Compartment Attribute

## Required: Yes.

**Type:** Text or (integer) Number.

**Description:** Compartment attributes that are needed in the *order-compartment* compatibility are listed on the row 1, starting from column B.

## 3.14.3 All other cells (except A1)

**Required:** Yes (only for TRUE).

Type: Boolean.

**Description:** If the corresponding order attribute (on column A) and compartment attribute (on row 1) are compatible, the cell value is TRUE; otherwise, empty.

# 3.15 Sheet: Order-Order Compatibility

This sheet may be omitted if *order-order* compatibility is not needed. It is in matrix format with both the first column and the first row for order attributes, the upper-left corner being empty and all other cells being TRUE or empty.

Note that this sheet may not contain extra columns that are *not* recognised by the optimiser.

## 3.15.1 Column A: Order Attribute

Required: Yes.

Type: Text or (integer) Number.

**Description:** Order attributes that are needed in the *order-order* compatibility are listed on the column A, starting from the row 2.

## 3.15.2 Row 1: Order Attribute

Required: Yes.

Type: Text or (integer) Number. Description: Starting from the column B, the row 1 has the same sequence of order attributes as the one on the column A.

## 3.15.3 All other cells (except A1)

**Required:** Yes (only for TRUE and the upper triangular matrix). **Type:** Boolean.

**Description:** If the corresponding order attribute (on column A) and order attribute (on row 1) are compatible, the cell value is TRUE; otherwise, empty. Since the matrix is symmetrical, only the values of the diagonal and above the diagonal are read by the optimiser.

# 3.16 Sheet: Location-Vehicle Compatibility

This sheet may be omitted if location-vehicle compatibility is not needed.

## 3.16.1 Cell A1

Required: No. Type: any type. Description: The content is not read by the DTO, so any content is acceptable, for example "Vehicle\Location".

## 3.16.2 Column A: Location Attribute

Required: Yes.Type: Text.Description: Location attributes that are needed in the location-vehicle compatibility are listed on the column A, starting from the row 2.

# 3.16.3 Row 1: Vehicle Attribute

Required: Yes. Type: Text. Description: Vehicle attributes that are needed in the location-vehicle compatibility are listed on the row 1, starting from the column B.

# 3.16.4 All other cells

Required: Yes (only for true). Type: Boolean.

**Description:** If the corresponding location attribute (on column A) and vehicle attribute (on row 1) are compatible, the cell value is **true**; otherwise, empty.

# 3.17 Sheet: Order-Vehicle Costs

This sheet is used to specify vehicle-specific assign costs and hidden assign costs for orders. This sheet may be omitted if *order-vehicle* costs are not needed.

# 3.17.1 Column: Order

**Required:** Yes. **Type:** Text or (integer) number. **Description:** The identifier of an order from the Orders sheet.

# 3.17.2 Column: Vehicle

**Required:** Yes. **Type:** Text or (integer) number. **Description:** The identifier of a vehicle from the **Fleet** sheet.

## 3.17.3 Column: Cost

Required: No.Type: Number.Default: 0.Description: The cost incurred if the order is assigned to the vehicle.

## 3.17.4 Column: Hidden Cost

Required: No. Type: Number. Default: 0. Description: The hidden cost incurred if the order is assigned to the vehicle.

# 3.18 Sheet: Order-Vehicle Preferences

This sheet is used to specify **preferred vehicles** for orders. This sheet may be omitted if *order-vehicle* preferences are not needed.

# 3.18.1 Column: Order

**Required:** Yes. **Type:** Text or (integer) number. **Description:** The identifier of an order from the Orders sheet.

# 3.18.2 Column: Vehicle

**Required:** Yes. **Type:** Text or (integer) number. **Description:** The identifier of a vehicle from the **Fleet** sheet.

# 3.18.3 Column: Fixed Reward

Required: No. Value Type: Number (positive integer). Description: The fixed reward for assigning the order to this vehicle.

## 3.18.4 Column: Weight Reward

Required: No. Value Type: Number (positive integer). Description: The weight reward for assigning the order to this vehicle.

## 3.18.5 Column: Volume Reward

Required: No. Value Type: Number (positive integer). Description: The volume reward for assigning the order to this vehicle.

# 3.19 Sheet: Order Precedences

This sheet is used to specify precedences between orders. This sheet may be omitted if order precedences are not needed.

## 3.19.1 Column: Before

**Required:** Yes. **Type:** Text or (integer) number. **Description:** The identifier of an order from the Orders sheet.

# 3.19.2 Column: After

Required: Yes. Type: Text or (integer) number. Description: The identifier of an order from the Orders sheet.

# 3.19.3 Column: Gap

Required: No.

Type: Excel time.

**Description:** The duration between the departure from the delivery **stop** of the *before* order and the arrival at the pickup **stop** of the *after* order.

**Default:** 00:00, meaning the pickup of the *after* order may occur immediately after the delivery of the *before* order.

# 3.20 Sheet: Previous Solution

This sheet is used to specify assignments and sequencing from a previous solution for replanning. This sheet can be omitted if replanning is not being used. It is ignored if the global parameter disable replanning is set to TRUE in the General sheet.

The columns in this sheet are a subset of those in the **Runsheet** sheet of DTO-Excel output workbooks. It is possible to copy that sheet directly from an output workbook. To facilitate this, the sheet name "Runsheet" is recognised as a synonym for "Previous Solution".

# 3.20.1 Column: Order

Required: Yes.Type: Text or (integer) number.Description: An identifier for an order. A row containing an identifier that is *not* in the Orders sheet will be ignored.

# 3.20.2 Column: Vehicle

Required: Yes.Type: Text or (integer) number.Description: An identifier for a vehicle. A row containing an identifier that is *not* in the Fleet sheet will be ignored.

## 3.20.3 Column: Stop

Required: Yes. Value Type: Text (stop type). Description: The type of stop. See Section 2.18 for further information.

# 3.21 Sheet: Time Matrix

The **Time Matrix** sheet may be used to define the transit times, in minutes, between each pair of locations in the Locations sheet.

This sheet is optional. If omitted, the optimiser will compute transit times.

## 3.21.1 The Matrix

**Description:** The matrix must be square and contain all locations provided in the Locations sheet. The location identifiers are used to reference the locations and the must be included. The row locations are listed in column A beginning in the second row, while the column locations in row 1 beginning in the second column. Note that negative values are not allowed and that blank cells will mean that it is impossible to transit between locations.

# 3.22 Sheet: Distance Matrix

The **Distance Matrix** sheet may be used to define the transit distances in **kilometres** between the locations given in the Locations sheet. Note that kilometres are used even if Use Miles is **true** in the General sheet.

This sheet is optional. If omitted, the optimiser will compute the transit distances.

# 3.22.1 The Matrix

**Description:** The matrix must be square and contain all locations provided in the Locations sheet. The location identifiers are used to reference the locations and the must be included. The row locations are listed in column A, beginning in the second row, while the column locations in row 1, beginning in the second column. Note that negative values are not allowed and that blank cells will mean that it is impossible to transit between locations.

# 3.23 Note On Matrices

Providing matrices are not necessary to use the DTO as they are automatically calculated if not provided. However, if you want to use your own then they should be defined here.

	$\mathbf{A}$	В	$\mathbf{C}$	D	$\mathbf{E}$	$\mathbf{F}$	$\mathbf{G}$
1		DEPOT	LOC1	LOC2	LOC3	LOC4	LOC5
<b>2</b>	DEPOT	0.00	15.27	12.90	4.00	4.68	4.68
3	LOC1	16.17	0.00	3.60	13.05	12.72	13.23
4	LOC2	12.57	2.97	0.00	9.47	9.12	9.63
5	LOC3	3.80	12.02	9.67	0.00	1.82	3.48
6	LOC4	4.78	11.77	9.40	1.97	0.00	1.72
7	LOC5	4.57	12.38	10.02	3.38	1.75	0.00

 Table 3: Matrix Example

# 4 Microsoft Excel Output

This section describes the Microsoft Excel output format for the optimiser.

The Excel output will contain references (e.g. location identifiers, order identifiers) to the scenario from which it was generated.

The optimiser generates output workbooks in Office OpenXML format (.xlsx).

## 4.1 Sheet: Runsheet

#### 4.1.1 Column: Vehicle

**Type:** Text. **Description:** The identifier of the vehicle as given in Id column of the Fleet sheet.

#### 4.1.2 Column: Load

**Type:** Number. **Description:** If **batched loads** is enabled, this is index of the current load.

#### 4.1.3 Column: Stop Number

Type: Number.

**Description:** Numerical identification of a stop generated by the optimiser. Stops are numbered from 0 in increasing order with each stop number being unique.

#### 4.1.4 Column: ETA

**Type:** Excel time. **Description:** The "Estimated Time of Arrival" of the vehicle at the location.

#### 4.1.5 Column: ETD

**Type:** Excel time. **Description:** The "Estimated Time of Departure" of the vehicle from the location.

#### 4.1.6 Column: Stop

**Type:** Text. **Description:** The type of stop. See Section 2.18 for further information.

#### 4.1.7 Column: Order

Type: Text.

**Description:** If the stop is a PICKUP or DELIVERY stop then this contains the identifier of the order in the Orders sheet. For START, FINISH and BREAK stops this will be blank.

#### 4.1.8 Column: Compartment

**Type:** Text. **Description:** If the vehicle has compartments and the stop is a PICKUP or DELIVERY stop then
this contains the identifier of the compartment that the order is assigned to. For START, FINISH and BREAK stops this will be blank.

## 4.1.9 Column: Address

Type: Text.

**Description:** The identifier of the stop location as defined in the Locations sheet.

## 4.1.10 Column: Volume

Type: Number.

**Description:** The total volume on the vehicle at the stop.

## 4.1.11 Column: Delta Volume

Type: Number.

**Description:** The change in volume at the stop. Positive values represent an increase in volume, while negative values represent a decrease.

## 4.1.12 Column: Weight

**Type:** Number. **Description:** The total weight on the vehicle at the stop.

## 4.1.13 Column: Delta Weight

Type: Number.

**Description:** The change in weight at the stop. Positive values represent an increase in weight, while negative values represent a decrease.

## 4.1.14 Column: Site Time

**Type:** Excel time. **Description:** The site time incurred at the stop. (See Site Time in Section 2.10.)

#### 4.1.15 Column: Load Time

**Type:** Excel time. **Description:** The load time incurred at the stop. (See Load Time in Section 2.10.)

## 4.1.16 Column: Unload Time

**Type:** Excel time. **Description:** The unload time incurred at the stop. (See Unload Time in Section 2.10.)

#### 4.1.17 Column: Service Time

**Type:** Excel time. **Description:** The service time incurred at the stop. (See Service Time in Section 2.12.)

## 4.1.18 Column: Transit Time

**Type:** Excel time.

**Description:** The transit time between the previous stop and this one.

## 4.1.19 Column: Transit Distance

**Type:** Number. **Description:** The transit distance between the previous stop and this one.

# 4.1.20 Column: Soft Time Window Start

**Type:** Excel time.

**Description:** If the order has a soft time window for the current stop type, then the start time of the window closest to the stop arrival time.

## 4.1.21 Column: Soft Time Window End

**Type:** Excel time.

**Description:** If the order has a soft time window for the current stop type, then the end time of the window closest to the stop arrival time.

## 4.1.22 Column: Soft Time Window Delta

Type: Excel time.

**Description:** If the order has a soft time window for the current stop type, then the time difference between the stop arrival time and the closest boundary of the time window defined by the Soft Time Window Start and Soft Time Window End columns.

# 4.2 Sheet: Vehicle Summary

This sheet is a summary of the Runsheet data for each vehicle.

## 4.2.1 Column: Vehicle

**Type:** Text. **Description:** The identifier of the vehicle as given by the Id column of the Fleet sheet.

## 4.2.2 Column: Used

**Type:** Text (Boolean). **Description:** TRUE if the vehicle has at least one order assigned to it.

#### 4.2.3 Column: Cost

**Type:** Number. **Description:** The cost incurred by the vehicle.

#### 4.2.4 Column: Peak Volume

**Type:** Number. **Description:** The highest volume achieved during the vehicle's route.

#### 4.2.5 Column: Total Volume

**Type:** Number. **Description:** The total volume carried during the vehicle's route.

#### 4.2.6 Column: Volume Capacity

**Type:** Number. **Description:** The maximum volume capacity of the vehicle as defined in the **Fleet** sheet.

#### 4.2.7 Column: Peak Weight

**Type:** Number. **Description:** The highest weight achieved during the vehicle's route.

#### 4.2.8 Column: Total Weight

**Type:** Number. **Description:** The total weight carried during the vehicle's route.

#### 4.2.9 Column: Weight Capacity

**Type:** Number. **Description:** The maximum weight capacity of the vehicle as defined in the **Fleet** sheet.

## 4.2.10 Column: Number Of Loads

#### Type: Number.

**Description:** If batched loads is enabled, this is the number of loads in the vehicle's route. (See Section 2.19.)

## 4.2.11 Column: Transit Distance

**Type:** Number. **Description:** The total transit distance of the vehicle in the route.

#### 4.2.12 Column: Loaded Distance

**Type:** Number. **Description:** The total transit distance of the vehicle while it is carrying at least one order.

## 4.2.13 Column: Empty Distance

**Type:** Number. **Description:** The total transit distance of the vehicle while it is empty.

#### 4.2.14 Column: Actual Start Time

**Type:** Excel time. **Description:** The time of arrival at the **START** stop in the route for the vehicle.

## 4.2.15 Column: Actual Finish Time

**Type:** Excel time. **Description:** The time of departure at the FINISH stop in the route for the vehicle.

#### 4.2.16 Column: Work Time

Type: Excel time. Description: The total time spent between the START and FINISH stops in the route for the vehicle.

#### 4.2.17 Column: Allowed Work Time

**Type:** Excel time.

**Description:** The maximum work time for the vehicle as defined in the Fleet sheet. See Vehicle Times for further information.

#### 4.2.18 Column: Transit Time

**Type:** Excel time. **Description:** The total time spent in transit by the vehicle.

## 4.2.19 Column: Allowed Transit Time

**Type:** Excel time.

**Description:** The maximum time that a vehicle can spend in transit during a route as defined in the Fleet sheet. See Vehicle Times for further information.

## 4.2.20 Column: Site Time

**Type:** Excel time.

**Description:** The total site time incurred by the vehicle. (See Site Time in Section 2.10.)

## 4.2.21 Column: Load Time

**Type:** Excel time. **Description:** The total load time incurred by the vehicle. (See Load Time in Section 2.10.)

## 4.2.22 Column: Unload Time

**Type:** Excel time. **Description:** The total unload time incurred by the vehicle. (See Unload Time in Section 2.10.)

#### 4.2.23 Column: Service Time

**Type:** Excel time. **Description:** The total service time incurred by the vehicle. (See Service Time in Section 2.12.)

#### 4.2.24 Column: Idle Time

**Type:** Excel time. **Description:** The total time spent idle by the vehicle.

#### 4.2.25 Column: Break Time

**Type:** Excel time. **Description:** The total time spent on breaks by the vehicle.

## 4.2.26 Column: Number Of Drops

**Type:** Number. **Description:** The number of deliveries such that consecutive deliveries at the same location are only counted once.

#### 4.2.27 Column: Number Of Orders

**Type:** Number. **Description:** The number of orders assigned to the vehicle.

#### 4.2.28 Column: Average Transit Speed

Type: Number.

**Description:** The average speed of the vehicle. This is derived by dividing the total transit distance by the total transit time. Units are in Km/h by default and Mph if Use Miles in the General sheet is TRUE.

#### 4.2.29 Column: Pickup Soft Time Window Early

**Type:** Excel time.

**Description:** The total time the vehicle serviced an order's pickup before the start of its closest soft time window.

## 4.2.30 Column: Pickup Soft Time Window Late

#### **Type:** Excel time.

**Description:** The total time the vehicle serviced an order's pickup after the end of its closest soft time window.

## 4.2.31 Column: Delivery Soft Time Window Early

Type: Excel time.

**Description:** The total time the vehicle serviced an order's delivery before the start of its closest soft time window.

## 4.2.32 Column: Delivery Soft Time Window Late

#### Type: Excel time.

**Description:** The total time the vehicle serviced an order's delivery after the end of its closest soft time window.

# 4.3 Sheet: Load Summary

If **batched loads** is enabled, this sheet is a summary of the Runsheet data for each load in each vehicle.

## 4.3.1 Column: Vehicle

**Type:** Text. **Description:** The identifier of the vehicle as given by the Id column of the Fleet sheet.

## 4.3.2 Column: Load Number

**Type:** Number. **Description:** The number of the load.

#### 4.3.3 Column: Peak Volume

**Type:** Number. **Description:** The highest volume achieved during the load.

## 4.3.4 Column: Volume Capacity

**Type:** Number. **Description:** The maximum volume capacity of the vehicle as defined in the **Fleet** sheet.

## 4.3.5 Column: Peak Weight

**Type:** Number. **Description:** The highest weight achieved during the load.

#### 4.3.6 Column: Weight Capacity

**Type:** Number. **Description:** The maximum weight capacity of the vehicle as defined in the **Fleet** sheet.

#### 4.3.7 Column: Loaded Distance

**Type:** Number. **Description:** The transit distance from the first pickup to the last delivery in the load.

## 4.3.8 Column: Empty Distance Before

## Type: Number.

**Description:** For the first load, this is the distance from the **START** stop to the first pickup; otherwise it is the distance from the final delivery of the previous load to the first pickup of this load.

## 4.3.9 Column: Empty Distance After

#### Type: Number.

**Description:** For the final load, this is the distance from the final delivery to the FINISH stop; otherwise it is the distance from the final delivery of this load to the first pickup of the next load.

## 4.3.10 Column: Work Time

**Type:** Excel time.

**Description:** The total time spent between the first stop and last stop in the load by the vehicle.

## 4.3.11 Column: Transit Time

**Type:** Excel time. **Description:** The time spent in transit by the vehicle for this load.

## 4.3.12 Column: Site Time

**Type:** Excel time. **Description:** The site time incurred by the vehicle for this load. (See Site Time in Section 2.10.)

#### 4.3.13 Column: Load Time

Type: Excel time.

**Description:** The total load time incurred by the vehicle for this load. (See Load Time in Section 2.10.)

## 4.3.14 Column: Unload Time

Type: Excel time.

**Description:** The total unload time incurred by the vehicle for this load. (See Unload Time in Section 2.10.)

## 4.3.15 Column: Service Time

**Type:** Excel time.

**Description:** The total service time incurred by the vehicle for this load. (See Service Time in Section 2.12.)

## 4.3.16 Column: Idle Time

**Type:** Excel time. **Description:** The time spent idle by the vehicle for this load.

## 4.3.17 Column: Break Time

**Type:** Excel time. **Description:** The time spent on breaks by the vehicle for this load.

## 4.3.18 Column: Number Of Drops

Type: Number.

**Description:** The number of deliveries in the load such that consecutive deliveries at the same location are only counted once.

70

## 4.3.19 Column: Average Transit Speed

#### Type: Number.

**Description:** The average speed of the vehicle during the load. This is derived by dividing the total transit distance by the total transit time. Units are in Km/h by default and Mph if Use Miles in the General sheet is true.

# 4.3.20 Column: Pickup Soft Time Window Early

## Type: Excel time.

**Description:** The total time the vehicle serviced an order's pickup before the start of its closest soft time window in the load.

# 4.3.21 Column: Pickup Soft Time Window Late

## **Type:** Excel time.

**Description:** The total time the vehicle serviced an order's pickup after the end of its closest soft time window in the load.

## 4.3.22 Column: Delivery Soft Time Window Early

#### **Type:** Excel time.

**Description:** The total time the vehicle serviced an order's delivery before the start of its closest soft time window in the load.

## 4.3.23 Column: Delivery Soft Time Window Late

#### Type: Excel time.

**Description:** The total time the vehicle serviced an order's delivery after the end of its closest soft time window in the load.

## 4.4 Sheet: Solution Summary

#### 4.4.1 Scenario Name

Type: Text.

**Description:** The name of the scenario from which the solution was generated. Matches the **name** field of the **General** sheet.

## 4.4.2 Total Iterations

**Type:** Number. **Description:** The actual number of iterations completed by the optimiser before termination.

## 4.4.3 Total Cost

**Type:** Number. **Description:** The total cost for all routes in the solution.

## 4.4.4 Total Distance

**Type:** Number. **Description:** The total transit distance of all routes in the solution.

## 4.4.5 Loaded Distance

Type: Number.

**Description:** The total transit distance travelled in all routes while the vehicles are carrying at least one order.

#### 4.4.6 Empty Distance

**Type:** Number. **Description:** The total transit distance travelled in all routes while the vehicles are empty.

## 4.4.7 Average Transit Speed (km/h)

Type: Number.

**Description:** The average speed of the vehicles in the solution. This is derived by dividing the total transit distance by the total transit time. Only present if Use Miles in the General sheet is false.

#### 4.4.8 Average Transit Speed (mph)

Type: Number.

**Description:** The average speed of the vehicles in the solution. This is derived by dividing the total transit distance by the total transit time. Only present if Use Miles in the General sheet is true.

#### 4.4.9 Loads

**Type:** Number. **Description:** If **batched loads** is true then the total number of loads in the solution.

## 4.4.10 Km Per Load

#### Type: Number.

**Description:** If batched loads is true and Use Miles is false then the average kilometres per load in the solution.

## 4.4.11 Miles Per Load

**Type:** Number. **Description:** If batched loads and Use Miles are both true then the average miles per load in the solution.

#### 4.4.12 Hours Per Load

**Type:** Excel time. **Description:** If **batched loads** is true then the average time per load in the solution.

## 4.4.13 Drops Per Load

Type: Number.

**Description:** If batched loads is true then the total average number of drops per load in the solution.

#### 4.4.14 Work Time

**Type:** Excel time. **Description:** The total work time for all routes in the solution.

#### 4.4.15 Transit Time

**Type:** Excel time. **Description:** The total transit time for all routes in the solution.

#### 4.4.16 Break Time

**Type:** Excel time. **Description:** The total break time for all routes in the solution.

#### 4.4.17 Idle Time

**Type:** Excel time. **Description:** The total idle time for all routes in the solution.

## 4.4.18 Total Weight

**Type:** Number. **Description:** The cumulative weight carried in all routes in the solution.

#### 4.4.19 Total Volume

**Type:** Number. **Description:** The cumulative volume carried in all routes in the solution.

# 4.4.20 Assigned Orders

## Type: Number.

**Description:** The number of assigned orders represented as a fraction of the total number of orders.

## 4.4.21 Fleet Usage

**Type:** Number. **Description:** The number of vehicles used in the solution represented as a fraction of the total number of vehicles.

# 4.4.22 Column: Pickup Soft Time Window Early

## Type: Excel time.

**Description:** The total time an order's pickup occurred before the start of its closest soft time window.

## 4.4.23 Column: Pickup Soft Time Window Late

**Type:** Excel time.

**Description:** The total time an order's pickup occurred after the end of its closest soft time window.

## 4.4.24 Column: Delivery Soft Time Window Early

**Type:** Excel time.

**Description:** The total time an order's delivery occurred before the start of its closest soft time window.

## 4.4.25 Column: Delivery Soft Time Window Late

**Type:** Excel time.

**Description:** The total time an order's delivery occurred after the end of its closest soft time window.

# 4.5 Sheet: Unassigned Orders

## 4.5.1 Column: Order

Type: Text.

Description: The identifier of the unassigned order as given by the Id column of the Orders sheet.

## 4.5.2 Column: Pickup Location

Type: Text.

**Description:** The identifier of the pickup location of the unassigned order as given by the Id column of the Locations sheet.

## 4.5.3 Column: Delivery Location

Type: Text.

**Description:** The identifier of the delivery location of the unassigned order as given by the Id column of the Locations sheet.

## 4.5.4 Column: Reason

Type: Text.

**Description:** A reason why the order was not assigned.

**Note:** There may be multiple reason columns. Reasons may not be provided for some unassigned orders; this usually indicates a lack of fleet capacity or insufficient available working hours.

# 4.6 Sheet: Replan Changes

This sheet is only generated when replanning. It contains a summary of differences between the previous solution and this one.

## 4.6.1 Column: Order

Type: Text.

**Description:** The identifier of an order whose vehicle assignment has changed between this solution and the previous one. Orders that are no longer present in the scenario are marked as <CANCELLED>.

#### 4.6.2 Column: Weight

**Type:** Number or Text. **Description:** The weight of the order in this solution. For orders that are no longer present, this is reported as N/A.

## 4.6.3 Column: Volume

Type: Number or Text.

**Description:** The volume of the order in this solution. For orders that are no longer present, this is reported as N/A.

#### 4.6.4 Column: Previous Vehicle

Type: Text.

**Description:** The identifier of the vehicle the order was assigned to in the previous solution. Vehicles that are no longer present in the scenario are marked as **<UNAVAILABLE>**.

#### 4.6.5 Column: New Vehicle

**Type**: Text. **Description**: The identifier of the vehicle the order is assigned to in this solution.

#### 4.6.6 Column: Previous Pickup Time

**Type**: Excel Time. **Description**: The pickup time of the order in the previous solution.

#### 4.6.7 Column: New Pickup Time

**Type**: Excel Time.

**Description:** The pickup time of the order in this solution. For orders that are no longer present, this is reported as N/A.

#### 4.6.8 Column: Previous Delivery Time

**Type**: Excel Time. **Description**: The delivery time of the order in the previous solution.

76

77

# 4.6.9 Column: New Delivery Time

Type: Excel Time.

**Description:** The delivery time of the order in this solution. For orders that are no longer present, this is reported as N/A.

# 5 JSON Input

This section describes the JSON input format to the optimiser.

Any fields that are omitted in the input take default values. Unless noted otherwise, these default values are 0 for numeric fields, false for Boolean fields and null for other types of field (including strings).

# 5.1 Scenario Object

The input to the optimiser is a **scenario** object. It has the following fields:

1. Field: id

Required: No.
Type: String.
Description: An identifier for this scenario assigned by the client. The form of this identifier is client dependent, for example it may be a database id or a UUID.

2. Field: tags

Required: No.Type: Array of strings.Description: Client-specific information associated with the scenario. This field is ignored by the optimiser.

- 3. Field: general Required: Yes.
  Type: A general object.
  Description: Contains global optimiser parameters and settings. Must not be null.
- 4. Field: iteration\_schemes Required: No. Type: Array of iteration batch objects. Description: Defines iteration schemes. The order of the batches in each scheme is defined by the order of this array.
- 5. Field: rush\_hour
  Required: No.
  Type: Array of rush hour objects.
  Description: Contains the rush hour time scaling settings.
- 6. Field: locations Required: Yes. Type: Array of location objects. Description: The set of order pickup and delivery, and vehicle start and finish locations. Must not be null or empty.
- 7. Field: orders
  Required: Yes.
  Type: Array of order objects.
  Description: The set of orders that are to be assigned by the optimiser. Must not be null or empty.
- 8. Field: fleet Required: Yes.

**Type:** Array of **vehicle** objects. **Description:** The set of vehicles making up the fleet. Must not be **null** or empty.

9.	<ul> <li>Field: compartments</li> <li>Required: No.</li> <li>Type: Array of compartment objects.</li> <li>Description: The set of compartments associated with the fleet.</li> </ul>
10.	<ul> <li>Field: fatigue_management</li> <li>Required: No.</li> <li>Type: Array of break objects.</li> <li>Description: The set of break objects that describe how the drivers should take breaks.</li> </ul>
11.	<ul> <li>Field: order_vehicle_compatibility</li> <li>Required: No.</li> <li>Type: Array of attribute pair objects.</li> <li>Description: The rules describing the compatibility between orders and vehicles.</li> </ul>
12.	<ul> <li>Field: order_compartment_compatibility</li> <li>Required: No.</li> <li>Type: Array of attribute pair objects.</li> <li>Description: The rules describing the compatibility between orders and compartments.</li> </ul>
13.	<ul> <li>Field: order_precedences</li> <li>Required: No.</li> <li>Type: Array of order precedence objects.</li> <li>Description: The order precedences for the scenario.</li> </ul>
14.	<ul> <li>Field: order_order_compatibility</li> <li>Required: No.</li> <li>Type: Array of attribute pair objects.</li> <li>Description: The rules describing the compatibility between different orders.</li> </ul>
15.	<ul> <li>Field: load_order_order_compatibility</li> <li>Required: No.</li> <li>Type: Array of attribute pair objects.</li> <li>Description: The rules describing the compatibility between different orders within a load.</li> </ul>
16.	<ul> <li>Field: current_routes</li> <li>Required: No (unless replanning).</li> <li>Type: Array of route objects.</li> <li>Description: If replanning, these are the routes from the previous solution. These are ignored if not replanning.</li> </ul>
17.	<ul> <li>Field: time_matrix</li> <li>Required: No.</li> <li>Type: Array of array of numbers which is a (square) matrix giving the transit times between all pairs of locations in the scenario measured in minutes. The rows and columns are ordered as per the order of the locations field. May be null, in which case the optimiser will compute the transit times itself.</li> </ul>

# 18. Field: distance\_matrix

 $\mathbf{Required:}\ \mathrm{No.}$ 

**Type:** Array of array of numbers which is a (square) matrix giving the transit distances between all pairs of locations in the scenario measured in kilometres. The rows and columns are ordered as per the order of the locations field. May be null, in which case the optimiser

will compute the transit distances itself.

16 June 2025

81

# 5.2 General Object

The **general** object contains settings for optimiser Global Parameters (see Section 2.8). It has the following fields:

- Field: name Required: No. Type: String. Description: A name for the scenario.
- 2. Field: iterations
  Required: No.
  Type: Number.
  Default: 3000.
  Description: The number of iterations that the optimiser should run for.
- 3. Field: iteration\_scheme Required: No. Type: String. Default: null.
  Description: The identifier of the iteration scheme to be used by the optimiser. Note: If an iteration scheme is specified, this it will override the values of the iterations and maximum\_neighbourhood\_size fields if they are also present.
- 4. Field: maximum\_neighbourhood\_size Required: No. Type: Number. Default: 200. Description: The maximum size of a neighbourhood for the optimiser. (See Maximum Neighbourhood Size in Section 2.8.)
- 5. Field: geofence\_north

Required: No.Type: Number in [-90, 90].Description: The latitude of the northern boundary for locations in scenario. Acts to enforce accuracy of geocodes.

6. Field: geofence\_east

Required: No.
Type: Number in [-180, 180].
Description: The longitude of the eastern boundary for locations in scenario. Acts to enforce accuracy of geocodes.

7. Field: geofence\_south Required: No. Type: Number in [-90, 90]. Description: The latitude of the southern boundary for locations in scenario. Acts to enforce accuracy of geocodes.

- Field: geofence\_west Required: No. Type: Number in [-180, 180]. Description: The longitude of the western boundary for locations in scenario. Acts to enforce accuracy of geocodes.
- 9. Field: hard\_geofence\_errors
  Required: No.
  Type: Boolean.
  Default: false.
  Description: If true, then treat any errors produced by geofence violations as being hard.
- 10. Field: colocated\_pickups
  Required: No.
  Type: Boolean.
  Default: false.
  Description: If true, then enforce the colocated pickups constraint.
- 11. Field: batched\_loads
  Required: No.
  Type: Boolean.
  Default: true.
  Description: If true, then enforce the batched loads constraint.
- 12. Field: use\_miles
  Required: No.
  Type: Boolean.
  Default: false.
  Description: If true, then distance values in the solution will be in miles rather than kilometres.
- 13. Field: preferred\_vehicle\_fixed\_reward Required: No. Type: Number (positive). Default: 0. Description: If replanning, a fixed reward for assigning an order to its originally planned vehicle.
- 14. Field: preferred\_vehicle\_weight\_reward Required: No. Type: Number (positive). Default: 0. Description: If replanning, a weight scaled reward for assigning an order to its originally planned vehicle.
- 15. Field: preferred\_vehicle\_volume\_reward Required: No. Type: Number (positive). Default: 0. Description: If replanning, a volume scaled reward for assigning an order to its originally planned vehicle.

- 16. Field: sticky\_deliveries
  Required: No.
  Type: Boolean.
  Default: false.
  Description: If true, then enforce the Sticky Deliveries constraint.
- 17. Field: cost\_scale
  Required: No.
  Type: Number.
  Default: 0 (Which means to not apply cost scaling.)
  Description: Scale costs by this amount before solving. (See Scaling Factors in Section 2.8).
- 18. Field: weight\_scale
  Required: No.
  Type: Number.
  Default: 0 (Which means to not apply weight scaling.)
  Description: Scale weights by this amount before solving. (See Scaling Factors in Section 2.8).
- 19. Field: volume\_scale
  Required: No.
  Type: Number.
  Default: 0 (Which means to not apply volume scaling.)
  Description: Scale volumes by this amount before solving. (See Scaling Factors in Section 2.8).
- 20. Field: arrival\_only\_in\_tw
  Required: No.
  Type: Boolean.
  Default: false.
  Description: Enables the Arrival Only in Time Window behaviour.

# 5.3 Location Object

Each location in the scenario is represented by a **location** object. It has the following fields:

- 1. Field: id Required: Yes. Type: String. **Description:** a unique identifier for the location. 2. Field: name Required: No. Type: String. **Description:** The name (e.g. a store or site name) for the location. 3. Field: tags Required: No. **Type:** Array of strings. **Description:** Client-specific information associated with the location. This field is ignored by the optimiser. 4. Field: attributes Required: No. **Type:** Array of strings. **Description:** An array containing the names of any attributes the location has. 5. Field: address Required: No. Type: String. **Description:** The street address of the location. 6. Field: longitude
- Required: Yes.
  Type: Number.
  Description: The longitude of the location in decimal degrees. Must be in [-180, 180].
- 7. Field: latitude Required: Yes. Type: Number. Description: The latitude of the location in decimal degrees. Must be in [-90, 90].
- 8. Field: site\_time Required: No. Type: String (*HH:MM*). Default: 00:00.
  Description: Site time for the location. (See Site Time in Section 2.10.)
- 9. Field: opening\_time Required: No. Type: String (*HH:MM*). Default: 00:00.
  Description: The opening time of a location. (See Opening Time in Section 2.10.)

- 10. Field: closing\_time
  Required: No.
  Type: String (*HH:MM*).
  Default: null (meaning no closing time).
  Description: The closing time of a location. (See Closing Time in Section 2.10.)
- 11. Field: time\_windows
  Required: No.
  Type: Array of time window objects.
  Default: Empty array (i.e. the location is *always* open).
  Description: An array containing the location's opening hours as time window objects.
- 12. Field: load\_time
  Required: No.
  Type: String (*HH:MM*).
  Default: 00:00.
  Description: Load time for a location. (See Load Time in Section 2.10.)
- 13. Field: unload\_time
  Required: No.
  Type: String (*HH:MM*).
  Default: 00:00.
  Description: Unload time for a location. (See Unload Time in Section 2.10.)
- 14. Field: cost\_per\_visit
  Required: No.
  Type: Number.
  Default: 0.
  Description: The cost incurred when a vehicle visits this location. This may be negative to represent gains associated with visiting the location. (See Cost per Visit in Section 2.10.)
- 15. Field: hidden\_cost\_per\_visit
  Required: No.
  Type: Number.
  Default: 0.
  Description: An additional cost incurred when a vehicle visits this location. This may be negative and is not included in the cost calculation for the solution. (See Hidden Cost per Visit in Section 2.10.)

# 5.4 Order Object

Each order in the scenario is represented by an **order** object, which has the following fields:

1. Field: id

Required: Yes. Type: String. Description: the unique identifier for the order. An order's identifier also acts as an attribute name.

- 2. Field: name Required: No. Type: String.
  Description: A descriptive name for the order.
- Field: tags Required: No. Type: Array of strings.
  Description: Client-specific information associated with the order. This field is ignored by the optimiser.
- 4. Field: attributes
  Required: No.
  Type: Array of strings.
  Description: An array containing the names of any attributes the order has.
- 5. Field: pickup\_location Required: Yes. Type: String.
  Description: The pickup location for the order. Must match the id of a location specified in a location object.
- 6. Field: earliest\_pickup\_time
  Required: No.
  Type: String (*HH:MM*).
  Default: null (meaning no earliest pickup time).
  Description: Earliest time the order can be picked up.
- 7. Field: latest\_pickup\_time Required: No.
  Type: String: (*HH:MM*).
  Default: null (meaning no latest pickup time).
  Description: Latest time the order can be picked up.
- 8. Field: pickup\_time\_windows
  Required: No.
  Type: Array of time window objects.
  Default: null (meaning the pickup can occur at any time).
  Description: Pickup time windows for the order. (See Order Time Windows in section 2.12.)
  An empty array is treated as if the value were null.

- 9. Field: pickup\_soft\_time\_windows
  Required: No.
  Type: A soft time window object.
  Default: null (meaning no soft pickup time windows).
  Description: An object that contains an array of soft time windows for the order's pickup as well as associated penalties.
- 10. Field: pickup\_service\_time
  Required: No.
  Type: String (*HH:MM*).
  Default: 00:00.
  Description: Service time for the pickup. (See Service Time in Section 2.12.)
- 11. Field: delivery\_location
  Required: Yes.
  Type: String.
  Description: The delivery location for the order. Must match the id of a location specified in a location object.
- 12. Field: earliest\_delivery\_time Required: No. Type: String (*HH:MM*). Default: null (meaning no earliest delivery time). Description: Earliest time the delivery can be made.
- 13. Field: latest\_delivery\_time
  Required: No.
  Type: String (*HH:MM*).
  Default: null (meaning no latest delivery time).
  Description: Latest time a delivery can be made.
- 14. Field: delivery\_time\_windows
  Required: No.
  Type: Array of time window objects.
  Default: null (meaning the delivery can occur at any time).
  Description: Delivery time windows for the order. An empty array is treated as if the value were null. (See Order Time Windows in section 2.12.)
- 15. Field: delivery\_soft\_time\_windows
  Required: No.
  Type: A soft time window object.
  Default: null (meaning no soft delivery time windows).
  Description: An object that contains an array of soft time windows for the order's delivery
  as well as associated penalties.
- 16. Field: delivery\_service\_time Required: No. Type: String (HH:MM). Default: 00:00.
  Description: Service time for the delivery. (See Service Time in Section 2.12).

- 17. Field: maximum\_duration
  Required: No.
  Type: String (*HH:MM*).
  Default: null (This means there is no maximum duration).
  Description: The maximum duration of the order.
- 18. Field: weight

Required: No. Type: Number. Default: 0. Description: The weight of the order consistent with vehicle maximum weights. Must be nonnegative.

19. Field: volume

Required: No. Type: Number. Default: 0. Description: The volume of the order consistent with vehicle maximum volumes. Must be nonnegative.

- 20. Field: assign\_cost
  Required: No.
  Type: Number.
  Default: 0.
  Description: The assign cost for the order.
- 21. Field: hidden\_assign\_cost
  Required: No.
  Type: Number.
  Default: 0.
  Description: The hidden assign cost for the order.
- 22. Field: vehicle\_assign\_cost
  Required: No.
  Type: Array of vehicle cost objects.
  Default: No vehicle-specific assign costs.
  Description: The vehicle-specific assign costs for the order.
- 23. Field: vehicle\_hidden\_assign\_cost
  Required: No.
  Type: Array of vehicle cost objects.
  Default: No vehicle-specific hidden assign costs.
  Description: The vehicle-specific hidden assign costs for the order.
- 24. Field: vehicle\_preference
  Required: No.
  Type: Array of vehicle preference objects.
  Default: No vehicle-specific preferences.
  Description: The vehicle-specific rewards for the order.

# 5.5 Vehicle Object

Each vehicle in the fleet is represented by a **vehicle** object which has the following fields:

1. Field: id

Required: Yes. Type: String. Description: A unique identifier for the vehicle. A vehicle's identifier also acts as an attribute name.

- 2. Field: name Required: No. Type: String.
  Description: A descriptive name for the vehicle.
- 3. Field: tags Required: No. Type: Array of strings.
  Description: Client-specific information associated with the vehicle. This field is ignored by the optimiser.
- 4. Field: attributes
  Required: No.
  Type: Array of strings.
  Description: An array containing the names of any attributes the vehicle has.
- 5. Field: compartments Required: No. Type: String.
  Default: null (No compartments).
  Description: The identifier of the compartment scheme used by this vehicle.
- 6. Field: start\_location Required: Yes. Type: String.
  Default: null (This mean the vehicle can start anywhere).
  Description: The start location of the vehicle. This must match the identifier of a location specified in a location object.
- 7. Field: finish\_location Required: No. Type: String.
  Default: null (This means the vehicle can finish anywhere).
  Description: The finish location of the vehicle. This must match the identifier of a location specified in the location object.
- Field: maximum\_weight Required: No. Type: Number. Default: 0. Description: The maximum weight capacity of the vehicle.

91

- 9. Field: maximum\_volume
  Required: No.
  Type: Number.
  Default: 0.
  Description: The maximum volume capacity of the vehicle.
- 10. Field: earliest\_start\_time Required: Yes. Type: String (*HH:MM*).
  Description: Earliest time a vehicle can start a route.
- 11. Field: latest\_start\_time Required: Yes. Type: String (*HH:MM*).
  Description: Latest time a vehicle can start a route.
- 12. Field: latest\_finish\_time Required: Yes. Type: String: (*HH:MM*).
  Description: Latest time a vehicle can finish a route.
- 13. Field: minimum\_paid\_time Required: No.
  Type: String (*HH:MM*).
  Default null.
  Description: A minimum at

**Description**: A minimum amount of time that this vehicle's cost-per-hour will be incurred. If the vehicle's actual working time is less than this, then the hourly cost will be this value multiplied by the cost-per-hour instead. If this value is **null** then the minimum paid time is 00:00.

- 14. Field: maximum\_drive\_time Required: No. Type: String (HH:MM). Default: null. Description: Maximum time a vehicle can spend in transit. If this is null, then the value of maximum\_work\_time is used.
- $15. \ \mathbf{Field:} \ \mathtt{maximum\_work\_time}$

Required: No.
Type: String (*HH:MM*).
Default: null.
Description: The maximum work time for the vehicle; if the value is null then their is no limit on the work time.

16. Field: speed\_scale
Required: No.
Type: Number.
Default: 0 (No speed scale).
Description: Nonnegative speed scale multiplier for the vehicle.

92

17.	<ul> <li>Field: load_time</li> <li>Required: No.</li> <li>Type: String (<i>HH:MM</i>).</li> <li>Default: 00:00.</li> <li>Description: A vehicle specific load time, will override a location load time if applicable. (See Load Time in Section 2.11.)</li> </ul>
18.	<ul> <li>Field: unload_time</li> <li>Required: No.</li> <li>Type: String (<i>HH:MM</i>).</li> <li>Default: 00:00.</li> <li>Description: Unload time for a vehicle. (See Unload Time in Section 2.11.)</li> </ul>
19.	<pre>Field: cost_per_use Required: No. Type: Number. Default: 0. Description: Flat cost to use the vehicle. Must be nonnegative. (See Cost Per Use in Section 2.11.)</pre>
20.	<pre>Field: cost_per_hour Required: No. Type: Number. Default: 0. Description: Cost incurred per hour of vehicle use. Must be nonnegative. (See Cost Per Hour in Section 2.11).</pre>
21.	<ul> <li>Field: cost_per_km</li> <li>Required: No.</li> <li>Type: Number.</li> <li>Default: 0.</li> <li>Description: Cost incurred per kilometre of transit distance for the vehicle. Must be nonnegative. (See Cost Per KM in Section 2.11).</li> <li>Note: If the Use Miles field is true then this must be 0.</li> </ul>
22.	<ul> <li>Field: cost_per_mile</li> <li>Required: No.</li> <li>Type: Number.</li> <li>Default: 0.</li> <li>Description: Cost incurred per mile of transit distance for the vehicle. Must be nonnegative.</li> <li>Note: If the Use Miles field is false then this must be 0.</li> </ul>
23.	<ul> <li>Field: cost_per_load</li> <li>Required: No.</li> <li>Type: Number.</li> <li>Default: 0.</li> <li>Description: Cost incurred for each load. Must be nonnegative. Only applied if batched</li> </ul>

loads is enabled.

- 24. Field: hidden\_cost\_per\_use
  Required: No.
  Type: Number.
  Default: 0.
  Description: The hidden cost for using this vehicle.
- 25. Field: hidden\_cost\_per\_hour
  Required: No.
  Type: Number.
  Default: 0.
  Description: Additional cost incurred per hour

**Description:** Additional cost incurred per hour of vehicle use. This can be negative and is not included in the cost calculation for the solution and could be used to prioritise certain vehicles. (See Hidden Cost Per Hour).

 $26. \ \mathbf{Field:} \ \mathtt{hidden\_cost\_per\_km}$ 

Required: No. Type: Number. Default: 0.

**Description:** Additional cost incurred per kilometre of vehicle use. This can be negative and is not included in the cost calculation for the solution and could be used to prioritise certain vehicles. (See Hidden Cost Per Km).

Note: If the Use Miles field is true then this must be  ${\tt 0}.$ 

27. Field: hidden\_cost\_per\_mile

Required: No.

Type: Number.

Default: 0.

**Description:** Additional cost incurred per mile of vehicle use. This can be negative and is not included in the cost calculation for the solution and could be used to prioritise certain vehicles. (See Hidden Cost Per Mile).

Note: If the Use Miles field is  ${\tt false}$  then this must be 0.

28. Field: hidden\_cost\_per\_load

Required: No.

 $\mathbf{Type:}\ \mathbf{Number}.$ 

Default: 0.

**Description**: Additional cost incurred for each load. This can be negative and is not included in the cost calculation for the solution. Only applied if batched loads is enabled.

- 29. Field: finish\_segment\_distance\_penalty

  Required: No.
  Type: Number.
  Default: 0.
  Description: The per-unit distance component of the finish segment penalty for this vehicle.
  Must be nonnegative.
- 30. Field: finish\_segment\_per\_hour\_penalty
  Required: No.
  Type: Number.
  Default: 0.
  Description: The per hour component of the finish segment penalty for this vehicle. Must be nonnegative.

31. Field: pickup\_segment\_fixed\_penalty

16 June 2025

94

Required: No. Type: Number. Default: 0. **Description:** The fixed component of the pickup segment penalty for this vehicle. Must be nonnegative. 32. Field: pickup\_segment\_distance\_penalty Required: No. Type: Number. Default: 0. Description: The per-unit distance component of the pickup segment penalty for this vehicle. Must be nonnegative. 33. Field: delivery\_segment\_fixed\_penalty Required: No. Type: Number. Default: 0. **Description:** The fixed component of the delivery segment penalty for this vehicle. Must be nonnegative. 34. Field: delivery\_segment\_distance\_penalty Required: No. Type: Number. Default: 0. **Description:** The per-unit distance component of the delivery segment penalty for this vehicle. Must be nonnegative. 35. Field: loaded\_weight\_distance\_penalty Required: No. Type: Number. Default: 0. **Description:** The weight scaled loaded distance penalty factor for this vehicle. Must be nonnegative. 36. Field: loaded\_volume\_distance\_penalty Required: No. Type: Number. Default: 0. **Description:** The volume scaled loaded distance penalty factor for this vehicle. Must be nonnegative. 37. Field: break\_scheme

Required: No. Type: String. Description: An identifier that specifies the break rules that should apply to this vehicle. (See Fatigue Management Scheme in Section 2.11.)

- 38. Field: forbid\_loaded\_breaks
  Required: No.
  Type: Boolean.
  Default: false.
  Description: If set to true, then loaded breaks will not be scheduled for the vehicle.
- 39. Field: maximum\_loads
  Required: No.
  Type: Number (integer).
  Default: 0, which means no maximum number of loads.
  Description: A nonnegative upper bound on the maximum number of loads for this vehicle which is only applied if batched loads is enabled. (See Maximum Loads in Section 2.11.)
  40. Field: max drops per load
- 40. Field: max\_drops\_per\_load Required: No.

**Type:** Number (integer).

 ${\bf Default:}\ 0,$  which means no maximum number of drops.

**Description:** A nonnegative upper bound on the maximum number of locations this vehicle can deliver to in a load. This is only applied if batched loads is enabled. (See Maximum Drops Per Load in Section 2.11.)

# 5.6 Break Object

A **break** object describes a single rule governing how driver should take breaks. A scheme consists of at least one **break** object. (See Section 2.13 for further information). It has the following fields:

- Field: id Required: Yes. Type: String.
   Description: An identifier giving the break scheme to which this rule belongs.
- 2. Field: number
  Required: Yes.
  Type: Number (integer).
  Description: The number of breaks required by the scheme.
- 3. Field: duration Required: Yes. Type: String (*HH:MM*). Description: The duration of the break.
- 4. Field: interval Required: Yes. Type: String: (*HH:MM*).
  Description: The interval of time in which the specified number of breaks must occur.

# 5.7 Compartment Object

An **compartment** object has the following fields:

- Field: scheme\_id Required: Yes. Type: String. Description: The identifier of the compartment scheme to which this compartment belongs.
- 2. Field: id Required: Yes. Type: String.
  Description: The identifier for the compartment. This must be unique within the scheme to which this compartment belongs.
- 3. Field: name Required: No. Type: String.
  Description: A descriptive name for the compartment. This must be unique with the scheme to which this compartment belongs.
- 4. Field: maximum\_weight Required: Yes. Type: Number. Description: The maximum weight capacity of the compartment.
- 5. Field: maximum\_volume
  Required: Yes.
  Type: Number.
  Description: The maximum volume capacity of the compartment.
- 6. Field: attributes
  Required: No.
  Type: Array of strings.
  Description: Any attributes attached to this compartment.

# 5.8 Time Window Object

A time window object has the following fields:

- Field: start Required: Yes. Type: String (*HH:MM*). Description: Starting time of the time window. Must not occur after the end time.
- 2. Field: end Required: Yes. Type: String (*HH:MM*).
  Description: Finishing time of the time window. Must be after start time.
### 5.9 Soft Time Window Object

A soft time window object has the following fields:

- Field: time\_windows
   Required: Yes.
   Type: Array of time window objects.
   Description: An array containing the order's soft delivery times as time window objects.
   An empty array means no soft time windows.
- 2. Field: fixed\_hidden\_cost
  Required: No.
  Type: Number.
  Default: 0.
  Description: The fixed cost for missing a soft time window. Must be nonnegative.
- 3. Field: variable\_hidden\_cost
  Required: No.
  Type: Number.
  Default: 0.
  Description: The variable cost for missing a soft time window. Must be nonnegative.

### 5.10 Attribute Pair Object

An **attribute pair** object is a pair of attributes and it is used for defining compatibility relationships between various entities. (See Section 2.17 for more information). It has the following fields:

Field: attribute1
 Required: Yes.
 Type: String.
 Description: It may be the order attribute in order-vehicle compatibility or order attribute in order-order compatibility.

2. Field: attribute2

Required: Yes.

 $\mathbf{Type:} \ \mathrm{String.}$ 

**Description:** It may be the vehicle attribute in *order-vehicle* compatibility or order attribute in *order-order* compatibility.

### 5.11 Rush Hour Object

This object holds the time dependent speed scaling information. It has the following fields:

- Field: start Required: Yes. Type: String (*HH:MM*). Description: The start of the time period where the scaling factor is applied.
- 2. Field: end Required: Yes. Type: String (*HH:MM*).
  Description: The end of the time period where the scaling factor is applied.
- 3. Field: speed\_scale
  Required: Yes.
  Type: Number.
  Description: The speed scaling factor to be applied.

### 5.12 Iteration Batch Object

An **iteration batch** object defines a single batch within an **iteration scheme**. It has the following fields:

- Field: id Required: Yes. Type: String. Description: The identifier of the iteration scheme to which this batch belongs.
- 2. Field: iterations
  Required: Yes.
  Type: Number (positive integer).
  Description: The number of iterations that the optimiser runs for in this batch.
- 3. Field: maximum\_neighbourhood\_size Required: Yes.
  Type: Number (positive integer).
  Description: The maximum size of a neighbourhood of the optimiser for the duration of this batch.

### 5.13 Vehicle Cost Object

A vehicle cost associates a vehicle with a cost.

- Field: vehicle Required: Yes. Type: String. Description: The id of a vehicle.
- 2. Field: cost Required: No. Type: Number. Default: 0. Description: A cost to associate with the vehicle.

### 5.14 Vehicle Preference Object

A vehicle preference associates a vehicle with rewards.

- Field: vehicle Required: Yes. Type: String. Description: The id of a vehicle.
- 2. Field: fixed\_reward
  Required: No.
  Type: Number.
  Default: 0.
  Description: A fixed reward for assigning the order to the vehicle.
- 3. Field: weight\_reward Required: No. Type: Number. Default: 0.
  Description: A weight reward for assigning the order to the vehicle. This reward is multiplied by the weight of the order.
- 4. Field: volume\_reward Required: No. Type: Number. Default: 0. Description: A volume reward for assigning the order to the vehicle. This reward is multiplied by the volume of the order.

### 5.15 Order Precedence Object

An **order precedence** object defines a **precedence** relationship between two orders. It has the following fields:

- Field: before Required: Yes. Type: String. Description: The id of the first order in the precedence relationship.
- Field: after Required: Yes. Type: String. Description: The id of the second order in the precedence relationship.
- 3. Field: gap

Required: No. Type: String (*HH:MM*).

**Description:** The minimum duration that must between the departure from the delivery stop of the *before* order and the arrival at the pickup stop of the *after* order.

**Default:** null, the arrival at the pickup stop for the *after* order may occur immediately after the departure from the delivery stop for the *before* order.

# 6 JSON Output

This section describes the JSON output format for the optimiser.

### 6.1 Solution Object

Each solution object consists of the following fields:

- Field: routes
   Type: Array of route objects.
   Description: An array of route objects, each of which describes the generated plan for a
   single vehicle.
- Field: route\_kpis
   Type: Array of route\_kpi objects.
   Description: An array of route KPI objects, each of which describes the KPIs for a single vehicle.
- 3. Field: costType: Number.Description: The total cost for all routes in the solution.
- 4. Field: distance
  Type: Number.
  Description: The total transit distance of all routes in the solution.
- 5. Field: delivery\_distance
  Type: Number.
  Description: The transit distance while making deliveries in all routes in the solution.
- 6. Field: loaded\_distance
  Type: Number.
  Description: The total transit distance travelled in all routes while the vehicles are carrying at least one order.
- 7. Field: empty\_distance
  Type: Number.
  Description: The total transit distance travelled in all routes while the vehicles are empty.
- 8. Field: work\_time
  Type: String (*HH::MM*).
  Description: The total work time for all routes in the solution.
- 9. Field: transit\_time
  Type: String (*HH:MM*).
  Description: The total transit time for all routes in the solution.

- 10. Field: service\_time
  Type: String (*HH:MM*).
  Description: The total service time for all routes in the solution. (See Service Time in Section 2.12.)
- 11. Field: site\_time
  Type: String (HH:MM).
  Description: The site time incurred in all routes in the solution. (See Site Time in
  Section 2.10.)
- 12. Field: idle\_timeType: String (*HH:MM*).Description: The total idle time for all routes in the solution.
- 13. Field: break\_timeType: String (*HH:MM*).Description: The total break time for all routes in the solution.
- 14. Field: weight Type: Number.Description: The cumulative weight carried in all routes in the solution.
- 15. Field: volumeType: Number.Description: The cumulative volume carried in all routes in the solution.
- 16. Field: assigned\_orders Type: Number (integer).Description: The number of orders that have been assigned to a vehicle by the optimiser.
- 17. Field: unassigned\_orders
  Type: Array of strings.
  Description: An array containing the order identifiers of any unassigned orders. An empty array means that there are no unassigned orders.
- 18. Field: unassigned\_reasons
  Type: Array of unassigned reason objects.
  Description: The results of the unassigned order analysis for any unassigned orders.
- 19. Field: total\_orders
  Type: Number (integer).
  Description: The total number of orders in the scenario. This is also the length of the orders array in the scenario.
- 20. Field: used\_vehiclesType: Number (integer).Description: The number of vehicles used in the solution.
- 21. Field: total\_vehicles
  Type: Number (integer).
  Description: The total number of vehicles in the scenario.
- 22. Field: loadsType: Number (integer).Description: If batched loads is true then the total number of loads in the solution.

ABN: 13 146 662 053

23. Field: drops\_per\_load

Type: Number.

**Description:** If **batched loads** is **true** then the average number of drops per load in the solution.

24. Field: km\_per\_load

**Type:** Number.

**Description:** If batched loads is true and Use Miles is false then the average number of kilometres per load in the solution.

- 25. Field: miles\_per\_load
  Type: Number.
  Description: If batched loads and Use Miles are both true then the average number of miles per load in the solution.
- 26. Field: drops\_per\_load
  Type: Number.
  Description: If batched loads is true then the average number of drops per load in the solution.
- 27. Field: hours\_per\_loadType: String (*HH:MM*).Description: If batched loads is true then the average time per load.
- 28. Field: pickup\_soft\_time\_window\_early
  Type: String (*HH:MM*).
  Description: If there are pickup soft time windows present in the orders then the total amount of time that orders were picked up early.
- 29. Field: pickup\_soft\_time\_window\_late
  Type: String (*HH:MM*).
  Description: If there are pickup soft time windows present in the orders then the total amount of time that orders were picked up late.
- 30. Field: delivery\_soft\_time\_window\_early
  Type: String (*HH:MM*).
  Description: If there are delivery soft time windows present in the orders then the total amount of time that orders were delivered early.
- 31. Field: delivery\_soft\_time\_window\_late
  Type: String (*HH:MM*).
  Description: If there are delivery soft time windows present in the orders then the total amount of time that orders were delivered late.
- 32. Field: iterationType: Number (integer).Description: The iteration that generated this solution.
- 33. Field: total\_iterations
  Type: Number (integer).
  Description: The total number of iterations that the optimisation that generated this solution was set to run for.

34. Field: errors

**Type:** Array of strings.

**Description:** A list of errors in the solution; will be **null** or empty if there are no errors. **Note:** This field provide feedback when a solution is modified *after* optimisation.

35. Field: warnings

**Type:** Array of strings.

**Description:** A list of warnings about the solution; will be **null** or empty if there are no warnings.

- 36. Field: date\_generated
  Type: String.
  Description: A string giving the day, date, time and year the solution was generated.
  Example: Fri Jul 07 10:12:18 AEST 2017
- 37. Field: scenario

Type: String.

**Description:** The name of the scenario from which the solution was generated. Matches the **name** field of the general object in the scenario object.

#### 38. Field: run\_time

**Type:** String (*HH:MM:SS*). **Description:** The length of time the optimiser ran for.

#### 6.2 Route Object

A route object describes the planned route for a single vehicle. It has the following fields:

1. Field: route\_id

**Type:** Number (integer). **Description:** An number assigned to the route by the optimiser. It is the index of the route in the **routes** field of a solution object.

- 2. Field: vehicle\_id
  Type: String.
  Description: The identifier of the vehicle assigned to this route.
- Field: stops Type: Array of stop objects. Description: See Section 6.3 for further information.

The following fields have been deprecated and users should see Section 6.5 for further information

- 4. Field: cost
  Type: Number.
  Description: The total cost of this route.
- 5. Field: distanceType: Number.Description: The total transit distance during the route.
- 6. Field: loaded\_distance
  Type: Number.
  Description: The total transit distance travelled in the route while the vehicle is carrying at least one order.
- 7. Field: empty\_distance
  Type: Number.
  Description: The total transit distance travelled in the route while the vehicle is empty.
- 8. Field: delivery\_distance
  Type: Number.
  Description: The transit distance while making deliveries as part of the route.
- 9. Field: work\_time
  Type: String (*HH:MM*).
  Description: The total time elapsed from the START to the FINISH stop in the route.
- 10. Field: transit\_time Type: String (HH:MM).Description: The time spent in transit in the route.
- 11. Field: service\_time
  Type: String (*HH:MM*).
  Description: The service time incurred in the route. (See Service Time in Section 2.12.)
- 12. Field: site\_timeType: String (*HH:MM*).Description: The site time incurred in the route. (See Site Time in Section 2.10.)

ABN: 13 146 662 053

- 13. Field: idle\_timeType: String (*HH:MM*).Description: The time spent idle in the route.
- 14. break\_timeType: String (*HH:MM*).Description: The time spent on breaks in the route.
- 15. Field: actual\_start\_timeType: String (*HH:MM*).Description: The time the vehicle started the route.
- 16. Field: actual\_finish\_timeType: String (*HH:MM*).Description: The time the vehicle finished the route.
- 17. Field: weight Type: Number. Description: The cumulative weight carried by the vehicle in the route.
- 18. Field: volume Type: Number.Description: The cumulative volume carried by the vehicle in the route.
- 19. Field: assigned\_orders Type: Number.Description: The number of orders assigned to the vehicle in the route.
- 20. Field: number\_of\_loads
  Type: Number (integer).
  Description: If batched\_loads is true, this is the number of loads in the route. (See Section 2.19).
- 21. Field: loadsType: If batched\_loads is true, this is an array of load\_kpi objects.Description: See Section 6.6 for further information.

### 6.3 Stop Object

Each stop object in the stops field of a route has the following fields:

1. Field: stop\_id

**Type:** Number (integer). **Description:** An number given to the stop by the optimiser. Matches the stop's location in the **stops** array.

- Field: stop\_type
   Type: String (a stop type code).
   Description: The type of stop.
- 3. Field: order

Type: String.

**Description:** For stop\_type PICKUP or DELIVERY the identifier of the order being picked up or delivered. This matches the id of an order. For other types of stop this will be null.

- 4. Field: location Type: String.
  Description: The identifier of the location at which the stop occurs. This matches the id of a location.
- 5. Field: arrival\_time
  Type: String (*HH:MM*).
  Description: The time at which the vehicle is planned to arrive at the stop location.
- 6. Field: departure\_time
  Type: String (*HH:MM*).
  Description: The time at which the vehicle is planned to depart the stop location.
- 7. Field: weight Type: Number.Description: The total weight loaded on the vehicle at the departure from the stop.
- 8. Field: volume
  Type: Number.
  Description: The total volume loaded on the vehicle at the departure from the stop.
- 9. Field: transit\_time
  Type: String (*HH:MM*).
  Description: The time spent in transit between the previous stop and this one.
  Note: If the stop has stop\_type START or BREAK then this will be 00:00.
- 10. Field: transit\_distance

Type: Number. Description: The transit distance between the previous stop and this one. Note: If the stop has stop\_type START or BREAK then this will be 0.

- 11. Field: compartment
  - Type: String.

**Description:** For stop\_type PICKUP or DELIVERY where the compartment scheme (compartments field) for the vehicle is not null this will be the identifier of the compartment that the order has been assigned to. For other types of stop, or where the vehicle lacks a compartment scheme, this will be null.

#### 12. Field: soft\_time\_window\_start Type: String (*HH:MM*).

**Description:** For PICKUP stops where the order has pickup soft time windows, this field is the start time of the pickup soft time window closest to the stop's arrival\_time. For DELIVERY stops where the order has delivery soft time windows, this field is the start time of the delivery soft time window closest to the stop's arrival\_time. For other stop types, for PICKUP stops for orders with no pickup soft time windows or for DELIVERY stops for orders with no delivery soft time windows this field is null.

# 13. Field: soft\_time\_window\_end

**Type:** String (HH:MM).

**Description:** For PICKUP stops where the order has pickup soft time windows, this field is the end time of the pickup soft time window closest to the stop's arrival\_time. For DELIVERY stops where the order has delivery soft time windows, this field is the end time of the delivery soft time window closest to the stop's arrival\_time. For other stop types, for PICKUP stops for orders with no pickup soft time windows or for DELIVERY stops for orders with no delivery soft time windows this field is null.

### 14. Field: soft\_time\_window\_delta

**Type:** String (HH:MM).

**Description:** For PICKUP stops where arrival\_time falls outside of the closest pickup soft time window, this field is either the amount of time from the arrival time to the start of the of the closest pickup soft time window (for early arrivals) or the amount of time from the end of the closest pickup soft time window (for late arrivals). This field is 00:00 if the arrival time is within a pickup soft time window. For DELIVERY stops, the value of this field is similarly defined except that it is with respect to delivery soft time windows. For other stop types, PICKUP stops with no pickup soft time windows, or DELIVERY stops with no delivery soft time windows this field is null.

## 6.4 Stop Type Codes

Stop type codes give the type of a stop. The possible values are:

- 1. START
- 2. PICKUP
- 3. DELIVERY
- $4. \ {\tt BREAK}$
- 5. FINISH

The names of the stop type codes and their meanings are the same as given in Section 2.18.

### 6.5 Route KPI Object

A route object describes the planned route for a single vehicle. It has the following fields:

1. Field: route\_id

**Type:** Number (integer). **Description:** An number assigned to the route by the optimiser. It is the index of the route in the **routes** field of a solution object.

- 2. Field: vehicle\_idType: String.Description: The identifier of the vehicle assigned to this route.
- Field: cost Type: Number. Description: The total cost of this route.
- 4. Field: distanceType: Number.Description: The total transit distance during the route.
- 5. Field: loaded\_distance
   Type: Number.
   Description: The total transit distance travelled in the route while the vehicle is carrying at least one order.
- 6. Field: empty\_distance
  Type: Number.
  Description: The total transit distance travelled in the route while the vehicle is empty.
- 7. Field: delivery\_distance
  Type: Number.
  Description: The transit distance while making deliveries as part of the route.
- 8. Field: work\_time
  Type: String (*HH:MM*).
  Description: The total time elapsed from the START to the FINISH stop in the route.
- 9. Field: transit\_time
  Type: String (*HH:MM*).
  Description: The time spent in transit in the route.
- 10. Field: service\_time
  Type: String (*HH:MM*).
  Description: The service time incurred in the route. (See Service Time in Section 2.12.)
- 11. Field: site\_time
  Type: String (*HH:MM*).
  Description: The site time incurred in the route. (See Site Time in Section 2.10.)
- 12. Field: idle\_timeType: String (*HH:MM*).Description: The time spent idle in the route.

- 13. Field: break\_timeType: String (*HH:MM*).Description: The time spent on breaks in the route.
- 14. Field: actual\_start\_timeType: String (*HH:MM*).Description: The time the vehicle started the route.
- 15. Field: actual\_finish\_timeType: String (*HH:MM*).Description: The time the vehicle finished the route.
- 16. Field: weightType: Number.Description: The cumulative weight carried by the vehicle in the route.
- 17. Field: volume Type: Number. Description: The cumulative volume carried by the vehicle in the route.
- 18. Field: assigned\_orders Type: Number.Description: The number of orders assigned to the vehicle in the route.
- 19. Field: pickup\_soft\_time\_window\_early
  Type: String (HH:MM).
  Description: If there are pickup soft time windows present in the orders then the total amount of time that orders were picked up early in the route.
- 20. Field: pickup\_soft\_time\_window\_late
  Type: String (*HH:MM*).
  Description: If there are pickup soft time windows present in the orders then the total amount of time that orders were picked up late in the route.
- 21. Field: delivery\_soft\_time\_window\_early
  Type: String (*HH:MM*).
  Description: If there are delivery soft time windows present in the orders then the total amount of time that orders were delivered early in the route.
- 22. Field: delivery\_soft\_time\_window\_late
  Type: String (*HH:MM*).
  Description: If there are delivery soft time windows present in the orders then the total amount of time that orders were delivered late in the route.
- 23. Field: number\_of\_loads
  Type: Number (integer).
  Description: If batched\_loads is true, this is the number of loads in the route. (See Section 2.19).
- 24. Field: load\_kpisType: If batched\_loads is true, this is an array of load\_kpi objects.Description: See Section 6.6 for further information.

ABN: 13 146 662 053

### 6.6 Load KPI Object

Field: load\_id

Type: Number (integer).

**Description:** A integer assigned to each distinct load for a route by the optimiser. This is the index of the load in the loads field of a solution object.

- Field: start\_stop\_id
   Type: Number (integer).
   Description: The stop\_id of the stop which starts the load.
- 3. Field: end\_stop\_id
  Type: Number (integer).
  Description: The stop\_id of the stop which ends the load.
- 4. Field: costType: Number.Description: The total cost for the load.
- 5. Field: loaded\_distance
  Type: Number.
  Description: The transit distance from the first pickup to the last of delivery in the load.
- 6. Field: empty\_distance\_before
- Type: Number.

**Description:** For the first load, this is the distance from the **START** stop to the first pickup; otherwise it is the distance from the final delivery of the previous load to the first pickup of this load.

- 7. Field: empty\_distance\_after
  - Type: Number.

**Description:** For the final load, this is the distance from the final delivery to the FINISH stop; otherwise it is the distance from the final delivery of this load to the first pickup of the next load.

- Field: delivery\_distance Type: Number.
   Description: The transit distance while making deliveries as part of the load.
- 9. Field: work\_time
  - **Type:** String (*HH:MM*).

**Description:** The total time from the arrival at the first PICKUP stop to the departure of the last DELIVERY stop in the load.

- 10. Field: transit\_time
  Type: String (*HH:MM*).
  Description: The time spent in transit during this load.
- 11. Field: service\_time
  Type: String (*HH:MM*).
  Description: The service time during this load. (See Service Time in Section 2.12.)
- 12. Field: site\_time
  Type: String (*HH:MM*).
  Description: The site time during his load. (See Site Time in Section 2.10.)

Registered Office: Level 1, 18 Kavanagh St, Southbank, VIC, 3006 117

ABN: 13 146 662 053

- 16 June 2025
- 13. Field: idle\_time **Type:** String (*HH:MM*). **Description:** The time spent idle in for this load. 14. Field: break\_time **Type:** String (*HH:MM*). **Description:** The time spent on break during this load. 15. Field: weight Type: Number. **Description:** The cumulative weight carried by the vehicle in the load. 16. Field: volume Type: Number. **Description:** The cumulative volume carried by the vehicle in the load. 17. Field: weight\_utilization Type: Number. **Description:** A number between [0, 1] that represents the fraction of the vehicle's maximum weight used during the load. 18. Field: volume\_utilization Type: Number. **Description:** A number between [0, 1] that represents the fraction of the vehicle's maximum volume used during the load. 19. Field: number\_of\_deliveries **Type:** Number (integer). **Description:** The number of **DELIVERY** stops in the load. 20. Field: assigned\_orders **Type:** Number (integer). **Description:** The number of orders assigned to this load. 21. Field: pickup\_soft\_time\_window\_early **Type:** String (*HH:MM*). **Description:** If there are pickup soft time windows present in the orders then the total amount of time that orders were picked up early in the load. 22. Field: pickup\_soft\_time\_window\_late **Type:** String (*HH:MM*). **Description:** If there are pickup soft time windows present in the orders then the total amount of time that orders were picked up late in the load. 23. Field: delivery\_soft\_time\_window\_early **Type:** String (*HH:MM*). **Description:** If there are delivery soft time windows present in the orders then the total amount of time that orders were delivered early in the load. 24. Field: delivery\_soft\_time\_window\_late **Type:** String (*HH:MM*). **Description:** If there are delivery soft time windows present in the orders then the total

amount of time that orders were delivered late in the load.

### 6.7 Unassigned Reason Object

- Field: id Type: String. Description: The identifier of the unassigned order.
- Field: reasons Type: Array of strings. Description: Reasons why the order was not assigned.

# Appendices

# A JSON Input Example

This appendix contains an example of the JSON input required for creating a schedule for two vehicles and two orders to allocate. Site time is specified at each location. The vehicles operate using a cost-per-hour rate.

Time windows are specified at all locations (e.g. at the Hallam location, the time windows are 5AM to 10AM and 2PM to 6PM). There are also fatigue management rules in place (all drivers are required to take a single 35 minute break in every 4 hour period of work).

```
{
  "id": "8046e1d7-f713-447a-9fb0-3ce9115ade29",
  "general": {
    "name": "Example Scenario",
    "iterations": 3000,
    "iteration_scheme": "Batched",
    "colocated_pickups": true,
    "batched_loads": true,
    "rush_hour": true
 },
  "locations": [
    {
      "id": "DEPOT",
      "name": "Opturion Office",
      "address": "18 Kavanagh Street St, Southbank, VIC 3006",
      "longitude": 144.966167,
      "latitude": -37.82261,
      "site_time": "0:30",
      "attributes": [],
      "time_windows": [
        {
          "start": "5:00",
          "end": "10:00"
        },
        {
          "start": "11:00",
          "end": "18:00"
        }
      ]
    },
    {
      "id": "Hallam",
      "name": "Hallam ",
      "address": "11 Hallam Crescent",
      "site_time": "00:05",
      "longitude": 145.28085,
      "latitude": -38.018 ,
```

# OPTURION

```
"time_windows": [
      {
        "start": "5:00",
        "end": "10:00"
      },
      {
        "start": "11:00",
        "end": "18:00"
      }
    ]
  },
  {
    "id": "Skye",
    "name": "Skye",
    "address": "11 Skye Road",
    "site_time": "00:10",
    "longitude": 145.19373,
    "latitude": -38.12327,
    "time_windows": [
      {
        "start": "5:00",
        "end": "10:00"
      },
      {
        "start": "11:00",
        "end": "18:00"
      }
    ]
  }
],
"orders": [
  {
    "id": "Order1",
    "pickup_location": "DEPOT",
    "delivery_location": "Hallam",
    "earliest_delivery_time": "10:30",
    "latest_delivery_time": "14:20",
    "delivery_service_time": "0:30",
    "pickup_time_windows":[
      {
        "start": "5:00",
        "end": "10:00"
      },
      {
        "start": "14:00",
        "end": "18:00"
      }
    ],
    "weight": 100,
```

```
"volume": 3
  },
  {
    "id": "Order2",
    "pickup_location": "DEPOT",
    "delivery_location": "Skye",
    "earliest_delivery_time": "10:30",
    "latest_delivery_time": "14:20",
    "delivery_service_time": "0:30",
    "pickup_time_windows":[
      {
        "start": "05:00",
        "end": "10:00"
      },
      {
        "start": "11:00",
        "end": "18:00"
      }
    ],
    "weight": 300,
    "volume": 3
  }
],
"fleet": [
  {
    "id": "Vehicle1",
    "registration": "Opturion-001",
    "driver": "D1",
    "start_location": "DEPOT",
    "finish_location": "DEPOT",
    "maximum_weight": 600,
    "maximum_volume": 10,
    "earliest_start_time": "6:00",
    "latest_start_time": "10:00",
    "latest_finish_time": "18:30",
    "maximum_drive_time": "8:00",
    "maximum_work_time": "9:00",
    "cost_per_use": 150,
    "cost_per_hour": 30.75,
    "cost_per_km": 0.2,
    "attributes": [],
    "break_scheme": "SH"
  },
  {
    "id": "Vehicle2",
    "registration": "Opturion-002",
    "driver": "D2",
    "start_location": "DEPOT",
    "finish_location": "DEPOT",
```

```
"maximum_weight": 600,
      "maximum_volume": 10,
      "earliest_start_time": "6:00",
      "latest_start_time": "10:00",
      "latest_finish_time": "18:30",
      "maximum_drive_time": "8:00",
      "maximum_work_time": "9:00",
      "cost_per_use": 150,
      "cost_per_hour": 30.75,
      "cost_per_km": 0.2,
      "attributes": [],
      "break_scheme": "SH"
    }
  ],
  "rush_hour": [
    {
      "start": "08:00",
      "end": "10:00",
      "speed_scale": 0.8
    }
  ],
  "fatigue_management": [
    {
      "id": "SH",
      "number": 1,
      "duration": "0:35",
      "interval": "4:00"
    }
 ]
}
```

{

# **B** JSON Output Example

The appendix provides the JSON output example for the problem specified in Appendix A.

```
"routes": [
  {
    "route_id": 0,
    "vehicle_id": "Vehicle1",
    "stops": [
      {
        "stop_id": 0,
        "stop_type": "START",
        "location": "DEPOT",
        "arrival_time": "09:55",
        "departure_time": "09:55",
        "transit_time": "00:00",
        "transit_distance": 0.0,
        "load_time": "00:00",
        "unload_time": "00:00",
        "site_time": "00:00",
        "service_time": "00:00"
      },
      {
        "stop_id": 1,
        "stop_type": "PICKUP",
        "order": "Order1",
        "location": "DEPOT"
        "arrival_time": "09:55",
        "departure_time": "10:25",
        "transit_time": "00:00",
        "transit_distance": 0.0,
        "load_time": "00:00",
        "unload_time": "00:00",
        "site_time": "00:30",
        "service_time": "00:00"
      },
      {
        "stop_id": 2,
        "stop_type": "PICKUP",
        "order": "Order2",
        "location": "DEPOT",
        "arrival_time": "10:25",
        "departure_time": "10:25",
        "transit_time": "00:00",
        "transit_distance": 0.0,
        "load_time": "00:00",
        "unload_time": "00:00",
        "site_time": "00:00",
        "service_time": "00:00"
      },
      {
        "stop_id": 3,
        "stop_type": "DELIVERY",
```

],

```
"order": "Order1",
      "location": "Hallam",
      "arrival_time": "11:00"
      "departure_time": "11:35",
      "transit_time": "00:35",
      "transit_distance": 39.74,
      "load_time": "00:00",
      "unload_time": "00:00",
      "site_time": "00:05",
      "service_time": "00:30"
    },
    {
      "stop_id": 4,
      "stop_type": "DELIVERY",
      "order": "Order2",
      "location": "Skye",
      "arrival_time": "11:55",
      "departure_time": "12:35",
      "transit_time": "00:20",
      "transit_distance": 20.93,
      "load_time": "00:00",
      "unload_time": "00:00",
      "site_time": "00:10",
      "service_time": "00:30"
    },
    {
      "stop_id": 5,
      "stop_type": "BREAK",
      "arrival_time": "13:18",
      "departure_time": "13:53",
      "transit_time": "00:00",
      "transit_distance": 0.0,
      "load_time": "00:00",
      "unload_time": "00:00",
      "site_time": "00:00",
      "service_time": "00:00"
    },
    {
      "stop_id": 6,
      "stop_type": "FINISH",
      "location": "DEPOT",
      "arrival_time": "13:55",
      "departure_time": "13:55",
      "transit_time": "00:45",
      "transit_distance": 52.39,
      "load_time": "00:00",
      "unload_time": "00:00",
      "site_time": "00:00",
      "service_time": "00:00"
    }
  ]
}
```

```
"route_kpis": [
 {
    "route_id": 0,
"vehicle_id": "Vehicle1",
    "cost": 277.67,
    "distance": 113.06,
    "delivery_distance": 60.67,
    "loaded_distance": 60.67,
    "empty_distance": 52.39,
    "work_time": "04:00",
    "transit_time": "01:40",
    "service_time": "01:00",
    "site_time": "00:45",
    "load_time": "00:00",
    "unload_time": "00:00",
    "idle_time": "00:00",
    "break_time": "00:35",
    "weight": 400.0,
    "volume": 6.0,
    "average_speed": 67.84,
    "peak_weight": 400.0,
    "peak_volume": 6.0,
    "assigned_orders": 2,
    "number_of_deliveries": 2,
    "number_of_loads": 1,
    "load_kpis": [
      {
        "load_id": 0,
        "start_stop_id": 1,
        "end_stop_id": 4,
        "cost": 244.13,
        "distance": 60.67,
        "delivery_distance": 60.67,
        "maximum_delivery_distance": 39.74,
        "loaded_distance": 60.67,
        "empty_distance_before": 0.0,
        "empty_distance_after": 52.39,
        "work_time": "02:40",
        "transit_time": "00:55",
        "service_time": "01:00",
        "site_time": "00:35",
        "idle_time": "00:10"
        "break_time": "00:00",
        "load_time": "00:00",
        "unload_time": "00:00",
        "weight": 400.0,
        "volume": 6.0,
        "average_speed": 66.19,
        "weight_utilization": 0.67,
        "volume_utilization": 0.6,
        "number_of_deliveries": 2
      }
   ]
 }
```

OPTURION

}

16 June 2025

], "cost": 277.67, "distance": 113.06, "delivery\_distance": 60.67, "maximum\_delivery\_distance": 60.67, "loaded\_distance": 60.67, "empty\_distance": 52.39, "work\_time": "04:00", "transit\_time": "01:40", "idle\_time": "00:00", "break\_time": "00:35", "weight": 400.0, "volume": 6.0, "average\_speed": 67.84, "assigned\_orders": 2, "unassigned\_orders": [], "total\_orders": 2, "used\_vehicles": 1, "total\_vehicles": 2, "loads": 1, "km\_per\_load": 113.06, "miles\_per\_load": 0.0, "hours\_per\_load": "04:00", "drops\_per\_load": 2.0, "scenario": "dtoExampleScenario", "objective": 295.0, "iteration": 3000, "total\_iterations": 3000, "date\_generated": "Thu Jul 19 15:29:34 AEST 2018", "run\_time": "00:00"

# C Unsupported Excel Functions

The following table lists all of the Excel functions that are *not* supported by the optimiser.

ABSREF	DBCS	GET.CELL	LOGEST	RSQ
APP.TITLE	DDB	GET.DOCUMENT	LOGINV	SAVE.TOOLBAR
ARGUMENT	ENABLE.TOOL	GET.WINDOW	LOGNORMDIST	SEARCHB
ASC	END.IF	GET.WORKBOOK	MIDB	SKEW
BETADIST	ERROR	GET.WORKSPACE	N	SLN
BETAINV	EVALUATE	GETPIVOTDATA	NEGBINOMDIST	STEP
BINOMDIST	EXEC	GOTO	NUMBERSTRING	STEYX
CALL	EXPONDIST	GROWTH	PERMUT	SYD
CELL	FDIST	HARMEAN	PHONETIC	TINV
CHIDIST	FINDB	HYPGEOMDIST	PRESS.TOOL	TRIMMEAN
CHIINV	FINV	INFO	PROB	TTEST
CHITEST	FISHER	ISPMT	QUARTILE	TYPE
CONFIDENCE	FISHERINV	KURT	REGISTER.ID	USDOLLAR
CRITBINOM	FTEST	LAST.ERROR	RELREF	VDB
DATEDIF	GAMMADIST	LEFTB	REPLACEB	WEIBULL
DATESTRING	GAMMAINV	LENB	RETURN	WINDOW.TITLE
DB	GAMMALN	LINEST	RIGHTB	ZTEST