

TransXChange

Examples 2.4

The following development [examples](#) demonstrate the use of TransXChange to encode simple and complex bus schedules. For each example, both the XML document in TransXChange format, and the PDF file output from the [TransXChange Publisher](#) are provided.

See also the [TransXChange Samples](#) which provide further examples of using TransXChange from live systems.

Group	Name	Description & Features	XML	Output
Basic	Linear (See section below) Registration Schema	A single straight route run by a single operator. All vehicle journeys have the same timings. <ul style="list-style-type: none">• Linear route.• Registration details.• Route Track Map.• Local Track data, including instructions and Mapping System Reference.• Frequent Service, with Frequency based journey times, specified as an interval.• Inbound and outbound service.• Operator Details including Parent Operator (TXC v2.4) & Licence details.• Reversing manoeuvre.	XML	Matrix PDF Map PDF
	Express (See section below) General Schema	A linear route with express journey patterns running over it that omit stops. <ul style="list-style-type: none">• Express service.• Reuse of Vehicle journey timing link in multiple journeys.• Overriding of Journey Pattern Timing Link Run	XML	Matrix PDF Map PDF

		<p>Times with different values on the Vehicle Journey Timing links for some journey.</p> <ul style="list-style-type: none"> • Use of Wait Time. • Holiday Day Type Exclusion. • Local stop point definitions for an off-street Bus Station: BCQ, BCE and BCS stop type, including SMS codes for TXC V2.4 rules, e.g. London. • Journey Footnote. • Local stop area definition. • Variable Bay Allocation. • Supporting document. • Marketing Name (TXC v2.4). 		
	<p>Cancellation (See section below) Registration Schema</p>	<p>Minimal details needed to identify a registration for cancellation. (TXC C v2.4)</p> <p>Cancellation of a registration.</p>	XML	Matrix PDF
Complex	<p>Interchange (See section below) Registration Schema</p>	<p>Two routes run by two different operators. All vehicle journeys have the same timings.</p> <ul style="list-style-type: none"> • Inbound and Outbound timetable. • An Interchange. • Linear route, with different stop visiting pattern at one end. • Express stop. • Frequent Service journey times, specified as an interval, but not a Frequent Service. • Combining operating days from service, journey pattern and vehicle journey level. 	XML	Matrix PDF

		<ul style="list-style-type: none"> • Serviced Organisation & School dates, including Classification (TXC v2.4). • More than one operation. • Use of Stop Sequence Numbers. • Timetable Note. • Line Colours (TXC v2.4). 		
	Circular (See section below) General Schema	A circular route. <ul style="list-style-type: none"> • Circular route. • Reuse of route section. • Dead runs, Positioning links (Including TXC v2.4 duty crew). • Late night services that cross midnight till next day. • Different weekday and Weekend variants. • Journey Times past midnight for a given day type (TXC v2.4). • Partial traversal of Journey pattern. • Operator Garage. • AVL data - Vehicle Type Ticket Machine, Duty crew. • Multiple Variants of data per journey (TXC v2.4). • Reusable Day Types (TXC v2.4). • Vehicle Type Equipment for Accessibility (TXC v2.4) • Weekend service. • WGS8. • Service Level Vias. • Running Board / Dynamic Destinations. 	XML	Matrix PDF
	Cloverleaf (See section below) General Schema	A cloverleaf route shape with three petals. <ul style="list-style-type: none"> • Multiple routes composed of common route section. 	XML	Matrix PDF

		<ul style="list-style-type: none"> Multiple journey patterns composed of common journey pattern section. Dynamic destination display. Recommended End date on Period (v2.4). 		
	Lollipop (See section below) General Schema	Lollipop shaped route, with two parallel branches. <ul style="list-style-type: none"> Circular and parallel section. Reuse of journey pattern section. Reuse of VehicleJourney Links. Stop Sequence Number to control row order. Timing status on stop usage other than 'Principle Timing Point' (<i>PTP</i>). Complex day types for regular and bank holiday operation. Use of TXC publisher option. Layover Point. Service Classification combinations including for '&' in code (TXC v2,4) Two services run by two different operator. Connecting service. Express stopping pattern for some journeys. Use of Stop Sequence Number. Use of Dynamic Destination headings. Partial Frequency Based service (TXC v2.4). 	XML	Matrix PDF
	Eye (See section	An eye shaped route, with two alternative branches.	XML	PDF en

	below) Registration Schema	<ul style="list-style-type: none"> Multiple routes composed of common route section. Multiple journey patterns composed of common journey pattern section. Stop Sequence Number to control row order. Local stop point definition. New Stops required. Bilingual stop names & schedule (Cymraeg). Dynamic Destinations & Vias (Running Board) (TXC v2.4). Multiple classifications (including TXC v2.4 relaxation of combinations). 	en XML cy	PDF cy Map PDF
	Flexible (see section below) Registration Schema	Use of flexible zones <ul style="list-style-type: none"> Flexible zone. Flexible time band. 	XML	Matrix PDF
	Grouping (see section below) Registration Schema	Use of Journey Grouping (TXC v2.4). <ul style="list-style-type: none"> Built in Journey Bed labels. (TXC v2.4). Custom Journey Beds. (TXC v2.4). 	XML	Matrix PDF Map PDF
	Hail & Ride (See section below) Registration Schema	Use of hail and ride stops. <ul style="list-style-type: none"> Hail and ride section. Local stop point definition. Full lollipop topology. Frequency based journey times, specified as minutes past the hour, but not a frequent service. Short notice registration detail. Scottish Bank Holidays including St Andrews Day 	XML	Matrix PDF Map PDF

		& January 2 Displacement (TXC v2.4). <ul style="list-style-type: none"> Workflow Attributes (TXC v2.4). 		
	Large Route (See section below) Registration Schema	Very large timetable. <ul style="list-style-type: none"> More stops than fit down a page. More journeys than fit across a page. Basing of vehicle journeys on other vehicle journey. Timing links with zero duration. Multiple Frequency based services, specified as intervals, but not a frequent service. 	XML	Matrix PDF Map PDF
	Merge Frequent Journeys (See section below) Registration Schema	Individually coded frequent services that are to be merged as a single column. <ul style="list-style-type: none"> Frequent service with, frequent journeys coded individually. Merging by publisher. Page overflow. Non PTP point. Default Operating Profile. 	XML	Matrix PDF Map PDF
	Footnotes (See section below) Registration Schema	Service with complex conditions requiring footnotes frequent services that are to be merged as a single column. <ul style="list-style-type: none"> Large number of services (144). Frequent journeys. Footnote. Page overflow. Garage detail. Short Notice Registration. Stop Sequence Numbers to 	XML	Matrix PDF Map PDF

		control row order. <ul style="list-style-type: none"> Operational data: Block, Vehicle type, layover point etc. 		
Other Data Exchange	Operators (See section below) General Schema	Exchange of just operators. <ul style="list-style-type: none"> Multiple operators, no timetable 	XML	
	Routes (See section below) General Schema	Exchange of just routes.	XML	
	Delta (See section below) General Schema	Exchange of just changes to a vehicle journey.	XML	

Examples 2.4 - Cancellation

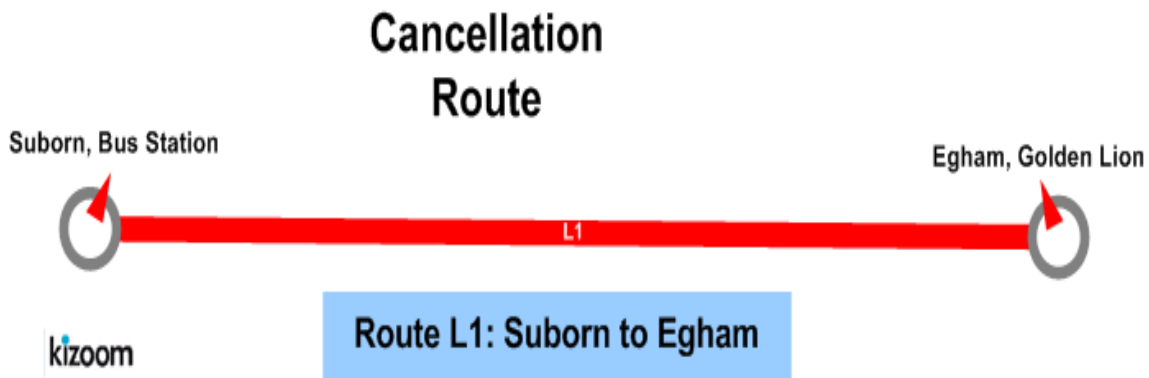
Summary

Example of a cancellation of a previously submitted route (see **Linear section**). Only the details necessary to identify the route are supplied

- Concise Cancellation route.

TransXChange XML Document	Particulars and Matrix output PDF	Route Map Output
cancellation.xml	Particulars and Matrix PDF	Cannot be published

Route Map



Timetable

Not available (see **Linear example**).

The XML Representation

Service Registration

- The **Registration** is for a single **Operator**

Service Structure

- There is a single **Service** instance *SVI* , with one **Line** - '*LI*' .

The Operating Days

- The service **OperatingProfile** says it runs Monday to Friday, every day of the year.

Notes

This example cannot be published as a full timetable. It is also possible to include the full timetable in which case it can be published.

Page last updated: 2013/04/13

Examples 2.4 - Circular route

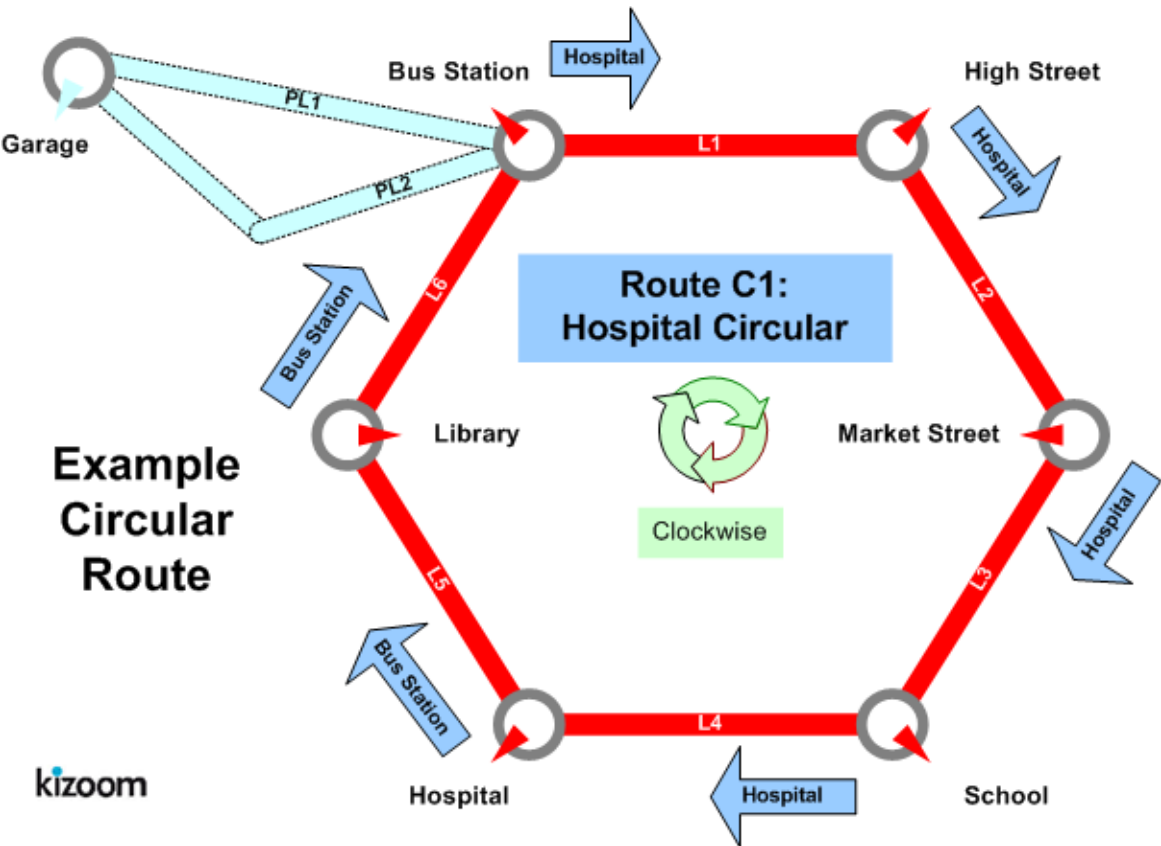
Summary

A circular route served in a clockwise direction by a single operator. All vehicle journeys have the same timings.

- Circular route.
- Reuse of route sections.
- Multiple Line.
- Operational Profile (see below) for different Weekend service, with fewer journeys and different evening journey times. Overriding of a default profile.
- Journeys that run past end of day/midnight boundary.
- Journeys that start after midnight on Saturday but should be shifted to be shown as starting on the Monday to Friday service i.e. be shifted to a different bed and marked with a note that they are in a different day. (TXC v2.4)
- Partial Traversal of Route / Journey Pattern.
- Dead runs, Positioning links (Including TXC v2.4 duty crew).
- Operator Garage.
- Service level Vias
- Dynamic Destination Displays.
- Fare Stages (see below).
- Operational data: Vehicle Type, Ticket Machine, Duty crew (see below).
- Operational data Variants for different days (TXC v2.4).
- Reusable Day Types (TXC v2.4).
- Vehicle Type Equipment for Accessibility (TXC v2.4)
- WGS8 coordinates
- General Schema.

Published as: [PDF](#) [PDF Timetable only](#)

Route Map



Journeys

Bus Station - High Street - Hospital - Bus Station.

		Based on Ref to	JP1	JP1	JP1	VJ2	VJ2	JP1	JP1	JP1	JP1	JP1	JP1
		Links based on	->JP1	JP1 + start dead run	->JP1	->VJ2	->VJ2	P1 + end dead run	->JP1	->JP1	->JP1	->JP1	->JP1
		Operational Profile	Override	->JP1	Override	->VJ2	Override	->JP1	Override	Override	Override	Override	Override
		VJ	#A	#1	#2	#3	#4	#5	#6	#7	#8	#B	#C
		Line	1 Night	1	1	1	1 ++	1	1 Night	1 Night	1 Night	1 Night	1 Night
Days	Service	MTWTFSS	MTWTFSS	MTWTFSS	--	MTWTFSS	MTWTFSS	MTWTFSS	MTWTFSS	MTWTFSS	MTWTFSS	MTWTFSS	MTWTFSS
Of Week	Vehicle Journey	MTWTF--	"	MTWTF--	(MTWTF--)	c	"	MTWTF--	MTWTF--	-----SS	-TWTFSS-	-----F-	
	DayShift	-1	0	0	0	0	0	0	0	0	0	+1	+1
	Actual Days	MTWT--S	MTWTFSS	MTWTF--	MTWTF--	MTWTFSS	MTWTF--	MTWTF--	MTWTF--	-----SS	-TWTFSS-	-----S-	

		Beds	MF	MF, SS	MF	MF	MF, SS	MF, SS	MF	MF	SS	MF	MF
		Dep	(23:55)	(10:29)	(11:00)	(12:00)	(13:00)	(14:00)	(22:30)	(23:30)	(23:33)	(00:00)	(00:10)
xL1		Bus Station	23:55**	-	11:00	12:00	13:00	14:00	22:30	23:30	23:33	00:10*	00:20*
	xL2	High St	23:58**	-	11:03	12:03	13:03	14:03	22:33	23:33	23:37	00:13*	00:23*
xL3		Market St	00:02	-	11:07	12:07	13:07	14:07	22:37	23:37	23:50	00:17*	00:27*
	xL4	School	00:15	-	11:20	12:20	13:20	14:20	22:50	23:50	23:59	00:30*	00:40*
xL5		Hospital	00:24	10:29	11:29	12:29	13:29	14:29	22:59	23:59	00:18*	00:39*	00:49*
	xL6	Library	00:43	10:48	11:48	12:48	13:48	-	23:18	00:18*	00:30*	00:58*	01:08*
		Bus Station	00:55	11:00	12:00	13:00	14:00	-	23:30	00:30*	00:42:*	01:10*	01:20*
			Sunday to Friday	Everyday	Monday to Friday	Monday to Friday	Not Thursday	Everyday	Monday to Friday	Monday to Friday	Saturday & Sunday Only	Tuesday to Saturday	Fridays Only

* Next Day

Timetable - Clockwise, Monday to Friday

Bus Station - High Street - Hospital - Bus Station.

		VJ	#A	#1	#2	#3	#4	#5	#6	#7	#B	#C
		Line	1 Night	1	1	1	1 ++	1	1 Night	1 Night	1 Night	1 Night
Days		Actual Days	MTWT--S	MTWTFSS	MTWTF--	MTWTF--	MTW-FSS	MTWTFSS	MTWTF--	MTWTF--	-TWTFs-	-----S-
Of Week		Days of week	MTWTF--	MTWTFSS	MTWTF--	MTWTF--	MTW-FSS	MTWTFSS	MTWTF--	MTWTF--	MTWTF--	-----F-
		DayShift	-I	0	0	0	0	0	0	0	+I	+I
		Dep	(23:55))	(10:29)	(11:00)	(12:00)	(13:00)	(14:00)	(22:30)	(23:30)	(00:00)	(00:10)
xL1		Bus Station	23:55**	-	11:00	12:00	13:00	14:00	22:30	23:30	00:10*	00:20*
	xL2	High St	23:58**	-	11:03	12:03	13:03	14:03	22:33	23:33	00:13*	00:23*
xL3		Market St	00:02	-	11:07	12:07	13:07	14:07	22:37	23:37	00:17*	00:27*
	xL4	School	00:15	-	11:20	12:20	13:20	14:20	22:50	23:50	00:30*	00:40*
xL5		Hospital	00:24	10:29	11:29	12:29	13:29	14:29	22:59	23:59	00:39*	00:49*
	xL6	Library	00:43	10:48	11:48	12:48	13:48	-	23:18	00:18*	00:58*	01:08*
		Bus Station	00:55	11:00	12:00	13:00	14:00	-	23:30	00:30*	01:10*	01:20*
			Sunday to Friday				Not Thursday				Tuesday to Saturday	Fridays Only

Footnotes

Service runs Monday to Friday

** Previous Day

* Next Day

++ Not Thursday

Timetable Clockwise, Saturday & Sunday

Bus Station - High Street - Hospital - Bus Station

		VJ	#1	#4	#5	#8
		Line	1	1	1	1 Night
Days Of Week		Actual Days	MTWTFSS	MTW-FSS	MTWTFSS	-----SS
		Days of Week	MTWTFSS	MTW-FSS	MTWTFSS	-----SS
		DayShift	0	0	0	0
xL1		Dep	(10:29)	(13:00)	(14:00)	(23:30)
	xL2	High St	-	13:03	14:03	23:33
xL3		Market St	-	13:07	14:07	23:37
	xL4	School	-	13:20	14:20	23:50
xL5		Hospital	10:29	13:29	14:29	23:59
	xL6	Library	10:48	13:48	-	00:18*
		Bus Station	11:00	14:00	-	00:30*
				Not Thursday		* next day

The XML Representation

[XML Document](#)

Service Registration

- The service is not registered.
- There is a single **Operator**.

Service Structure

- There is a single **Service** instance **SVI**, with two **Lines** - '1' and '1 Night'.
- There is a single **RouteSection**, **RS1**, connecting the six stops. It has six **RouteLinks** between the six stops;
 - (RL1) Bus Station to High Street,
 - (RL2) High Street to School,
 - (RL3) School to Market Street,
 - (RL4) Market Street to Hospital,
 - (RL5) Hospital to Library, and

- (RL6) Library to Bus Station.
- There is a single **Route RI**, with the one **RouteSection, RS1**.
- There is a single **JourneyPattern, JP1**, with a single **JourneyPatternSection** instance **JS1**; corresponding to the route section **RS1**, and containing six **JourneyPatternTimingLink** instances, **JPTL1 - JPTL6**, which project onto the respective **RouteLink** instances **RL1-RL6**.
 - **JPTL1** has a **DynamicDestination** heading of *Hospital*. This will be in effect until the next heading is encountered.
 - The **JPTL4-To** usage has a **DynamicDestination** heading of *BusStation*: from this point a different destination will be shown.
- There are ten **VehicleJourney** instances, with different day profile conditions, so different journeys get selected for Monday to Friday and for Saturday and Sunday:
 1. **VJ1** has an initial **StartDeadRun** run to position the bus at the *Hospital* stop from the **Operator's Garage**. It has just two vehicle journey timing links, **VJTL5 & VJTL6**, corresponding to the last two sequential links of the journey pattern, **JPTL5 & JPTL6**, to visit the last three stops of the route.
 - It inherits an **OperationalProfile** from **JP1** that says it runs **MondayToFriday**.
 2. **VJ2** has six vehicle journey timing links, corresponding to all six journey pattern links, starting at *Bus Station* at 11:00.
 - It has an override **OperationalProfile** that says it runs **MondayToFriday**
 3. **VJ3, VJ4** reference **VJ2** for all their timing links, specifying only a different start time. (12:00 and 13:00).
 - **VJ3** inherits an **OperationalProfile** from **VJ2** that says it runs **MondayToFriday**
 - **VJ4** has an override **OperationalProfile** that says it does not run on **Thursday**, but otherwise runs Monday to Sunday.
 4. **VJ5** has just three vehicle journey timing links to complete the journey, corresponding to the first three journey pattern timing links, and a final **EndDeadRun** to return the bus to the *Garage*. It has an override **RunningBoard** showing the short running
 5. **VJ6** follows all six links of **JP1** starting at *Bus Station* at 22:30.
 - It has an **OperationalProfile** that says it runs **MondayToFriday**
 6. **VJ7** follows all six links of **JP1**, starting at *Bus Station* at 23:30, so its later stages will take place in the next calendar day.
 - It has an **OperationalProfile** that says it runs **MondayToFriday**
 7. **VJ8** follows all six links of **JP1** starting at *Bus Station* at 22:30.
 - It has an **OperationalProfile** that says it runs **Saturday and Sunday only**.
- Three of the **VehicleJourney** instances, are day shifted in a different bed, so that a journey that actually takes place on saturday morning appears as part of the Monday to Friday service (with a footnote) and a journey that actually takes place late on sunday night is grouped as a Monday to Friday service.

0. **VJ_A** follows all six links of **JP1** for all its timing links starting at **Bus Station** at **00:10**, .
 - It has an **OperationalProfile** that says it runs **Tuesday, Wednesday, Thursday, Friday and Saturday**.
 - A negative **Day shift** causes it to be placed in the **MondayToFriday** bed as the first journey of the day.
1. **VJ_B** follows all six links of **JP1** for all its timing links starting at **Bus Station** at **00:20**, .
 - It has an **OperationalProfile** that says it runs only on **Saturday**.
 - A negative **Day shift** causes it to be placed in the **MondayToFriday** bed at the beginning of the day.
2. **VJ_C** follows all six links of **JP1** for all its timing links starting at **Bus Station** at **23:55**, .
 - It has an **OperationalProfile** that says it runs **Sunday, Monday, Tuesday, Wednesday, , and Thursday**.
 - A negative **Day shift** causes it to be placed in the **MondayToFriday** bed as the first journey of the day.

Operational Profile

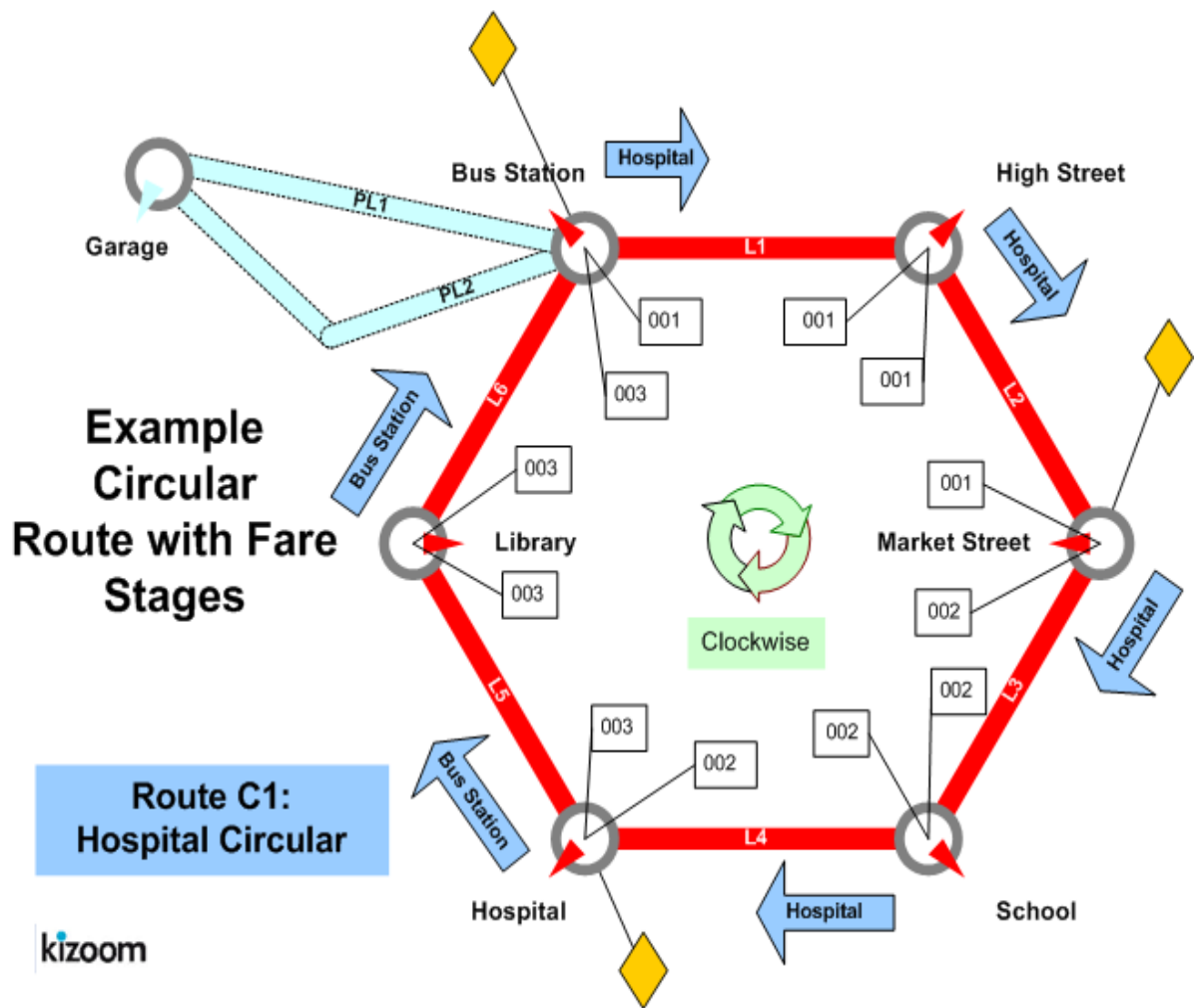
- On the **JourneyPattern**, **OperationalProfile / RegularDayType/ DaysOfWeek / MondayToSunday** states that by default that the service runs everyday of the week.
- Some **VehicleJourney** instances, override this as described above

Operational Details

- The **Operator** has a **Garaged** defined.
- A default **Service / TicketMachineServiceCode** is specified.
- On the **JourneyPattern**, **Operational** details are specified:
 - A default **TicketMachine / JourneyCode** is specified. The **TicketMachine / Direction** is different from that of the service.
 - A **VehicleType** is specified.
 - A default **RunningBoard** is specified.
- **DutyCrew CRW1** runs the first four journeys. A second **DutyCrew CRW2** takes over at '**Bus Station**' to run the final journey #5.
- **There are different variants for different days of the week.**

Fare Stages

- There are **FareStage** instances at '**Bus Station**', '**Market Street**', and '**Hospital**'.



Examples 2.4 - Cloverleaf route

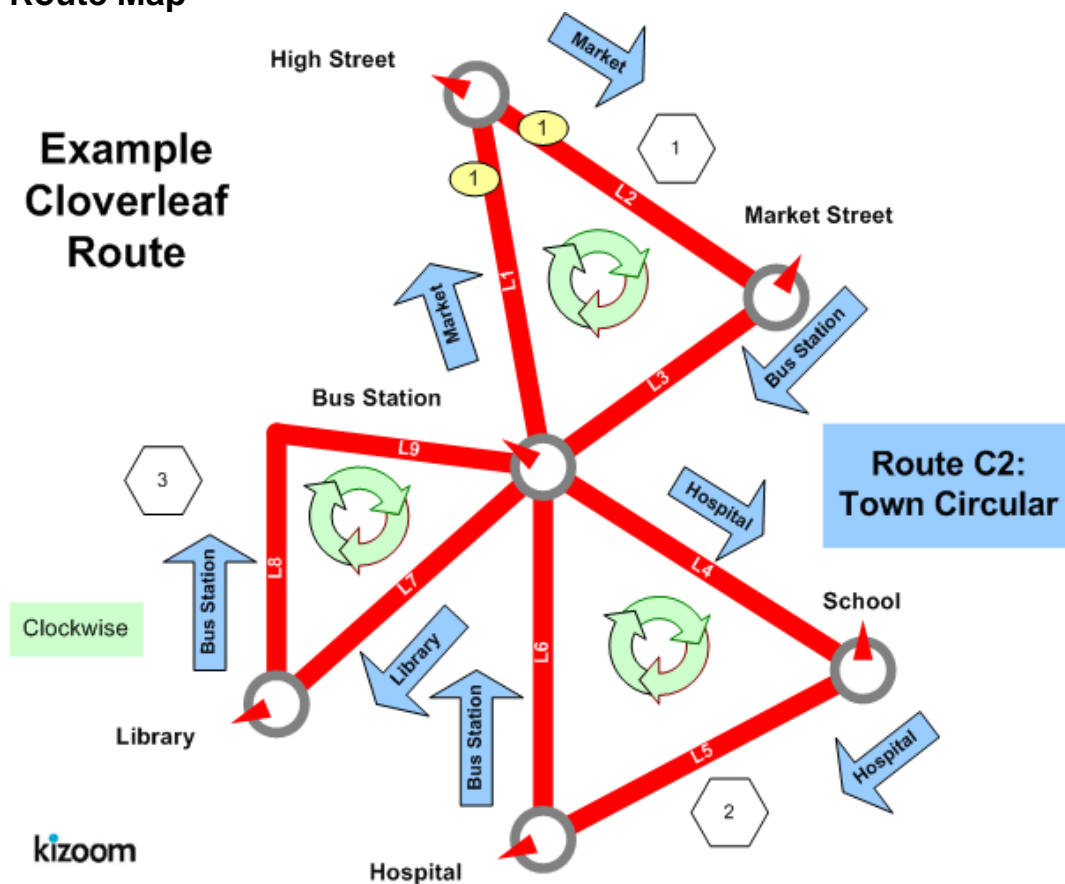
Summary

The bus traces out the outline of a cloverleaf shape with three petals. All vehicle journeys have the same timings.

- Multiple routes composed of shared route sections.
- Multiple journey patterns composed of shared journey pattern sections.
- Visiting the same stop more than once within the same route.
- Frequent services with different Frequency Phrases (see below).
- Stop Sequence Numbers (see below) to control the timetable presentation matrix.
- Dynamic destination displays (see below).
- Short Working.
- Recommended Period end date for service (TXC 2.4).
- Commercial basis on some links (TXC 2.4)

Published as: [PDF](#)

Route Map



Timetable

Bus Station to Bus Station												
			#1	#2	#3	#4	#5	#6	#7	#8	#9	#10
			2	2		2		2		2		2
Petal 1		Bus Station	-	11:00		12:00		13:00		14:00		15:00
		High St	-	11:03		12:03		13:03		14:03		15:03
		Market St	-	11:07		12:07		13:07		14:07		15:07
Petal 2		Bus Station	-	11:20	<i>then about every 5 minutes until</i>	12:20	<i>then at 3-7 minutes intervals until</i>	13:20	<i>then at intervals of no more than 7 mins until</i>	14:20	<i>then at 3-8 minutes intervals until</i>	15:20
		School	-	11:29		12:29		-		-		-
		Hospital	-	11:48		12:48		-		-		-
Petal 3		Bus Station	11:00	12:00		13:00		13:20		14:20		-
		Library	11:19	12:19		13:19		13:39		14:39		-
		Bus Station	11:31	12:31		13:31		13:51		14:51		-

The XML Representation

[XML Document](#)

Service Registration

- The service is not registered
- There is a single **Operator**.

Service Structure

- There is a single **Service** instance **SV1**, whose routes are all labelled as one **Line** - '2'.
- There are three **RouteSection** instances, one for each leaf of the clover:
 - **RS1**, has three links between (**L1**) *Bus Station* to *High Street*, (**L2**) *High Street* to *Market Street*, (**L4**) *Market Street* to *Bus Station*.

- **RS2**, has three links between (**L1**) *Bus Station to School*, (**L3**) *School to Hospital*, (**L5**) *Hospital to Bus Station*.
 - **RS3**, has two links between (**L1**) *Bus Station to Library*, and (**L6**) *Library to Bus Station*.
- There are five **Route** instances defined:
 - **R1, R2, R3**, each with a single section (**RS1, RS2, RS3** respectively) describing a single petal. (**R2** is not actually used)
 - **R4** visits the first and third petals only (**RS1 & RS3**).
 - **R5** describes a route round all three petals, reusing the three sections (**RS1, RS2, RS3**) in succession.
- There are three **JourneyPatternSection** instances **JS1, JS2, and JS3**, corresponding to the three **RouteSection** instances, with corresponding journey pattern timing links.
- There are five **JourneyPatterns**, **JP1 - JP5**, composed from the **JourneyPatternSection** instances in the same way the routes are composed from the route sections.
- There are six **VehicleJourney** instances, one for each column:
 - **VJ1**, starting at 11:00 (column #1), references **JP3** for its links, to run round the third petal.
 - **VJ2**, starting at 11:00 (column #2), references **JP5** for its links and so has eight **VehicleJourneyTimingLinks**, corresponding to all eight **JourneyPatternLinks**.
 - The **Frequency** element specifies that is a **Frequent** with a statutory **ScheduledFrequency** of every 5 minutes, and an **EndTime** of 12:00, **FrequentService** is *true*. The publisher generates both column #1 and column #2.
 - **VJ3**, starting at 12:00 (column #3), references **JP5** for its links, specifying only a different start time.
 - **VJ4**, starting at 13:00 (column #4), references **JP4** for its links, to run over the first and third petals
 - **VJ5**, starting at 13:00 (column #4), references **VJ4** for its links, to also run over the first and third petals
 - **VJ6**, starting at 11:00 (column #5), references **JP1** for its links, running over just the first petal

Operational Data

- The **DynamicDestinationDisplay** is specified to change on certain links of the Journey pattern so that the heading changes between the outward and inward link, for instance for petal 1 at *Bus Station* and *High Street* it shows '*Market Street*', but at *Market Street*, it shows '*Bus Station*'

Use of Stop Sequence Numbers

Stop sequence numbers are used to coerce a specific ordering of the stops within a matrix timetable when published.

Frequency Phrases

The phrase used in frequency group column in the published output will be derived from the Frequency Parameters as follows.

	Frequency					
Case	Frequent Service	Scheduled-Frequency Interval (mins)	Minimum Frequency Interval (mins)	(mins)	Result Phrase to show in matrix column for NON-REGISTRATION details	Result Phrase to show in matrix column for REGISTRATIONS
VJ2	true	5	-	-	<i>then about every 5 minutes until</i>	<i>Frequent service at least every 10 mins until</i>
VJ3	true	7	4	-	<i>then at 3-7 minutes intervals until</i>	<i>Frequent service at least every 10 mins until</i>
VJ4	true	8	-	7	<i>then at intervals of no more than 7 mins until</i>	<i>Frequent service at least every 10 mins until</i>
VJ5	true	6	3	8	<i>then at 3-8 minutes intervals until</i>	<i>Frequent service at least every 10 mins until</i>

Alternative XML Representation

Vehicle Journeys #1 & #5 could also be coded to use the same Route & Journey pattern as Vehicle Journey #2 , using start and end Dead Runs to indicate short workings for the route. **R1**, **R5**, **JP1**, **JP5** would then not be needed.

Notes

N.B. The variable phrasing for the Frequent Services is only available with the 20072.4 Enhanced publisher

Examples 2.4 - Delta Example (V2.4)

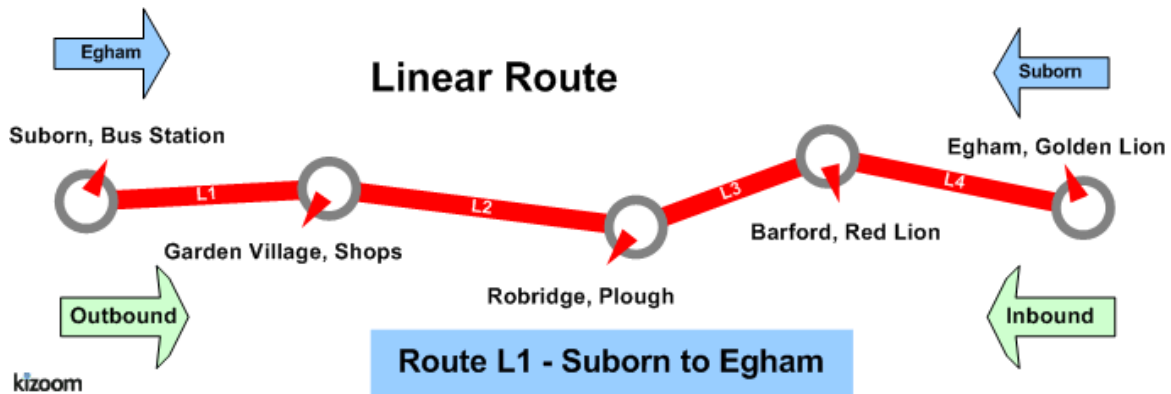
Summary

General Delta exchange of just the changes to a timetable. Uses the general_delta schema. This does not cross check key references. Changes are based on the Linear timetable example (see section).

- delta route.
- Changes to just individual elements

TransXChange XML Document	Particulars and Matrix output PDF	Route Map Output
delta.xml (One direction only)	not publishable	not publishable

Route Map



Timetable

Outbound			
Line L1	Journeys		
	#1		#2
Suborn, Bus Station	07:02	And then every 7 minutes until 18:30	19:00
Garden Village, Shops	07:20		19:20
Robridge, Plough	07:40		19:40
Barford, Red Lion	07:50		19:50
Egham, Golden Lion	08:00		20:00

The XML Representation

Outbound

- T Only a single outbound **VehicleJourney** instances v1_1, based on **JourneyPattern** *jp_1* : is changed
 - It has a revised **DepartureTime** of 7.02.

Notes

Page last updated: 2013/04/13

Examples 2.4 - Express route

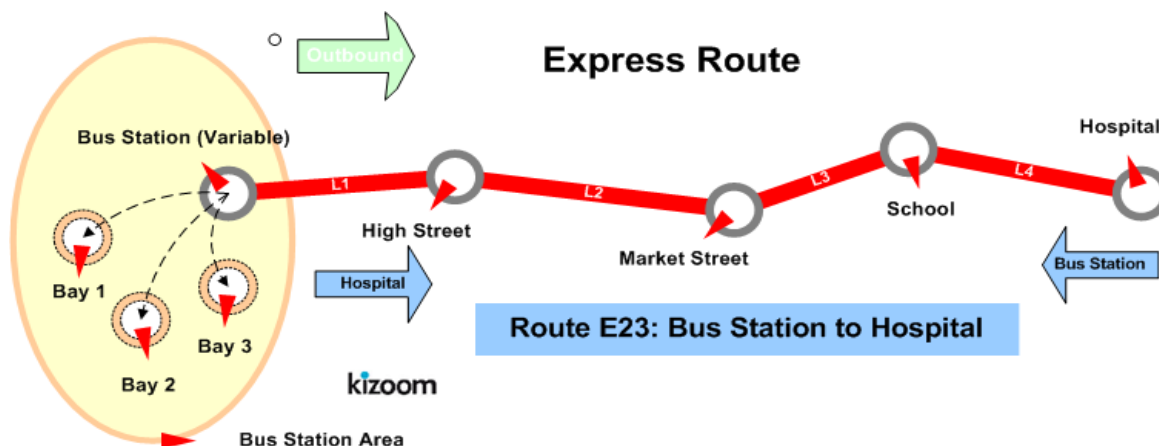
Summary

A linear route with express journey patterns running over it that omit stops.

- Express service.
- Short working.
- Vehicle journey timing link reuse.
- Overriding of Journey Pattern Timing Link Run Times with different values on the Vehicle Journey Timing links for some journeys.
- Additional Wait times at stop; on arrival, on departure, on first or intermediate stops.
- Monday to Sunday Service.
- Holiday Day Type Exclusions (see below).
- Local Stop Point definitions for an Off Street Bus Station: BCQ, BCE and BCS Stop Types, SMS stop codes.
- Local Stop Area definition for bus station.
- Vehicle Journey Footnote.
- Variable Bay Allocation (see below).
- General Schema.
- Marketing Name (TXC v2.4).

TransXChange XML Document	Particulars and Matrix output PDF	Route Map Output
express.xml (One direction only)	Particulars and Matrix PDF	Route Map PDF Route Map PDF (no background)

Route Map



Timetable

Bus Station to Hospital								
		#1	#2	#3	#4	#5	#6	Stop Types
		E23	E23	E23	E23	E23	E23	
<i>Bus Station, Bay Area</i>	dep	10:00	11:00	12:00	-	14:10	15:10	BCE + BCQ + BCS(x3)
<i>Tweeham, High St</i>	arr	10:03	11:03	-	-	14:18	15:18	BCT
	dep	10:03	11:03	-	-	14:28	15:28	
<i>Tweeham, Market St</i>	dep	10:07	-	12:07	13:07	14:37	15:37	BCT
<i>Tweeham, School</i>	arr	10:20	11:20	-	13:20	-	-	
	dep	10:20	11:20	-	13:30	-	-	BCT
<i>Tweeham, Hospital</i>	dep	10:29	11:29	12:29	13:39	-	-	BCT

- Service operates from 01/01/2004 until 13/06/2004
- Service operates Monday to Sunday
- Service does not run Christmas Day, Boxing Day, Good Friday, New Years Day, Late Summer Bank Holiday (Not Scotland), May Day, Easter Monday, Christmas Day Holiday, New Years Day Holiday, ChristmasEve, NewYearsEve
- Service does not run 02/06/2004
- Service runs 01/06/2004
- Services #1-#4 normally run from Bus Station Bay 1.
- Service #1 runs from Bus Station Bay2 2004-08-01 to 2004-10-
- Service #5 always runs from Bus Station Bay 3,

In this example the last journey overrides the default run times. In additional wait times are specified for some stops. The following table shows the timetable additionally annotated with the vehicle journey run times (default values inherited from the journey pattern shown in brackets) and the wait times (additional wait times prefixed by a +).

	JP Run Time (Mins)		VJ run & Wait	#1	VJ run & Wait	#2	VJ run & Wait	#3	VJ run & Wait	#4	VJ run & Wait	#5	#6
				E23		E23		E23		E23		E23	E23
		start		10:00		11:00		12:00				14:00	15:00
<i>Bus Station</i>		dep		10:00		11:00		12:00			+w10	14:10	15:10
	r3		(r3)		(r3)		(r3)		sr		+r8		
<i>High St</i>		arr		10:03		11:03	pass				+w10	14:08	15:08
		dep		10:03		11:03	pass		sr			14:28	15:28
	r4		(r4)		(r4)		(r4)				+r9		
<i>Market St</i>		dep		10:07	pass			12:07		13:07		14:37	15:37
	r13		(r13)				(r13)		(r13)		sr		

School		arr		10:20		11:20	pass			13:20		-	-
		dep		10:20		11:20	pass		+w10	13:30		-	-
	r9		(r9)		(r9)		(r9)		(r9)		sr		
Hospital		dep		10:29		11:29		12:29		13:39			

The XML Representation

One way encoding this example would be to have a separate route and journey pattern for each column, thus there would be five routes and five journey patterns, each with a single section. Since however the vehicle travels over the same route in the same order, but just passes by certain stops it is possible also to encode it in a less verbose manner by having a single journey pattern with a stop activity of pass at certain stops.

[XML Document](#)

Service Registration

- The service is not registered
- There is a single **Operator**.

Service Structure

- There is a single **Service** instance *SVI*, with one **Line** - 'I'.
- There are nine **StopPoint** instances.
 - Five stops make up a locally defined bus station, comprising an **Entrance** (bus stop type *BCE*), an off-street general **AccessArea** (bus stop type *BCQ*), and three **Bay** instances (bus stop type *BCS*).
 - There are four on-street stops, all references to existing NaPTAN stops.
- There is a single **RouteSection** *rs_1* with four **RouteLink** instances *rl_1-rl_4* connecting the stops.
- There is a single **JourneyPatternSection** *JPS1* made up of four **JourneyPatternTimingLink** instances, *JPTLI-4*, with **RunTime** values of 3, 4, 13, and 9 minutes respectively.
- There is a single **JourneyPattern** instances;
 - *JP_1*, section: *JS_1*
- There are seven **VehicleJourney** instances, all for *Ln_1* - 'E23', and using **JP1**. The instances all use the same set of **JourneyPatternTimingLink** instances, but define different stop activities (e.g. pass) to specify the different express stopping patterns:
 - *VJ_1*, with a departure time of 10:00, stopping at all stops. *VJ_1* appears in column #1.

- There is a variable stop allocation the bay in the bus station, specified on the **From** part of the first **VehicleJourneyTimingLink** - see below.
- **VJ_2**, with a departure time of *11:00* (column #2), which has a timing links annotated with a **VehicleJourneyStopUsage / Activity** of 'pass' for the *Market Street* stop to indicate that the bus does not stop there.
- **VJ_3**, with a departure time of *12:00* (column #3), which has a **VehicleJourneyStopUsage / Activity** of 'pass' for *High Street*. and for *School*.
- **VJ_4**, with a departure time of *13:07* (column #4), which has a short working: a dummy **StartDeadRun** is used to indicate that the service starts at *Market Street*.
 - There is an extra **WaitTime** of *10 Minutes* on departure from the *School* stop, specified on the **From** usage of **VehicleJourneyTimingLink VJ4_TL4**.
- **VJ_5**, with a departure time of *14:00* (column #5), which has a short working: a dummy **EndDeadRun** is used to indicate that the service ends at *Market Street*.
 - A **DynamicDestinationDisplay** of *Market Street*. is used to override the default destination of *Hospital*.
 - Both links of this journey takes longer so has a explicit override values of *8.00* and *9.00* minutes for the **RunTime** on the **VehicleJourneyTimingLink** instances of the Journey.
 - There is an extra **Waittime** of *10 Minutes* on arrival at the *High Street* stop, specified on the **To** usage of **VehicleJourneyTimingLink VJ5_TL1**.
- **VJ_6**, with a departure time of *15:00* (column #6), Reuses the links from **VJ_5**.
- **VJ_7**, with a departure time of *18:00* (column #6), Reuses the links from **VJ_1**.

The Operating Days

- The **Service operating period** for **SV_1** starts on *02/01/2002* for all vehicle journeys and continues indefinitely
- The **OperatingProfile** for the **Service SV_1** states values that apply to all journeys unless overridden on a journey pattern or individual vehicle journey.
 - The **RegularDayType** specifies journeys of the service run **Monday** to **Sunday** every week of the year
 - The **BankHolidayOperation / DaysOfNonOperation** states journeys of the service do not run **LateSummerBankHolidayNotScotland**, **MayDay**, **EasterMonday**, **ChristmasDayHoliday**, **NewYearsDayHoliday**, **ChristmasEve**, **NewYearsEve**.
 - The **SpecialDaysOperation / DaysOfOperation** states that journeys of the service are will run on *01/06/2004* regardless.

- The **SpecialDaysOperation / DaysOfNonOperation** states that all services will not run on *02/06/2004*.

Variable Stop Allocation

The use of bays in the bus station for the stop varies for different vehicle journeys within the day, and on different days

- The **Route** and **JourneyPattern** specify that the **Service** goes to a **StopPoint** of type **BCQ** in the *Bus Station*.
- The *Bus Station* is also defined as a **StopArea** of type *GBCS*, and the *BCQ StopPoint* and the three bay *StopPoint* instances of type *BCS* are assigned to it. (This is good practice but is not strictly required in order for the variable allocation to work).
- On the **JourneyPattern** *jp_1*, a **DefaultStopAllocation** assigns the '*Bus Station*' *BCQ* stop to depart from *BCS* stop '*Bay 1*' unless otherwise specified.
- The **VehicleJourney** instances override this as follows:
 - *VJ_1*, with a departure time of *10:00*,
 - *VJ_2*, *VJ_3*, *VJ_4*, use the journey pattern default.

Alternate XML Representation

It would also be possible instead of using dead runs to indicate the short working, to simply use a 'pass' activity for the two successive end stops not visited on each of the two journeys that are short workings.

Examples 2.4 - Eye route

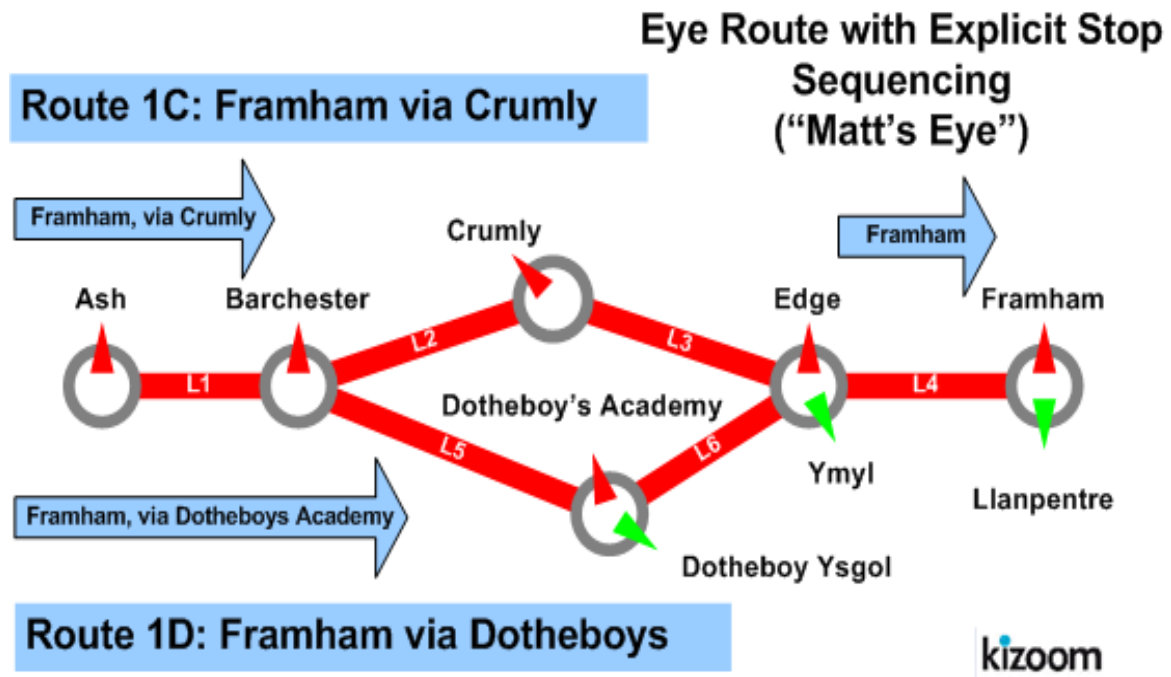
Summary

A line running over an eye physical shaped route, with two alternative branch variants.

- Multiple routes composed of some shared route sections.
- Multiple journey patterns composed of some shared journey pattern sections.
- Use of Stop Sequence Numbers (see below) to control the stop row order in the timetable matrix.
- Bilingual support (see below) in Welsh.
- Dynamic Destination Displays (see below).
- Different Running Boards (see below).
- Registration Schema.

Published as: [PDF \(English\)](#) Published as: [PDF \(Cymraeg\)](#)

Route Map



Timetable

Ash to Framham, via Crumly or Dotheboy's

		#1	#2	SequenceNumber
English	Welsh	1C	1D	#
Ash		10:00	11:00	1

Barchester		10:10	11:10	2
Crumley		10:12	-----	3
Dotheboy's	Dotheboy ysgol	-----	11:15	4
Edge	Ymyl	10:15	11:21	5
Framham	Llanpentre	10:20	11:26	6

The XML Representation

[XML Document \(English\)](#)

[XML Document \(Cymraeg\)](#)

Service Registration

- There is a single **Operator**, **O1** - 'Dai Larid'
- The service is registered, classified as a rural service with **NormalStopping**

Service Structure

- There is a single **Service** instance **SV1**, with two **Line** instances: '**IC**' and '**ID**'
- There are five **StopPoint** instances. Three of the stops have bilingual names.
- There are four **RouteSection** instances, **RS1-RS4**, with **RouteLink** instances to connect the six stops. See Diagram below.
 - **RS1** has **RL1** only, connecting **A-B**
 - **RS2** has **RL2** & **RL3**, connecting **B-C-E**.
 - **RS3** has **RL5** & **RL6**, connecting **B-D-E**.
 - **RS4** has **RL4** only, connecting **D-F**
- There are two **Route** instances;
 - **R1**, comprising **RouteSection** instances **RS1-RS2-RS4**, such that the **RouteLinks** run **A-B-C-E-F**.
 - **R2**, comprising **RouteSection** instances **RS1-RS3-RS4**, such that the **RouteLinks** run **A-B-D-E-F**.
- There are four **JourneyPatternSection** instances **JS1-JS4**; corresponding to the route sections, and containing **JourneyPatternTimingLink** instances. **JPTL1** - **JPTL6**, which project onto the respective **RouteLink** instances **RL1-RL6**.
- There are two **JourneyPattern** instances,
 - **JP1** running **A-B-C-E-F** over **R1**, and comprising **JourneyPatternSection** instances **JS1-JS2-JS4**, with timings on each of the timing links. A preferred stop sequence is specified:
 - **JS1** [**JPTL1** [10 mn, from:1, to: 2]]
 - **JS2**

[**JPTL2** [2 mn, from:2, to: 3], J

PTL3 [3 mn, from:3, to: 5] with **DynamicDestination** 'Framham' and Vias 'Edge'].

- **JS4** [**JPTL4** [4 mn, from:5, to: 6] with **DynamicDestination** 'Framham' and Vias/None].
- The **Vias** at the beginning show : 'Crumley', 'Edge'.
- **JP2** running A-B-D-E-F over **R2**, and comprising **JourneyPatternSection** instances **JS1-JS3-JS4** A with timings on each of the timing links. A preferred stop sequence
 - **JS1** [**JPTL1**[10 mn, from:1, to: 2]]
 - **JS3**

[**JPTL5** [5 mn, from:2, to: 4],

JPTL6 [6 mn, from:4, to: 5] with **DynamicDestination** 'Framham' and Vias 'Edge'].

- **JS4** [**JPTL4** [4 mn, from:5, to: 6]] With **DynamicDestination** 'Framham' and Vias/None
- There are two **VehicleJourney** instances:
 - **VJ1** following **JP1** over A-B-C-E-F , starting at 10:00.
 - The **DynamicDestination** at the beginning shows : 'Framham, Via Crumly'.
 - The **vias** at the beginning show : 'Crumley', 'Edge'.
 - The journey pattern specifies that the **DynamicDestination** changes at Edge to just 'Framham'.
 - **VJ2** following **JP2** over A-B-D-E-F , starting at 11:00, with a longer time for **JPTL4** .
 - **JS4** [**JPTL4**[6 mn, from:5, to: 6]].
 - The **DynamicDestination** at the beginning shows : 'Framham, Via Dotheboy's'.
 - The **vias** at the beginning show : 'Crumley', 'Dotheboy's'.
 - The journey pattern specifies that the **DynamicDestination** changes at Edge to just 'Framham' .

Operational Data

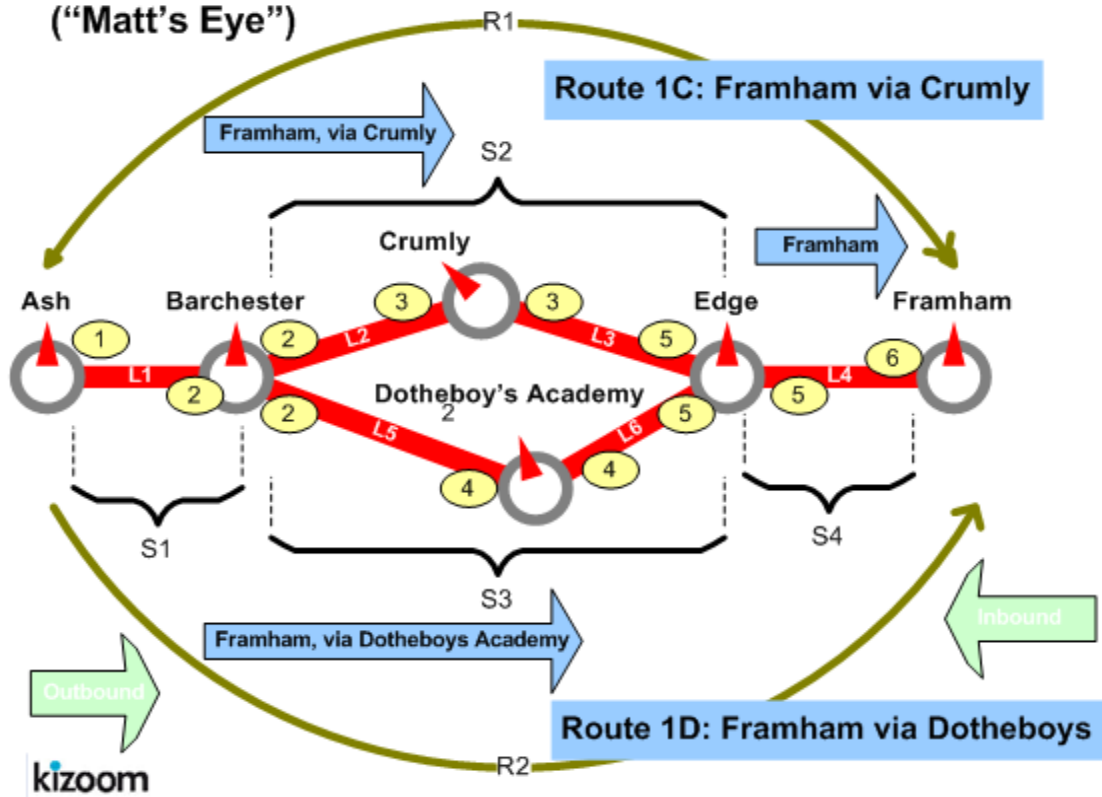
- Different **DynamicDestinationDisplays** are used for the service via *Crumley* and via *Dotheboy's*.

Use of Sections & Stop Sequence Numbers

Sections are used to reuse links between journeys.

Stop sequence numbers are used to coerce a specific ordering of the stops within a matrix timetable when published. The following diagram shows the journey pattern sections, with individual timing links annotated with stop section numbers.

Eye Route with Explicit Stop Sequencing ("Matt's Eye")



Bilingual Support

The last three stops of the route lie within a Welsh speaking area and have bilingual stop names. Most text elements in TransXChange have

- The NaPTAN **StopPoint** definitions include alternative common names and other descriptor elements in Welsh
- The primary language for the TransXChange document is specified on the root **TransXChange** element - for Welsh this is 'cy'.
 - When published in **Welsh**, the Welsh versions of the stop names are used, along with any Welsh alternatives for Destinations, Notes and other text that is available from the originating system.

Cymraeg [XML](#) | published as [PDF](#)

Alternative Representation

The following shows the same example published without explicit stop sequence numbers

Unsequenced [XML](#) | published as [PDF](#)

Page last updated: 2013/04/13

Examples 2.4 - Flexible route

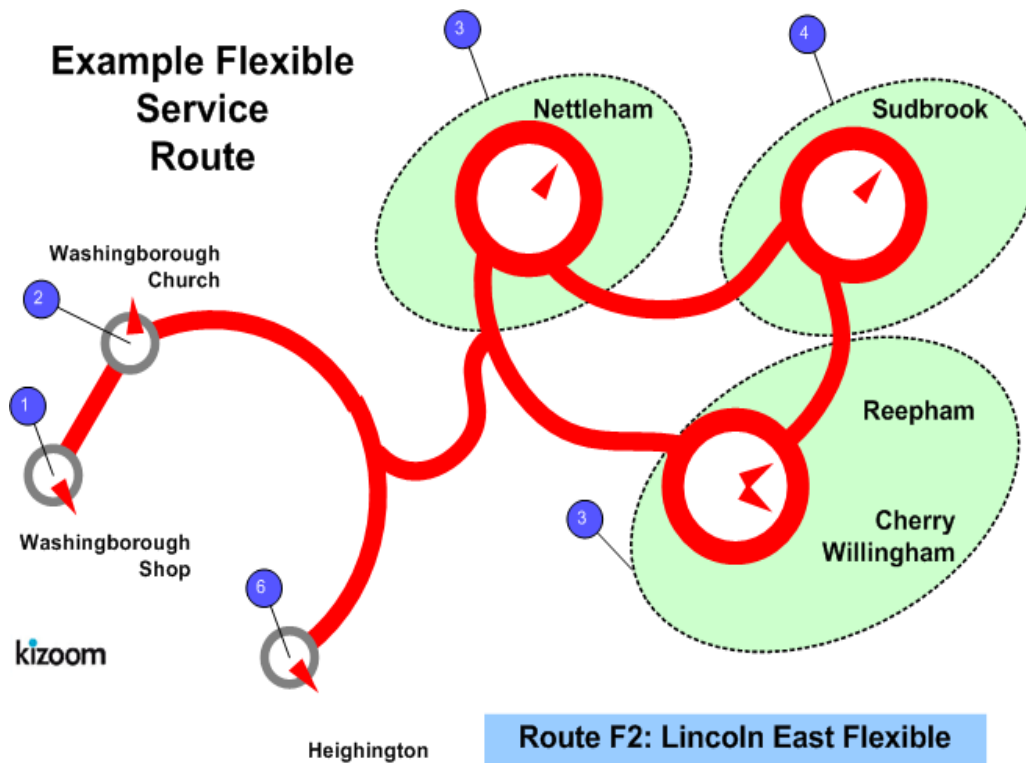
Summary

A flexible service route.

- Flexible zones for pick up and set down.
- Fixed stops for pick up and set down.
- Flexible time bands (see below).
- Registration Schema.

Published as: [PDF](#)

Route Map



Timetable		
	#1	#2
	Monday to Friday, excluding public holidays	Bank Holiday Mondays
Band 1	09:00 - 18:00	10:00-12:00
Band 2		16:00:1400

The XML Representation

[XML Document](#)

Service Structure

- Here is a single **Service** instance *SV1*, with one **Line** - '*L1*', with **LineName** '*H145*'.
- The **Registration** is for a single **Operator**.
- There is a single **FlexibleJourneyPattern**, *jp_1*, which references six **StopPoint** instances:
 - Three **Fixed** stops: *Washingborough Church*, *Washingborough Shop*, *South Highton*.
 - Three **FlexibleZone** stops: *Nettleham*, *Sudbrook* and *Cherry Willingham*. The *Cherry Willingham* stop is also associated with the *Reepham* locality.
- There are two single **FlexibleVehicleJourney** instances, both based on **JourneyPattern** *jp_1*:
 - The first, *VJ1*, follows *jp_1* on the default operational days of the service. It has a single **TimeBand** from **09:00** to **18:00** any changes to the timings on the **JourneyPatternTimingLink** instances.
 - The second *VJ2* runs only on **Bank Holiday Mondays** and has a start time of **10.00**. It has a frequent service of every *10 minutes*, and an end time of *16:30*.

The Operating Days

- The **Service OperatingPeriod** for *SV_1* starts on *02/01/2005* for all vehicle journeys and continues indefinitely
- The service **OperatingProfile** says it runs **MondayToFriday** except on any bank holidays.
 - **The RegularDayType** specifies journeys of the service run **MondayToFriday** every week.
 - **The BankHolidayOperation / DaysOfNonOperation** states journeys of the service do not run on public holidays or early run off days. A special public holiday is also defined for *2005/13/0*.
- **VehicleJourney** *Vj2* has an override **OperatingProfile** which specifies it runs only on **BankHolidayMondays**.
 - **The RegularDayType** specifies journeys of the service run **Holidays** only.
 - **The BankHolidayOperation / DaysOfoperation** states journeys of the service run on **HolidayMondays**.

Examples 2.4 - Footnotes

Summary

Registration for a large route with many footnotes.

- Many stops (101).
- Many vehicle journeys (144).
- Some operating date rule.
- Layover points and Operational date.
- Use of Stop Sequence numbers to control row order in matrix.
- Complex footnote conditions and optimisation (2007 Enhanced publisher only).
- Page overflow in both horizontal and vertical directions.

Published as: [PDF](#) Matrix only: [PDF](#)

Route Map

See Map PDF

Timetable

See PDF

The XML Representation

[XML Document](#)

- The service has multiple individual services and more journeys than will fit on a single page.

Service Registration

- The service is Registered.
- There is a single **Operator**.

Service Structure

- There is a single **Service** instance *SVI*, with two **Lines** - '215' and '215A'.
- There are 101 **StopPoint** instances, more than will fit vertically on a single page.
- There are 10 **Route** instances and 33 **RouteSection** instances with **RouteLink** instances connecting the stops.
- There are 33 **JourneyPatternSection** corresponding to the **RouteSections**.

- There are 10 **JourneyPattern** instances, five outbound and five return.
- There are 144 **VehicleJourney** instances, all for *Ln_1*, all based on JP_1.

Notes

N.B. The *footnote optimisation* is only available with the 2007 2.4 Enhanced publisher. Prior to that the individual journeys would all be published as separate Frequency Group columns.

Page last updated: 2013/04/13

Examples 2.4 - Grouping route (TXC v2.4)

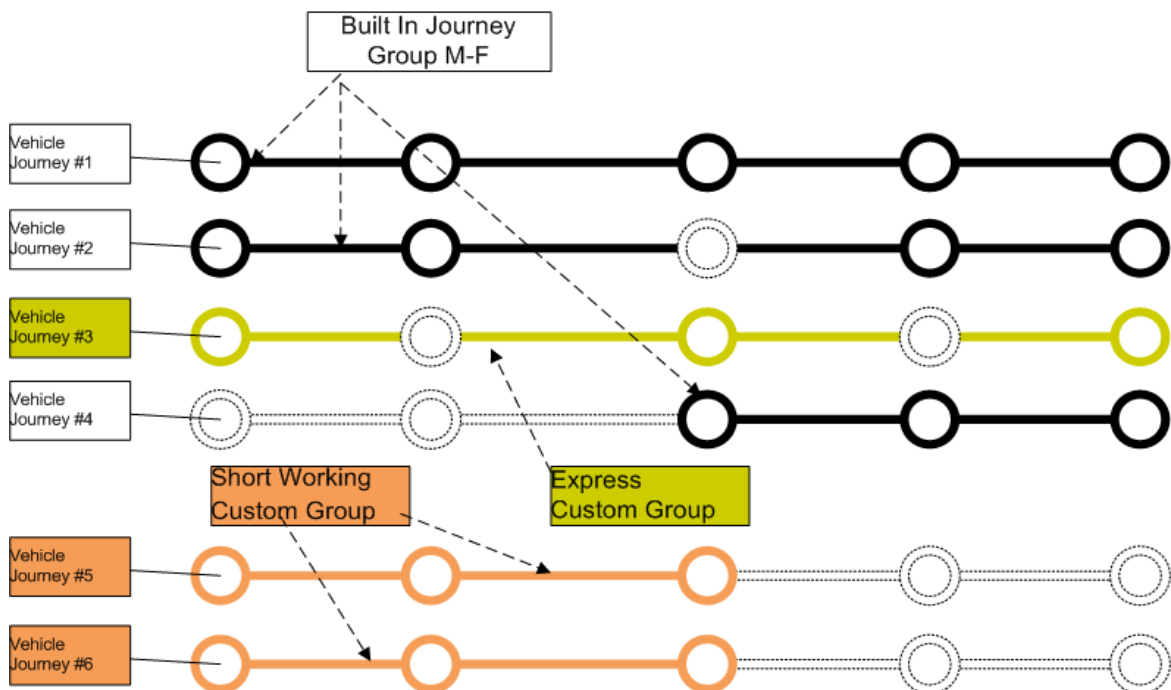
Summary

A linear route with Custom Grouping of journeys to create additional bed.

- Custom Grouping (see below) of service journeys (TXC v2.4).
- Override labels for Journey groupings (TXC v2.4).
- Suppression of a Built-in journey Grouping / Matrix bed (TXC v2.4).
- Short working.
- Vehicle journey timing link reuse.
- Overriding of Journey Pattern Timing Link Run Times with different values on the Vehicle Journey Timing links for some journeys.
- Additional Wait times at stop; on arrival, on departure, on first or intermediate stops.
- Monday to Sunday Service. Day Type Restrictions (see below) on certain journeys
- Holiday Day Type Exclusions (see below).
- Vehicle Journey Footnote.
- Operational Calendar (see below) (TXC v2.4). Day Assignments
- Data Rights (see below). (TXC v2.4)
- Registration Schema.

TransXChange XML Document	Particulars and Matrix output PDF	Route Map Output
Grouping.xml (One direction only)	Particulars and Matrix PDF	Route Map PDF Route Map PDF (no background)

Route Map



Timetable

This example segregates the journeys in the following table into four beds (as coloured white, green , orange, yellow)

Saffron Crossroads to Weldon Road								
		#1	#2	#3	#4	#5	#6	#7
		E23	E23	E23	E23	E23	E23	E23
Saffron Crossroads	dep	10:00	11:00	12:00	-	14:10	15:10	18:00
Glenhills Boulevard	arr	10:03	11:03	-	-	14:18	15:18	18:03

	dep	10:03	11:03	-	-	14:28	15:28	18:03
<i>Roehampton Drive</i>	dep	10:07	-	12:07	13:07	14:37	15:37	18:07
<i>Carleton Drive</i>	arr	10:20	11:20	-	13:20	-	-	18:20
	dep	10:20	11:20	-	13:30	-	-	18:20
<i>Weldon Road</i>	dep	10:29	11:29	12:29	13:39	-	-	18:29

- Service operates from 01/01/2004 until 13/06/2004
- Service operates Monday to Sunday

1. Built-in Bed: Saffron Crossroads to Weldon Road E23 Regular Service

The first three journeys and the last one are added to the built in beds for Monday to Friday, and Saturday (which are then merged into a single bed because they are the same T

Outbound, Monday to Saturday

		#1	#2	#4	#7
		E23	E23	E23	E23
<i>Saffron Crossroads</i>	dep	10:00	11:00	-	18:00
<i>Glenhills Boulevard</i>	arr	10:03	11:03	-	18:03
	dep	10:03	11:03	-	18:03
<i>Roehampton Drive</i>	dep	10:07	-	13:07	18:07
<i>Carleton Drive</i>	arr	10:20	11:20	13:20	18:20
	dep	10:20	11:20	13:30	18:20
<i>Weldon Road</i>	dep	10:29	11:29	13:39	18:29

2. Custom Bed #1: Saffron Crossroads to Weldon Road Express

The express Journey is shown in a custom bed

Outbound, Express

		#3
		E23
<i>Saffron Crossroads</i>	dep	12:00
<i>Roehampton Drive</i>	dep	12:07
<i>Weldon Road</i>	dep	12:29

3. Custom Bed #2: Saffron Crossroads to Weldon Road, Tuesday & Thursday only

The two short working Journeys #5 #6 are shown in a another custom bed

		#5	#6
		E23	E23
<i>Saffron Crossroads</i>	dep	14:10	15:10
<i>Glenhills Boulevard</i>	arr	14:18	15:18
	dep	14:28	15:28
<i>Tweeham, Roehampton Drive</i>	dep	14:37	15:37

Service runs Tuesdays and Thursdays only

4. Suppressed Bed

The Sunday built in journey grouping , is suppressed completely: this means that the last Journey, #7, which is in both the *Monday to Friday*, *Saturday* and the *Sunday* bed, appears as a *Monday to Saturday* journey but that there is no *Sunday* bed.

		#7
		E23
<i>Saffron Crossroads, Bay Area</i>	dep	18:00
<i>Glenhills Boulevard</i>	arr	18:03
	dep	18:03
<i>Roehampton Drive</i>	dep	18:07
<i>Carlton Drive</i>	arr	18:20

	dep	18:20
<i>Weldon Road</i>	dep	18:29

In this example the last journey overrides the default run times . In additional wait times are specified for some stops. The following table shows the timetable additionally annotated with the vehicle journey run times (default values inherited from the journey pattern shown in brackets) and the wait times (additional wait times prefixed by a +).

	JP Run Time (Mins)		VJ run & Wait	#1	VJ run & Wait	#2	VJ run & Wait	#3	VJ run & Wait	#4	VJ run & Wait	#5	#6	VJ run & Wait	#7
				E23		E23		E23		E23		E23	E23		E23
		start		10:00		11:00		12:00				14:00	15:00		18:00
<i>Saffron Crossroads</i>		dep		10:00		11:00		12:00			+w10	14:10	15:10		18:00
	r3		(r3)		(r3)		(r3)		sr		+r8			(r3)	
<i>Glenhills Boulevard</i>		arr		10:03		11:03	pass				+w10	14:08	15:08		18:03
		dep		10:03		11:03	pass		sr			14:28	15:28		18:03
	r4		(r4)		(r4)		(r4)				+r9			(r4)	
<i>Roehampton Drive</i>		dep		10:07	pass			12:07		13:07		14:37	15:37		18:07
	r13		(r13)				(r13)		(r13)		sr			(r13)	
<i>Carlton Drive</i>		arr		10:20		11:20	pass			13:20		-	-		18:20
		dep		10:20		11:20	pass		+w10	13:30		-	-		18:20
	r9		(r9)		(r9)		(r9)		(r9)		sr			(r9)	
<i>Weldon Road</i>		dep		10:29		11:29		12:29		13:39					18:29

The XML Representation

One way encoding this example would be to have a separate route and journey pattern for each column, thus there would be five routes and five journey patterns, each with a single section. Since however the vehicle travels over the same route in the same order, but just passes by certain stops it is possible also to encode it in a less verbose manner by having a single journey pattern with an stop activity of pass at certain stops.

[XML Document](#)

Service Registration

- The service is not registered

- There is a single **Operator**. code "*Op_02*", with id **01**.

Service Structure

- There is a single **Service** instance **SV1**, with one **Line** - '*1*'.
- There are six on-street **StopPoint**, all references to existing NaPTAN stops.
- There is a single **RouteSection** **rs_1** with four **RouteLink** instances **rl_1-rl_4** connecting the stops.
- There is a single **JourneyPatternSection** **JPS1** made up of four **JourneyPatternTimingLink** instances, **JPTL1-4**, with **RunTime** values of 3, 4, 13, and 9 minutes respectively.
- There is a single **JourneyPattern** instances;
 - **JP_1**, section: **JS_1**
- There are six **VehicleJourney** instances, all for **Ln_1** - '*E23*', and using **JP1**. The instances all use the same set of **JourneyPatternTimingLink** instances, but define different stop activities (e.g. pass) to specify the different express stopping patterns:
 - **VJ_1**, with a departure time of *10:00*, stopping at all stops. **VJ_1** appears in column #1.
 - There is a variable stop allocation the bay in the Saffron Crossroads, specified on the **From** part of the first **VehicleJourneyTimingLink** - see below.
 - **VJ_2**, with a departure time of *11:00* (column #2), which has a timing links annotated with a **VehicleJourneyStopUsage / Activity** of 'pass' for the *Roehampton Drive* stop to indicate that the bus does not stop there.
 - **VJ_3**, with a departure time of *12:00* (column #3), which has a **VehicleJourneyStopUsage / Activity** of 'pass' for *Glenhills Boulevard*. and for *Carlton Drive*.
 - **VJ_4**, with a departure time of *13:07* (column #4), which has a short working: a dummy **StartDeadRun** is used to indicate that the service starts at *Roehampton Drive*.
 - There is an extra **WaitTime** of *10 Minutes* on departure from the *Carlton Drive* stop, specified on the **From** usage of **VehicleJourneyTimingLink** **VJ4_TL4**.
 - **VJ_5**, with a departure time of *14:00* (column #5), which has a short working: a dummy **EndDeadRun** is used to indicate that the service ends at *Roehampton Drive*.
 - A **DynamicDestinationDisplay** of *Roehampton Drive*. is used to override the default destination of *Weldon Road*.
 - Both links of this journey takes longer so has a explicit override values of *8.00* and *9.00* minutes for the **RunTime** on the **VehicleJourneyTimingLink** instances of the Journey.
 - There is an extra **WaitTime** of *10 Minutes* on arrival at the *Glenhills Boulevard* stop, specified on the **To** usage of **VehicleJourneyTimingLink** **VJ5_TL1**.

- **VJ_6**, with a departure time of *15:00* (column #6), Reuses the links from **VJ_5**.

The Operating Days

- The **Service operating period** for **SV_1** starts on *02/01/2002* for all vehicle journeys and continues indefinitely
- The **OperatingProfile** for the **Service SV_1** states values that apply to all journeys unless overridden on a journey pattern or individual vehicle journey.
 - The **RegularDayType** specifies journeys of the service run **Monday** to **Sunday** every week of the year
 - The **BankHolidayOperation / DaysOfNonOperation** states journeys of the service do not run **LateSummerBankHolidayNotScotland**, **MayDay**, **EasterMonday**, **ChristmasDayHoliday**, **NewYearsDayHoliday**, **ChristmasEve**, **NewYearsEve**.
 - The **SpecialDaysOperation / DaysOfOperation** states that journeys of the service are will run on *01/06/2004* regardless.
 - The **SpecialDaysOperation / DaysOfNonOperation** states that all services will not run on *02/06/2004*.
 - **VJ_5**, **VJ_6**, Have an override profile that say they run only on **Tuesdays** & **Thursdays**

The Journey Groupings

- Three *Outbound* Built in Journey Groupings are included
 - **MondayToFridayJourneyGrouping** with description *"E232 Regular"* and **vias** *"Glenhills Boulevard"*, *"Market"* and *"Carlton Drive"*.
 - **SaturdayJourneyGrouping** with description *"E232 Saturday Short Service"* and **vias** *"Glenhills Boulevard" only*.
 - **SundayJourneyGrouping** with **Contents / None** to suppress the matrix.
- Two **CustomJourneyGroupings** are added for
 - **cusgrp_01** with description *"E232 Express"* and **vias / None**. It includes **VehicleJourneys Vj5** and **Vj6**.
 - **cusgrp_02** with description *"E232 Sunday Short"* and **vias** *"Glenhills Boulevard"*. It includes **VehicleJourney Vj3** only.

The Data Rights

- Two sets of **DataRights** are specified
 - For the **Operator. 01. (Speedy buses)** there is a single right **dr_01** permitting unrestricted use under the given terms and conditions
 - For the **Contributor. 02 (Rival buses)** there is a single right **dr_02** requiring a commercial licences for use of tagged items and use under separate terms and conditions.

- The **DataRights** are referenced as follows
 - The document references ***dr_01***, indicating all of the contents except where specifically excluded are available under the associated terms.
 - **VehicleJourney Vj3** reference ***dr_02*** indicating it is owned by *Rival buses* and subject to different rights.

The Calendar

- A **Calendar** specifies the **DayAssignments** for *Christmas 2010*
 - Three separate **DayAssignments** specify the service for well known bank holidays **ChristmasEve**, **ChristmasDay** and **BoxingDay** to use the day types ***day_01***, ***day_02***, ***Day_02*** respectively.
 - Four **DayAssignments** specify the service for a special holiday "*Slevin Day*" as type ***Day_03***.

Page last updated: 2013/03/30

Examples 2.4 - Hail & Ride route

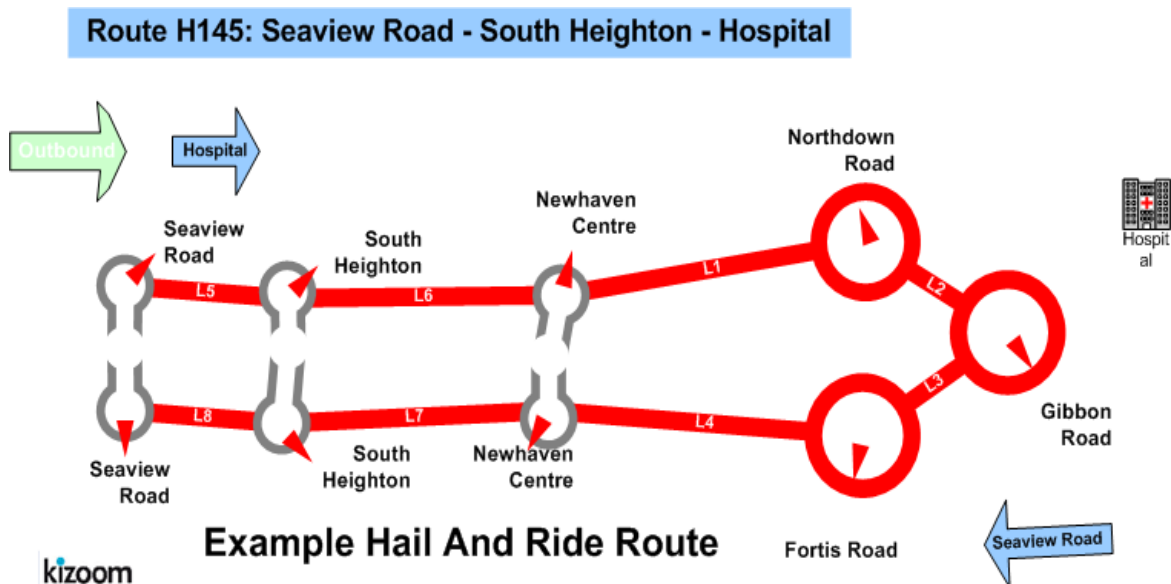
Summary

A hail and ride route:

- Use of Hail and Ride sections.
- Route with Hail and Ride only sections, Non-Hail and Ride stops and mixed stop.
- Full lollipop.
- Short Notice Registration (see below)
- Some locally defined stops.
- FrequentService, specified as minutes past the hour, but not a Frequent Service.
- Workflow Attributes (TXC v2.4).
- Scottish Holidays, including St Andrew's Day & January 2nd (TXC v2.4).

Published as: [PDF](#)

Route Map



Timetable				
Stops		Journeys		
		#1	#2	#3
Seaview Road		10:10	And then at 10 and 42 minutes past the hour until 1642	19:00
South Heighton		10:30		9:30
Newhaven		10:40		20:10

<i>Centre</i>				
<i>Northdown Road</i>	<i>Hail & Ride section</i>	10:50		19:40
<i>Gibbon Road</i>	<i>Hail & Ride section</i>	11:00		19:50
<i>Fortis Road</i>	<i>Hail & Ride section</i>	11:10		20:00
<i>Newhaven Centre</i>		11:20		20:10
<i>South Heighton</i>		11:30		20:20
<i>Seaview Road</i>		11:50		20:40

The XML Representation

[XML Document](#)

Service Registration

- The **Registration** is for a single **Operator**.
- The **Registration** is circulated to three authorities.
- It has **ShortNoticeRegistration** details specified as well

Service Structure

- Here is a single **Service** instance **SVI**, with one **Line** - '**LI**', with **LineName** '**H145**'.
- There are six **StopPoint** instances:
 - Three fixed stops: '*Seaview Road, South Heighton, Newhaven Centre*'
 - Three **HailAndRide** section stops: '*Northdown Road, Gibbon Road Fortis Road*'. '*Gibbon Road*' also has a fixed stop.
- There are three **RouteSection** *instances*:
 - **rs_b1** connecting the outward handle of the lollipop.
 - There are two **RouteLink** instances **rl_b5, rl_b6**, running '*Seaview Road - South Heighton - NewHaven Centre*'.
 - **rs_b2** connecting the hail and ride stops in the lollipop ring.
 - There are four **RouteLink** instances **rl_b1 - rl_b4**, running '*NewHaven Centre - South Heighton - Seaview Road*'.
 - **rs_b3** connecting the return handle of the lollipop.
 - There are two **RouteLink** instances **rl_b7 rl_b8**, running '*NewHaven Centre - South Heighton - Seaview Road*'.

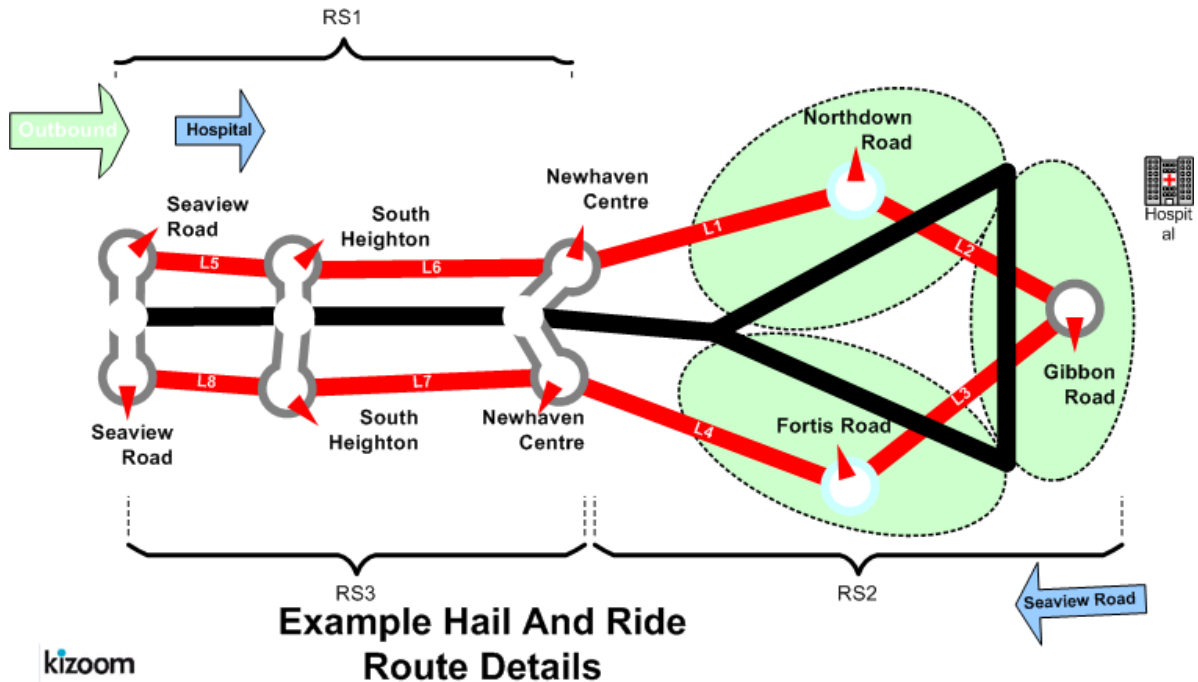
- There is a single **Route** *r_b1* containing the **RouteSections** *rs_b1*, *rs_b2*, *rs_b3*.
- There are three **JourneyPatternSections**:
 - *js_b1* covering the outbound traversal of *rs_2*.
 - There are two **JourneyPatternTimingLink** instances, *jptl_b5*, *jptl_b6*, each referencing the corresponding route link.
 - *js_b2* connecting the hail and ride stops in the lollipop ring.
 - There are four **JourneyPatternTimingLink** instances, *jptl_b1* - *jptl_b4*, each referencing the corresponding route link. The links are marked as **HailAndRide** links
 - *js_b3* covering the return traversal of *rs_2*.
 - There are two **JourneyPatternTimingLink** instances, *jptl_b7*, *jptl_b8*, each referencing the corresponding route link.
- There is a single **JourneyPattern**, *jp_b1*, that follows route *r_b1*.
- There are two **VehicleJourney** instances based on **JourneyPattern** *jp_b1*:
 - VJ_1, follows JP1 without any changes to the timings on the **JourneyPatternTimingLink** instances. It has a start time of 10.00.
 - The **Frequency** element specifies that is a Frequency Based service with a **MinutesPastTheHour** values of *10* and *42 minutes* past the hour, and an end time of 16:42. For which the publisher generates both column #1 and column #2. The service is not a statutory **Frequent Service**, so **FrequentService** is *false*
 - VJ_2, follows JP1 without any changes to the timings on the **JourneyPatternTimingLink** instances. It has a start time of *19.40*.

The Operating Days

- The service **OperatingProfile** says it runs **Monday** to **Friday** every day of the year.

Hail And Ride

Route H145: Seaview Road - South Heighton - Hospital



Notes

This example was refined for in the 2.4 Enhanced publisher to *not* be a Frequent Service.

Page last updated: 2013/04/13

Examples 2.4 - Interchange route

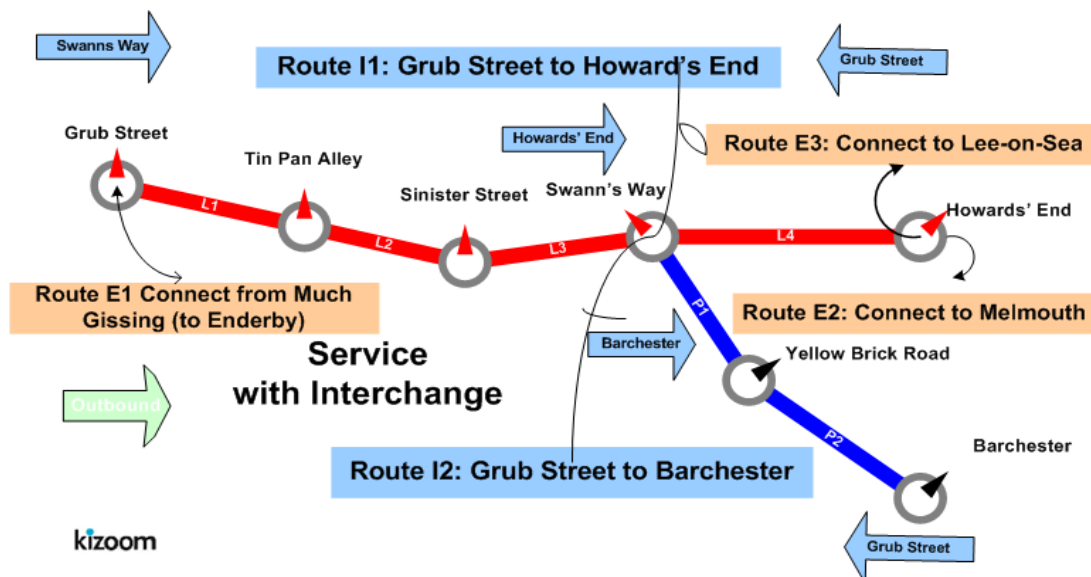
Summary

Two linear routes, connected with an interchange to make a single service.

- Linear route, with different stop visiting pattern at one end.
- An Interchange (see below) between two services.
- Connecting Vehicle Journeys (TXC v2.4)
- An Interchange with an incoming external feeder connecting vehicle journey. (TXC v2.4)
- An Interchange with two outgoing external distributor vehicle journeys. (TXC v2.4)
- Express stops, i.e. "Pass" activity where the bus does not stop.
- Frequency Based journey times, specified as an interval, but not a frequent service.
- Combining operating days (see below) from service, journey pattern and vehicle journey level.
- Use of Timetable Notes.
- Use of Stop sequence numbers to control Order of rows.
- More than one operator.
- Return Journeys: Outbound and Inbound (see below) Timetable.
- Serviced Organisation and School dates (see below).
- Line Colours (TXC v2.4).
- Registration Schema.

Published as [PDF](#)

Route Map



Timetable

Line I (Outbound) - Grub Street to Howard's End or Barchester					
	#1	#2	#3	#4	#5
Name\Line	I1	I1		I1	I2
	(++1)	(++2)		(++3)	(++4)
	School	Mon-Sat		Mon-Fri	Wed-W1-W3
Grub Street	8:02	9:02	? every 15 minutes until 11:47	12:02	--
Tin Pan Alley	8:12	9:12		12:12	--
Sinister Street	8:32	9:32		12:32	--
Swann's Way	8:40	9:40		12:40	12:45 (++4A)
Yellow Brick Road	--	--	--	--	12:48
Barchester	--	--	--	--	12:58
Howard's End	8:50	9:50	--	12:50	--

Timetable Notes		
Lines A1 & B1	Service operates from 02/01/2004 until further notice	
Service operates Monday to Friday		
Service does not run Christmas Day, New Years Day, Good Friday, Boxing Day		
Service runs Bank Holiday Mondays		SV1
Service runs 02/06/2004		SV1
Service does not run 01/06/2004		SV1
Service does not run 9/12/2004 -15/12/2004		SV1
(++1)	Journey runs during term days of Dotheboy's Academy	VJ1
Journey does not run during holidays of Dotheboy's Academy		VJ1
Journey does not run May Day		VJ1
Journey available 12/12/2004		JP1
Journey not available 11/1/2004 until 11/11/2004.		JP1
(++2)	Journey runs Monday to Saturday	VJ2
	Journey runs New Year's Day, Boxing Day	VJ2
	Journey available 12/12/2004	JP1
	Journey not available 11/1/2004 until 11/11/2004.	JP1
(++3)	Journey available 12/12/2004	JP1

	Journey not available 11/1/2004 until 11/11/2004.	JP1
(++)4)	Journey runs Wednesdays on 1 st and 3 rd weeks in month	VJ4
	Service does run on New Years day and Boxing day	VJ4
	Journey available 02/05/2004	VJ4
	Journey not available 04/08/2004-09/08/2004, 05/08/2005-10/08/2005	VJ4
(++)4A)	Connection only available 02/01/2004-17/10/2004.	

Line I (Inbound) - Barchester or Howard's End to Grub Street					
	#1	#2	#3	#4	#5
Name\Line	I1	I1		I1	I2
	(++)1)	(++)2)		(++)3)	(++)4)
	School	Mon-Sat		Mon-Fri	Wed-W1-W3
Howard's End	9:00	10:00	?	13:00	--
Barchester	--	--	--	--	<13:15
Yellow Brick Road	--	--	--	--	12:48
Swann's Way	8:40	9:40	every 15 minutes until 12:45	12:40	12:45 (++)4A)
Sinister Street	8:32	9:32		12:32	
Tin Pan Alley	8:12	9:12		12:12	
Grub Street	8:02	9:02		12:02	

The XML Representation

[XML Document](#)

Service Registration

- The service is not registered
- There are two **operator** instances;
 - **O1** - 'Smooth Buses', who register and provide the main service,
 - **O2** - 'Smart Buses' who provide just the journey in column #5 on behalf of 'Smooth Buses'.

Service Structure

- There is a single **Service** instance *SVI*, with two **Line** instances *Ln_1*. and *Ln_2*.
- There are seven **StopPoint** instances, all references to existing NaPTAN stops
- There are two **Route** instances *s* (*R_1*, *R_2*), each with a single **RouteSection** (*RS_1*, *RS_2*)
 - *RS_1* has four **RouteLink** instances (*RL_1*, *RL_2*, *RL_3*, *RL_4*), running '*Grub Street - Tin Pan Alley - Sinister Street - Swann's Way - Howard's End*'. Each link has two stop references (*RL_1a*, *RL1b*, etc)
 - *RS_2* has two **RouteLink** instances (*RL_5*, *RL_6*) links respectively. running '*Swann's Way - Yellow Brick Road - Howard's End*'.

Outbound Journeys

- There are two outbound **JourneyPatternSection** instances *JS_1* and *JS_2*; corresponding to the two **RouteSection** instances, with corresponding journey pattern timing links.
- The outbound service is made up of two **JourneyPattern** instances;
 - *JP_1*, section: *JS_1*, **JourneyPatternTimingLink** instances (*JL_1*{*RL_1*}, *JL_2*{*RL_2*}, *JL_3*{*RL_3*}, *JL_4*{*RL_4*}), Each journey pattern timing link has two stop usages (*JL_1a*, *JL1b*, etc)
 - *JP_2*, section: *JS_2*, **JourneyPatternTimingLink** instances *JL_5*{*RL_5*}, and *JL_6*{*RL_6*}
- There are four outbound **VehicleJourney** instances:
 - *VJ_1*, for line: *Ln_1* over *JP_1*, with a departure time of 8:02. This appears in column #1
 - *VJ_2*, also for line: *Ln_1* with a departure time of 9:02, which references *VJ_1* for its links, and so also follows *JP_1*.
 - The **Frequency** element specifies that is a Frequency Based service with a **ScheduledFrequency** of *every 15 minutes* which generates both column #2 and column #3. However The service is not a statutory Frequent Service (and the interval is greater than 10 minutes), so **FrequentService** is *false*.
 - *VJ_3*, also for line: *Ln_1* , with a departure time of 12:02, which references *VJ_1* for its links, and so also follows *JP_1*. This appears in column #4.
 - *VJ_4*, for line: *Ln_2*, over *JP_2*. with a departure time of 12:45. This appears in column #5. It has a different operator of *O2*.

Inbound Journeys

- There are two inbound **JourneyPatternSection** instances *JS_X1* and *JS_X2*; corresponding to the two **RouteSection** instances, with corresponding journey pattern timing links, specified in reverse order

- The outbound service is made up of two **JourneyPattern** instances;
 - **JP_Z1**, section: **JS_X1**, **JourneyPatternTimingLink** instances (**JL_X1 {RL_4}**, **JL_X2 {RL_3}**, **JL_X3 {RL_2}**, **JL_X4 {RL_1}**), Each journey pattern timing link has two stop usages (**JL_1a**, **JL1b**, etc)
 - **JP_2**, section: **JS_2**, **JourneyPatternTimingLink** instances **JL_5{RL_6}**, and **JL_6{RL_5}**
- There are four outbound **VehicleJourney** instances:
 - **VJ_X1**, for line: **Ln_1** ('ln1') over **JP_X1**, with a departure time of 9:00. This appears in column #1
 - **VJ_X2**, also for line: **Ln_1** ('ln1') with a departure time of 10:00, which references **VJ_X1** for its links, and so also follows **JP_X1**.
 - The **Frequency** element specifies that is a Frequency Based service with a **ScheduledFrequency** of every 15 minutes - for which the publisher generates both column #2 and column #3. The service is not a statutory Frequent Service, so **FrequentService** is *false*
 - **VJ_X3**, also for line: **Ln_1**, ('ln1') with a departure time of 13:00, which references *also* **VJ_X1** for its links, and so also follows **JP_X1**. This appears in column #4
 - **VJ_X4**, for line: **Ln_2** ('ln2'), over **JP_X2**. with a departure time of 12:45. This appears in column #5. It has a different operator of **O2**

Interchanges

Connecting Journeys

- There is an interchange at Swann's Way between **VJ_3** & **VJ_4**
- There are also three **ConnectingJourneys**

describing connections to externally defined journeys

- **EVJ_e1**, Line 'E1' which connects to feed **VJ_3** at *Grub street*
- **EVJ_e2** Line 'E2' & **EVJ_e3**, Line 'E3' which connects from **VJ_3** at *Howards End*.

Outbound Journey Interchange

- The service has an outbound **JourneyPatternInterchange** (**J1_1**) that connects the two journey patterns (**JP_1**, **JP_2**) at specific journey pattern timing link stop usages.
 - inbound: **JP_1** : **JL_3b**,
 - outbound: **JP_2**: **JL_5a**.
- The connection of the two journeys **VJ_3** & **VJ_4** at *Swann's Way* is modelled by two an outbound **VehicleJourneyInterchange** instances, one for each of the two vehicle journeys that connect; both reference the journey pattern interchange **J1_1**.

- interchange: **VJI_3**: for journey **VJ_3** has inbound: **VJ_3**,: outbound: **VJ_4**,
 - interchange **VJI_4**: for journey **VJ_4** has inbound: **VJ_3**,: outbound: **VJ_4**).
- There are also three **VehicleJourneyInterchanges** to describe the externally described connecting journey feeder & distributor interchanges with journey with journey **VJ_3**.
 - interchange: **VJI_E1** : for journey **VJ_3** has inbound: **EVJ_e1**,: outbound: **VJ_3**,
 - interchange: **VJI_E2** : for journey **VJ_3** has inbound: **VJ_3**,: outbound: **EVJ_e2**,
 - interchange: **VJI_E3** : for journey **VJ_3** has inbound: **VJ_3**,: outbound: **EVJ_e3**,

Inbound Journey Interchange

- The service has an inbound **JourneyPatternInterchange** (**JI_X1**) that connects the two journey patterns (**JP_X2**, **JP_X1**) at specific journey pattern timing link stop usages.
 - inbound: **JP_X2** : **JL_X6b**,
 - outbound: **JP_X1** : **JL_X2a**.
- The connection of the two journeys **VJ_3** & **VJ_4** is modelled by two inbound **VehicleJourneyInterchange** instances , one for each of the two vehicle journeys that connect; both reference the journey pattern interchange **JI_X1**.
 - interchange: **VJI_X3**: for journey **VJ_X3** has inbound: **VJ_X3**,: outbound: **VJ_X4**,
 - interchange **VJI_X4**: for journey **VJ_X4** has inbound: **VJ_X3**,: outbound: **VJ_X4**).

The Operating Days

- The **Service operating period** for **SV_1** starts on **02/01 2002** for all vehicle journeys and continues indefinitely.
- The **OperatingProfile** for the Service **SV_1** states values that apply to all journeys unless overridden on a journey pattern or individual vehicle journey.
 - The **RegularDayType** of **SV_1** specifies journeys of the service run Monday to Friday every week of the year.
 - The **BankHolidayOperation / DaysOfNonOperation** of **SV_1** states journeys of the service do not run Christmas (i.e. Christmas Day & Boxing Day) New Year's Day, and Good Friday, but that they do run on all Bank Holiday Mondays.
 - The **SpecialDaysOperation / DaysOfOperation** of **SV_1** states that journeys of the service are will run on **02/06/2004** regardless.
 - The **SpecialDaysOperation / DaysOfNonOperation** states that all services will not run between **09/12/2004** and **15/12/2004** regardless, and also will not run on **01/06/2004**.

- There is an override instance of the **OperatingProfile** for journey pattern **JP_1**, which applies to all its **VehicleJourney** instances:- **VJ1_2**, **VJ_2**, & **VJ_3**,
 - The **SpecialDaysOperation / DaysOfOperation** of **JP_1** states that journeys of the service are will run on *12/16/2004* regardless.
 - The **SpecialDaysOperation / DaysOfNonOperation** states that all services will not run between *01/11/2004* and *1111/2004* regardless.
- There are override instances of the **OperatingProfile** for the vehicle journeys **VJ_1**, **VJ_2**, & **VJ_4**, but not for **VJ_3**, which has the same values as the general service defaults for **SV_1**.
 - **VJ1: OperatingProfile**
 - The **RegularDayType** states that **VJ_1** runs **Monday** to **Friday** in the term time of serviced organisation **SO_2** (*Dotheboy's*).
 - The **BankHolidayOperation / DaysOfNonOperation** states that **VJ_1** does not run on **Mayday**, differing from the general defaults. **VJ_1** still does not run on **Christmas** (i.e. **ChristmasDay** & **BoxingDay**) **NewYearsDay**, and **GoodFriday**.
 - **VJ2: OperatingProfile**
 - The **RegularDayType** of states that **VJ_2** runs **Monday** to **Saturday**.
 - The **BankHolidayOperation / DaysOfOperation** states that **VJ_2** does run on **BoxingDay** and **NewYearsDay**.
 - **VJ3: OperatingProfile**
 - As for Service.
 - **VJ4: OperatingProfile**
 - The **RegularDayType** states that **VJ_4** runs on **Wednesdays**.
 - The **PeriodicDayType** states that **VJ_4** furthermore only runs on the first and third weeks of the month.
 - The **SpecialDaysOperation / DaysOfOperation** **VJ_4** states that **VJ_4** does run on *5/08/2004*.
 - The **SpecialDaysOperation / DaysOfNonOperation** states that **VJ_4** is does not run on *05/02/2004*, and between *04/08/2004* and *09/08/2004* and also not between *05/08/2005* to *10/08/2005*.
- The **VehicleJourneyInterchange** instances **VI_3** & **VI_4** which connect **VJ_3** and **VJ_4** have a validity period from *02/01/2004* to *17/10/2004*.

The Service Organisation Days

- **VehicleJourney VJ1** states on its **OperatingProfile** that the
 - The service only runs in the term time of **ServicedOrganisation SO2_1** (*Dotheboy's*).
- There are two **ServicedOrganisation** Definitions.
 - A local authority '*Bleakshire LEA*' , **SO_1**
 - *Dotheboys*, **SO_2** which has *Bleakshire LEA* as its parent
- The **ServicedOrganisation** definition for '*Bleakshire LEA*', **SO_1** defines the terms of the school year using working days.

- *Michaelmas Term* **StartDate** 2004-09-01, **EndDate** 2004-12-23 ,
Exception 2004-11-11.
- *Easter Term* **StartDate** 2005-01-01, **EndDate** 2005-04-30.
- *Summer Term* **StartDate** 2005-04-02, **EndDate** 2005-07-2.
- The **ServicedOrganisation** definition for *Dotheboy's* overrides the summer term working days.
 - *Summer Term* **StartDate** 2005-04-01 **EndDate** 2005-07-30

Notes

This example was refined for in the 2.4 Enhanced publisher to not be a Frequent Service

Page last updated: 2013/04/13

Examples 2.4 - Large route

Summary

Registration for a set of long routes

- Many stops, requiring page breaking vertically to publish.
- Many vehicle journeys, requiring page breaking horizontally to publish.
- Some operational date rules.
- Sharing of Sections & Journeys.
- Timing links with zero values.
- Multiple Frequency based services, breaking horizontally to publish.

Published as: [PDF](#)

Route Map

Not shown - see map pdf

Timetable

Not shown - see pdf

The XML Representation

[XML Document](#)

- There is a single **Service** instance *SVI*, with one **Line** - 'A1'.
- The service has more stops than will fit on a single page and more journeys than will fit on a single page

Service Registration

- The service is **Registered**.
- There is a single **Operator**.

Service Structure

- There is a single **Service** instance *SVI*, with one **Line** - 'A1'.
- There are 31 **StopPoint** instances, more than will fit vertically on a single page.

- There are two **RouteSection** *rs_1* and *rs_2* with **RouteLink** instances connecting the stops.
- There are two **JourneyPatternSection** *jp_s1* and *jp_s2* made up of 5 and 25 **JourneyPatternTimingLink** instances respectively.
- There is a single **JourneyPattern** instance *JP_1*, with two sections: *JS_1* and *js_2*.
- There are 27 **VehicleJourney** instances, all for *Ln_1*, more than will fit horizontally on a single page. The first journey *vj_1* is based on *JPI*, all the others are based on *VJ_1*, with different times. The journeys fall into two groups.
 - *Vj_1-20* which are single services and demonstrate page overflow for simple columns.
 - *Vj_2127* which are marked as frequent service and demonstrate page overflow for frequency based service column groups.

Page last updated: 2013/04/13

Examples 2.4 - Linear route Example

Summary

Registration for a single route run by a single operator. There are two vehicle journeys with the same timings.

- Linear route.
- Local bus stop definition.
- Route Track Map with Tracks. (Enhanced Publisher Only).
- Tracks (see below), including instructions and Mapping System References.
- Local data coordinates for a stop.
- Inbound and outbound journeys and journey patterns on the same route.
- *Frequent Service* Frequency based service, generating a frequency column specified with an interval and a minimum and maximum frequency.
- Operator details including parent (TXC v2.4).

TransXChange XML Document	Particulars and Matrix output PDF	Route Map Output
linear.xml (One direction only)	Particulars and Matrix PDF	Route Map PDF Route Map PDF (no background)

Route Map



Timetable

Outbound			
Line L1	Journeys		
	#1		#2
Netherley Road	07:00	And then every 7 minutes until 18:30	19:00

<i>Ashby Road</i>	07:20		19:20
<i>ASDA, opposite</i>	07:40		19:40
<i>Leicester Road, SE Bound</i>	07:50		19:50
<i>Golf Club, outside</i>	08:00		20:00

Inbound			
Line L1	Journeys		
	#3		#4
<i>Golf Club, outside</i>	07:30	And then every 7 minutes until 19:30	20:00
<i>Leicester Road, SE Bound</i>	07:40		20:10
<i>ASDA, opposite</i>	07:50		20:20
<i>Ashby Road, before</i>	08:10		20:30
<i>Netherley Road</i>	08:30		21:00

The XML Representation

Service Registration

- The **Registration** is for a single **Operator**

Service Structure

- There is a single **Service** instance *SVI* , with one **Line** - '*L1*' .
- There are five **StopPoint** instances.
 - One stop is defined locally (*Netherley Road*) ., the others are all references to existing NaPTAN stops.
- There is a single **RouteSection** *rs_1* with four **RouteLink** instances *rl_1-rl_4* connecting the stops.
- There is a single **Route** *r_1* containing the route section *rs_1*.

Outbound

- There is a single **JourneyPatternSection** *js_1*, made up of four **JourneyPatternTimingLink** instances, *jptl_1-jptl_4*, each referencing the corresponding route link.
 - *jptl_1* references *rl_1*
 - *jptl_2* references *rl_2*
 - *jptl_3* references *rl_3*
 - *jptl_4* references *rl_4*
- There is a single outbound **JourneyPattern**, *jp_1*, that follows route *r_1*.
- There are two outbound **VehicleJourney** instances, both based on **JourneyPattern** *jp_1* :
 - The first, *vj_1* , follows *jp_1* without any changes to the timings on the **JourneyPatternTimingLink** instances.
 - It has a **DepartureTime** of 7.00.
 - The **Frequency** element specified it is a frequency based journey with a **ScheduledFrequency** of every 8 minutes, and an **EndTime** of 18:30. The **Frequency** entry causes the publisher to create an additional column. In addition an optional minimum and maximum frequency interval are specified
 - The second, *vj_2*, references *vj_1* for all its links, with a **DepartureTime** of 19:00.

Inbound

- There is a single inbound **JourneyPatternSection** *js_r1*, made up of four **JourneyPatternTimingLink** instances, referencing a corresponding route link.
 - *jptl_r1* references *rl_4*
 - *jptl_r2* references *rl_3*
 - *jptl_r3* references *rl_2*
 - *jptl_r4* references *rl_1*
- There is a single inbound **JourneyPattern**, *jp_r1*, that follows route *r_1*.
- There are two inbound **VehicleJourney** instances, both based on **JourneyPattern** *jp_r1* :
 - The first, *vj_r1* , follows *jp_r1* without any changes to the timings on the **JourneyPatternTimingLink** instances. It has a **DepartureTime** of 7.30.
 - It has a **Frequency** of *every 20 minutes*, and an **EndTime** of 19:30.
 - The second, *vj_r2* references *vj_r1* for all its links, with a **DepartureTime** of 20:00.

The Operating Days

- The service **OperatingProfile** says it runs Monday to Friday, every day of the year.

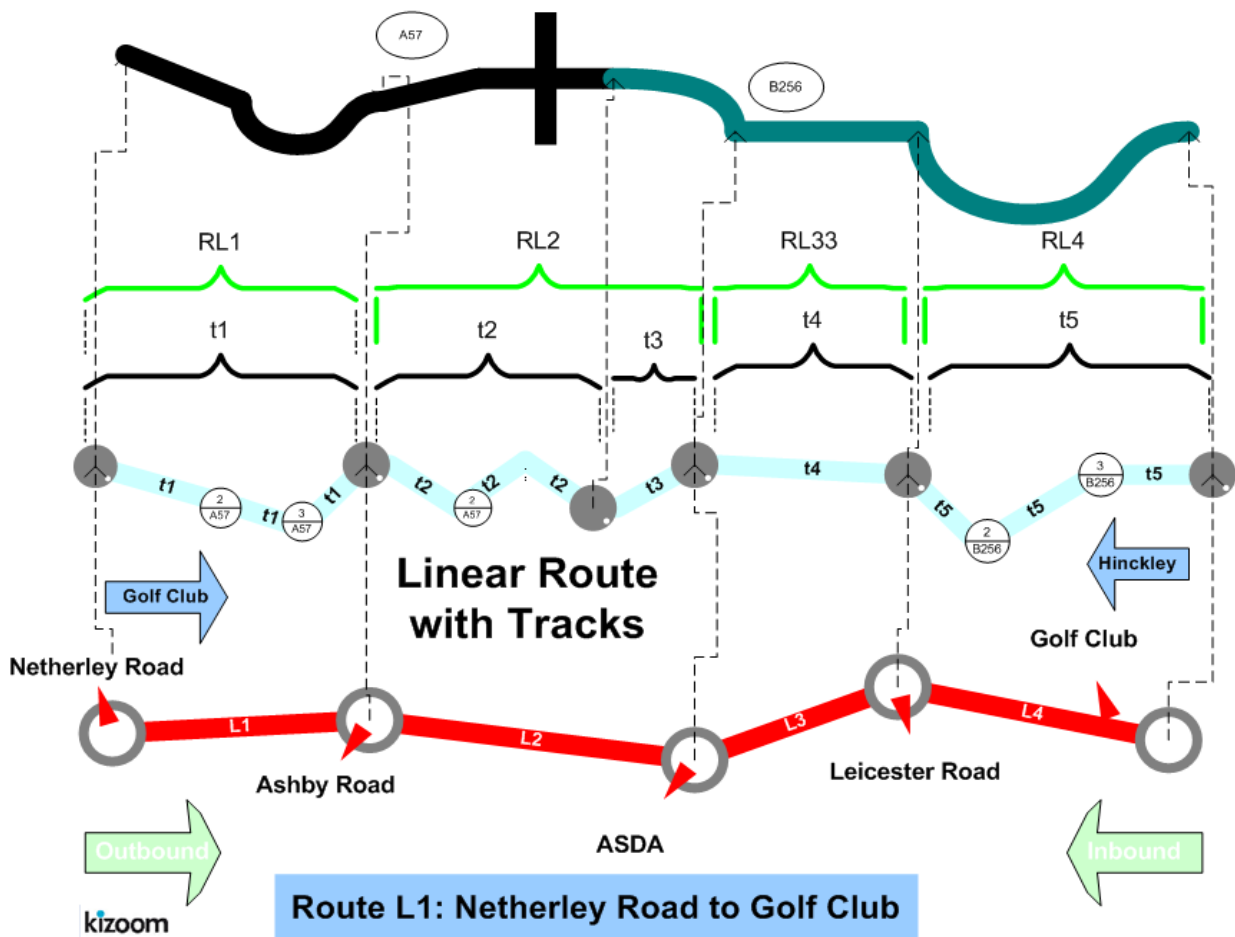
Use of Tracks

The example includes a description of the physical route as an ordered sequence of tracks, each containing a polyline of geospatial points.

- Each **RouteLink** has a single **Track**, except for the link between 'Ashby Road' and 'ASDA', which has two **Track** instances (*t2* & *t3*), because its road goes over two different roads; the A57 and the B256.
- Some of the **Tracks** include **MappingSystem** references to TOID instances.

The following diagram shows how the route projects onto the map representation of the road system, using the NaPTAN stop points as projection points between levels of discourse.

- A schematic map of the road network appears at the top.
- The stylised route map appears at the bottom.
- In between are shown the **route links** and track link, with the points



Notes

This example was refined for in the 2.4 Enhanced publisher to be a Frequent Service i.e. <10 minutes and to have a minimum and maximum.

In the enhanced publisher will group the journeys in a single bed Monday to Sunday, rather than as separate beds Monday to Friday, Saturday and Sunday.

Page last updated: 2013/04/13

Examples 2.4 - Lollipop route

Summary

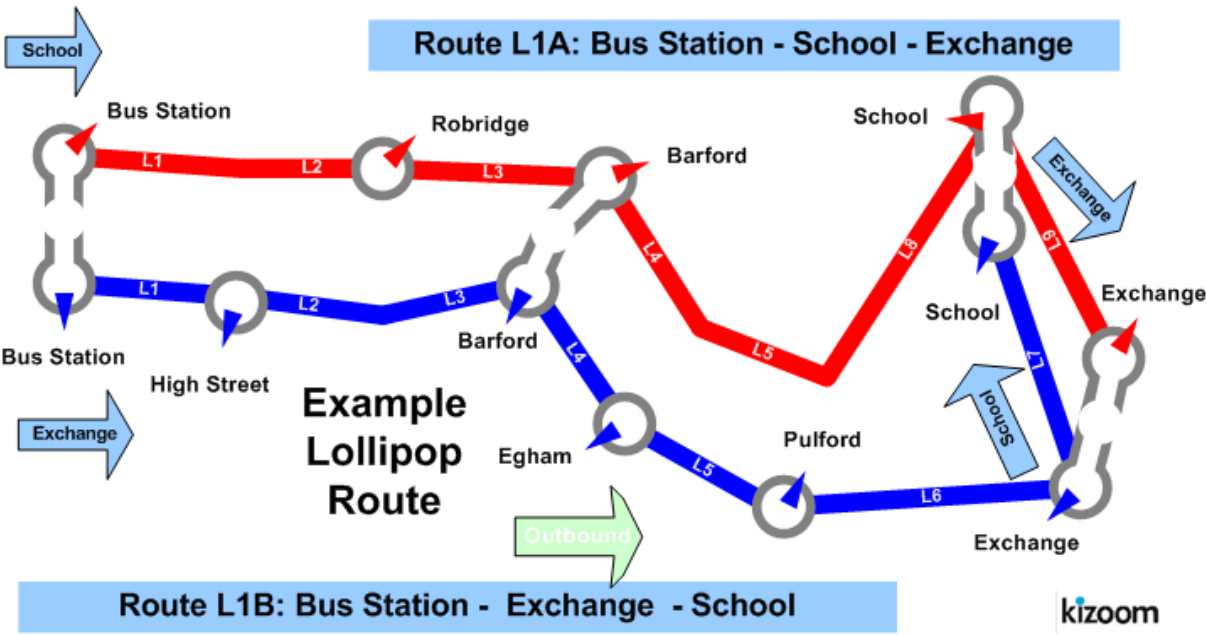
Same stops covered by two different routes in different directions in a partial lollipop. See also **Hail and Ride section** for a full lollipop topology.

- Lollipop route, with two journeys proceeding round the loop in opposite directions (In this example neither route completes the full lollipop course)
- Two services run by two different operators, sharing a line label.
- Two different physical routes with the same line label.
- Timing status on stop usage other than 'Principle Timing Point (PTP)'.
- Additional operating days (see below) for regular and bank holiday operation.
- Alternative Bank Holidays only operation
- Express stopping pattern for some journeys.
- Use of Stop Sequence Numbers (see below) to control row order in matrix..
- Use of Dynamic Destination headings.
- Reuse of shared sections (see below)
- Reuse of VehicleJourney Link.
- Layover Point.
- General Schema.
- Minimum Layover Duration (TXC v2.4)
- Partial Frequent Services (TXC v2.4).

Published as PDF:

- [PDF general](#): With *publish options* `lollipop.xml -full` TransXChange [publisher](#) option (includes all stops).
- [PDF vosa](#): With *publish options* `lollipop.xml -vosa` TransXChange [publisher](#) option (includes only stop usages with a timing status of 'PTP').

Route Map



Timetable

Bus Station To Exchange

Holidays Only #1 #2 #3 #4 #5 #6 ACO ACO ACO RED ACO ACO 1B 1A 1B 1B
1B 1B

Column							
Operator							
Line							
Bus Station	15:55	16:15	16:35	16:40	16:55		16:35
High St				16:46			
Robridge		16:26					
Barford	16:09	16:29	16:49	16:54	17:09		16:49
Egham	16:12		16:52	16:57	17:12		16:52
Pulford	16:15		16:55	17:00	17:15		16:55
Exchange	16:32	16:52	17:12		17:32		17:22
School		16:53		17:16			
Exchange				17:17			

The XML Representation

[XML Document](#)

Service Registration

- The service is not registered
- There are two **operator** instances, '*ACO*'. and '*RED*'

Service Structure

The detailed map below shows the sections and stop sequences used.

- There are two **Service** instances whose **Lines** are labelled to appear to the public as the same:
 - **S1**, which has two **Line** instances; **L1**, labelled '*L1A*', and **L2**, labelled '*L1B*'. **S1** is run by **Operator** '*ACO*'.
 - **S2** which has **Line** **L3** also labelled '*L1B*'. **S2** is run by **Operator** '*RED*'.
- There are three **RouteSection** instances;
 - **RS1**: '*Bus Station*' to '*Pulford*', containing **RouteLink** instances **RL1** to **RL5**.
 - **RS2**: '*Pulford*' to '*School*' via '*Exchange*', containing **RouteLink** instances **RL6** and **RL7**.
 - **RS3**: '*Pulford*' to '*Exchange*' via '*School*', containing **RouteLink** instances **RL8** and **RL9**.
- There are two **Route** instances:
 - **R1**: '*Bus Station to School (via Exchange)*', containing **RS1** and **RS2**.
 - **R2**: '*Bus Station to Exchange (via School)*', containing **RS1** and **RS3**.
- There are three corresponding **JourneyPatternSection** instances, **JS1**, **JS2** and **JS3**, one for each route section.
 - Egham is not a 'Principle Timing Point' ('*PTP*'), so for the call at Egham the **JourneyPatternTiminkLink** has an override **TimingStatus** of 'Timing Information Point' ('*TIP*').
- **S1** has two **JourneyPattern** instances:
 - **JP1**: Following **Route** **R1**, with sections **JS1** and **JS2**, passing '*Pulford*' without stopping.
 - **JP2**: Following **Route** **R2**, with sections **JS1** and **JS3**.
- **S2** has one **JourneyPattern** instance:
 - **JP3**: Also following **Route** **R2**, with sections **JS1** and **JS3**, passing '*Pulford*' without stopping. It contains two named layover points *lay_1* and *lay_2*.
- There are six **VehicleJourney** instances one for each column.
 - There are five **VehicleJourney** instances **Vj1-Vj5** that run on on regular days - see columns #1- #5,

- There is one **VehicleJourney** instance **Vj6** that run on Bank Holiday only service see column #6.
- S1 has five **VehicleJourney** instances:
 - **VJ1**: following **JourneyPattern JP1** over **Route R1**, '*Bus Station to School (via Exchange)*', as **Line 'L1B'**, passing '*High Street*' and '*Robridge*', without stopping. Short working of the second section, i.e. termination at Exchange, is indicated by a dead run.
 - **VJ2**: also following **JourneyPattern JP1** over **Route R1**, '*Bus Station to School (via Exchange)*' but as **Line 'L1A'**, passing '*High Street*', '*Egham*' and '*Pulford*' without stopping.
 - **VJ3**: reuses the timing links of **VJ1**, and so also follows **JourneyPattern JP1** over **Route R1**, '*Bus Station to School (via Exchange)*'.
 - **VJ5**: reuses the timing links of **VJ1**, and so also follows **JourneyPattern JP1** over **Route R1**, '*Bus Station to School (via Exchange)*'.
 - **VJ5**: reuses the timing links of **VJ1**, and so also follows **JourneyPattern JP1** over **Route R1**, '*Bus Station to School (via Exchange)*'. Runs only on bank holidays
- S2 has one **VehicleJourney** instance:
 - **VJ4**: following **JourneyPattern JP3** over **Route R2**, '*Bus Station to Exchange (via School)*', as **Line L3**, which has the same label of '*L1B*' as **SI/L2**, although it actually goes round the last two stops in a different sequence from the other journeys with the same label, and so needs a different underlying **Route**. **VJ4** passes '*Robridge*', without stopping.

Operating Days

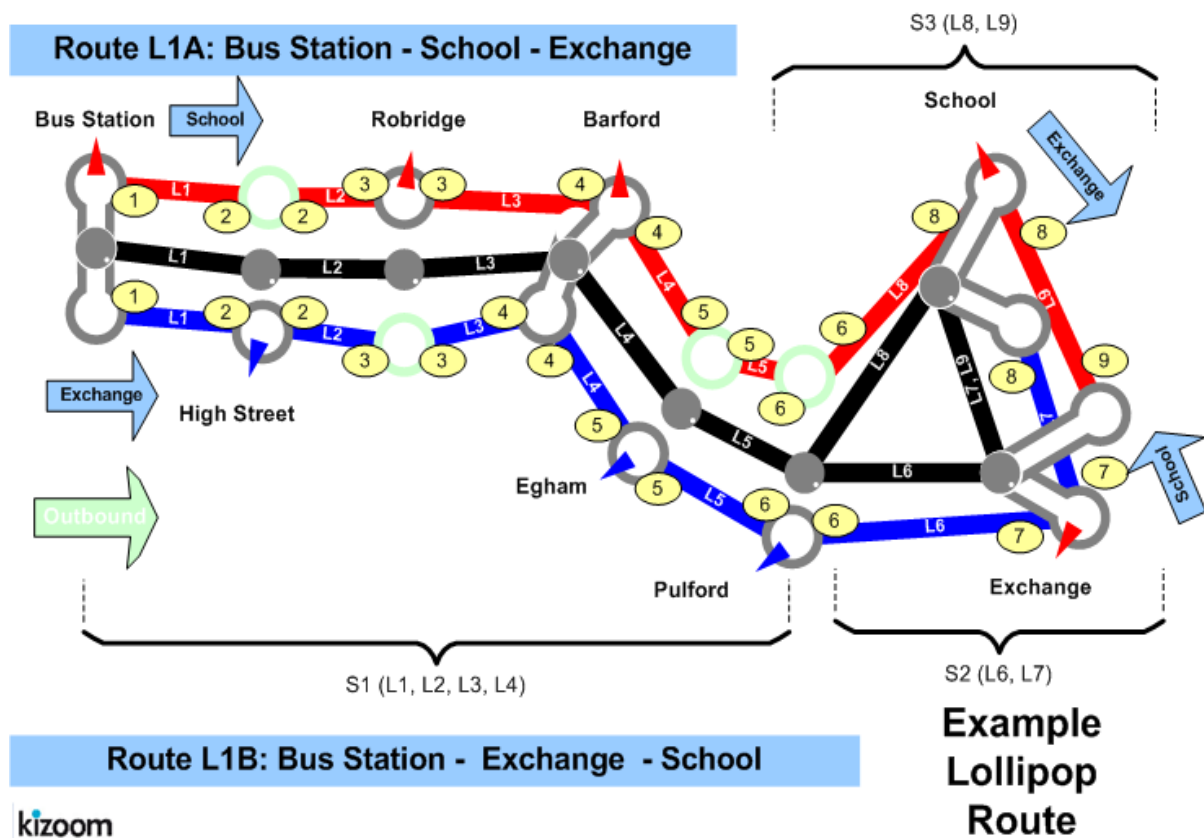
- The **Service OperatingPeriod** for **SV_1** starts on *01/01/2004* for all vehicle journeys, and continues until *13/06/2005*
- The **OperatingProfile** for the **Service SV_1** states values that apply to all journeys unless overridden on a journey pattern or individual vehicle journey.
 - The **RegularDayType DaysOfWeek** specifies journeys of the service by default runs **Monday** to **Saturday** throughout the year
 - The **BankHolidayOperation / DaysOfOperation** states journeys of the service run on **Jan2ndScotland**, **MayDay**, **EasterMonday**, **SpringBank**, **StAndrewsDay**, **AugustBankHolidayScotland**, **ChristmasDayHoliday**, **BoxingDayHoliday** and **ChristmasEve**.
 - The **BankHolidayOperation / DaysOfNonOperation** states journeys of the service do not run on **ChristmasDay**, **BoxingDay**, **GoodFriday**, **NewYearsDay**, **NewYearsDayHoliday** and **NewYearsEve**.
- The **Vehicle Journeys VJ1** and **VJ5** have a different set of day type specified.
 - The **RegularDayType DaysOfWeek** specifies journeys of the service run **Monday**, **Wednesday**, **Friday**, **Sunday** throughout the year.
- The **Service operating** period for **SV_2** starts on *01/01/2004* for all vehicle journeys, and continues until *13/06/2005*.

- The **OperatingProfile** for the **Service SV_2** states values that apply to all journeys unless overridden on a journey pattern or individual vehicle journey.
 - The **RegularDayType DaysOfWeek** specifies journeys of the service by default runs **Monday** to **Sunday** throughout the year.
 - The **BankHolidayOperation / DaysOfNonOperation** states journeys of the service do not run on **Christmas**, i.e. **ChristmasDay**, and **Boxing day**.

Use of Sections & Stop Sequence Numbers

Sections are used to reuse links between journeys.

Stop sequence numbers are used to coerce a specific ordering of the stops within a matrix timetable when published. The following diagram shows the journey pattern sections annotated with stop section numbers.



Examples 2.4 - Merge Frequency journeys

Summary

Registration for a route that has been coded as individual journeys but is to be shown as a Frequent Service.

- Many stops.
- Many vehicle journeys.
- Merging of multiple journeys by the publisher (2007 enhanced version only).
- Some operational date rules.
- Operational data for Journey pattern for vehicle type.
- Operational data for vehicle journey block.
- Vehicle Journey private code.
- Non PTP service.

Published as: (i) [PDF](#) (merged) or (ii) [PDF](#) (unmerged).

Route Map

Timetable

Service operates from 22/09/2004 until further notice.

Suborn - St James's Church								
Outbound, Monday to Friday								
	L1	L1	L1		L1	L1		L1
Suborn, Suborn Bus Station	07:00	07:30	08:00	Frequent service at least every 10 mins until	09:10	09:10	Frequent service at least every 10 mins until	10:45
Suborn, Garden Village Shops	07:20	07:50	08:20		09:30	09:30		11:05
Suborn, Plough	07:40	08:10	08:40		09:50	09:50		11:25
Suborn, Red Lion	07:50	08:20	08:50		10:00	10:00		11:35
Suborn, Golden Lion	08:00	08:30	09:00		10:10	10:10		11:45
Suborn, Paradise	08:10	08:40	09:10		10:20	10:20		11:55
Suborn, Kensal Green	08:30	09:00	09:30		10:40	10:40		12:15
Suborn, Much Binding	08:50	09:20	09:50		11:00	11:00		12:35
Suborn, Village	09:00	09:30	10:00		11:10	11:10		12:45

Hall							
Suborn, War Memorial	09:10	09:40	10:10		11:20	11:20	12:55
Suborn, The Cricketers	09:20	09:50	10:20		11:30	11:30	13:05
Suborn, Thirkhill Drive	09:30	10:00	10:30		11:40	11:40	13:15
Suborn, Woolworths	09:40	10:10	10:40		11:50	11:50	13:25
Suborn, Police Station	09:50	10:20	10:50		12:00	12:00	13:35
Suborn, Mattmans Garage	10:00	10:30	11:00		12:10	12:10	13:45
Grassy Knowle, Nelson Mandela Way	10:10	10:40	11:10		12:20	12:20	13:55
Grassy Knowle, Robinsons Store	10:20	10:50	11:20		12:30	12:30	14:05
Grassy Knowle, Post Office	10:40	11:10	11:40		12:50	12:50	14:25
Grassy Knowle, High Street	10:50	11:20	11:50		13:00	13:00	14:35
Grassy Knowle, Badgett	11:00	11:30	12:00		13:10	13:10	14:45
Grassy Knowle, Gropers Corner	11:10	11:40	12:10		13:20	13:20	14:55
Grassy Knowle, Hughes Hall	11:20	11:50	12:20		13:30	13:30	15:05
Grassy Knowle, Snow Hill	11:30	12:00	12:30		13:40	13:40	15:15
Grassy Knowle, The Jolly Roger	11:40	12:10	12:40		13:50	13:50	15:25
Grassy Knowle, Gibbards Cross	11:50	12:20	12:50		14:00	14:00	15:35
Grassy Knowle, White Water	12:00	12:30	13:00		14:10	14:10	15:45
Grassy Knowle, Key Holme	12:10	12:40	13:10		14:20	14:20	15:55
Grassy Knowle, Grassy Knowle	12:20	12:50	13:20		14:30	14:30	16:05

Grassy Knowle, Hughes Library	12:30	13:00	13:30		14:40	14:40		16:15
Grassy Knowle, Dixon's	12:40	13:10	13:40		14:50	14:50		16:25
Grassy Knowle, St James's Church	12:50	13:20	13:50		15:00	15:00		16:35

The XML Representation

[XML Document](#)

- There is a single **Service** instance *SVI*, with one **Line** - '*LI*'.
- The service has multiple individual services and more journeys than will fit on a single page

Service Registration

- The service is **Registered**.
- There is a single **Operator**.

Service Structure

- There is a single Service instance *SVI*, with one **Line** - '*LI*'.
- There are 31 **StopPoint** instances, more than will fit vertically on a single page. At least one is a TIP, i.e. non PTP
- There are two **RouteSection** *rs_1* and *rs_2* with **RouteLink** instances connecting the stops.
- There are two **JourneyPatternSection** *jp_s1* and *jp_s2* made up of five and 25 **JourneyPatternTimingLink** instances respectively.
- There is a single **JourneyPattern** instance *JP_1*, with two sections: *JS_1* and *js_2*.
- There are 27 **VehicleJourney** instances, all for *Ln_1*, all based on *JP_1*
- The vehicle journeys fall into three groups:
 - *vj_1* to *Vj_2*, which are described as individual journeys since they are spaced at more than 10 minutes.
 - *vj_3* to *Vj_15*, which are marked as belonging to the first frequency group by a **Frequency** element with a statutory **ScheduledFrequency** of *every 8 minutes*, and an **EndTime** of *09:10*, and **FrequentService** *true*. If published with the *mergeFrequentJourneys* option of the enhanced publisher, the journeys will be grouped as a Frequency group of a starting

column and then a column stating the frequency as at least every 8 minutes.

- *vj_16 to Vj_27*, which are marked as belonging to the second frequency group by a **Frequency** element with a statutory **ScheduledFrequency** of *every 10 minutes*, and an **EndTime** of *10:45*, and **FrequentService** *true*. If published with the *mergeFrequentJourneys* option of the enhanced publisher, the journeys will be grouped as a **Frequency** group of a starting column with the initial set of times and then a column stating the frequency. as at least *every 10 minutes*.
- Each vehicle journey has operational data on it

Notes

N.B. The *mergeFrequentJourney* option is only available with the 2007 2.4 Enhanced publisher. For Versions prior to that the individual journeys will all be published as separate Frequency Group columns.

If any frequent services are provided as individual journeys, then all the individual journeys must be provided.

Page last updated: 2013/04/13

Examples 2.4 - Operators Example

Summary

Data exchange of operator data

- List of operators
- Data rights TXC v2.4).

TransXChange XML Document	Particulars and Matrix output PDF	Route Map Output
Operators.xml (One direction only)	Not publishable	Not publishable

The XML Representation

Service Registration

- There is no service or registration

Operators

- There is are three **operators** instance *o1* , *o2*, *o3* ,

Page last updated: 2013/04/13