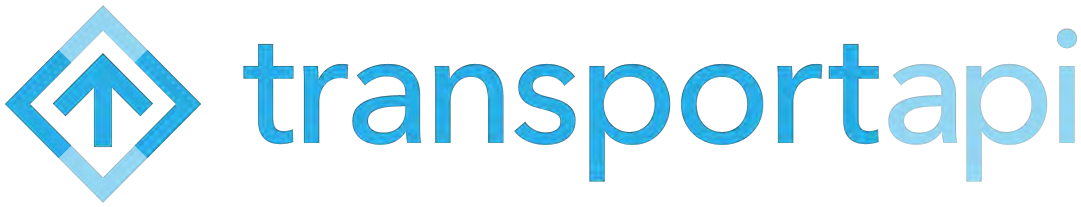


# TransportAPI

## Managed Service

G-CLOUD 15

Date: January 2026
Revision history: v1.0

transportapi

# Table of Contents

# 0 Aims

This Service Definition document sets out the details of the suite of SaaS-based products and services that collectively make up the TransportAPI service. TransportAPI is designed to operate from web application or customer information screen through to enterprise scale where it can fulfil a foundational role in customer information and product fulfilment services.

We set out below the delivery vision for TransportAPI, our operating model, our platform to deliver, and the key services available to power operator requirements. TransportAPI is based on delivery to some of the largest users of transport data in the UK and draws upon 15 years of experience.

# 1 Delivery vision

The following sections set out our high-level vision for the delivery of TransportAPI for diverse applications.

## 1.1 Single source of operating truth

The TransportAPI platform integrates and harmonises all national multimodal transport data into a single version of operating truth. This platform includes the stops/ stations, routes, schedules, fares, disruption and service tracking data, which collectively define a multimodal representation of national transport operations. TransportAPI also archives bus and rail movements to allow the reconstruction of transport operations in history.

TransportAPI is a comprehensive platform to deliver the operating truth to diverse applications including apps, screens, analytics and retail systems. The operating truth consists of key national datasets integrated with customer source data as validated by TransportAPI, and archived for the record.

The delivery of this operating truth is based on TransportAPI's SaaS-based, cloud native API offering that can be integrated by all components in the customers's ecosystem. TransportAPI enables customers to minimise friction and failed integrations across the product, retail and customer information landscape by ensuring the single version of operating truth is fully maintained and updated according to national standards.

## 1.2 Native API offer to power your applications

There is a natural operational division between the front end UX and the back-end data feeds. Once the user experience has been designed the backend data needed to fulfil the UX can be sourced and infrastructure can be put in place capable of delivering the performance and scale required. TransportAPI have experience in working in multidisciplinary teams to deliver our specialist expertise at the backend, and our role is not confused by any desire or necessity to use our own UX assets.

TranportAPI offers eight managed services covering the whole of the national public transport system, with dozens of endpoints giving access to all the data feeds that are required to implement our customer's objectives. We are a native API offering that is integrated into some of the most significant public transport sites in the country.

## 1.3 Monitoring a volatile data landscape

Our mission in the orchestration and delivery of the backend data feeds is that we should always notice problems with the data before the customer does. This is challenging because of the hundreds of thousands of stops, routes and services across the country. However, effective data quality monitoring can ensure the source data is as correct as possible, and that all issues with the data are reported upstream to the source for definitive resolution.

Over time, when this approach is consistently applied, it is possible to drive data quality to a place that matches the execution ambition of the business. Over a typical annual cycle our data quality work has reduced the number of errors in the core datasets for the delivery of customer information by around 25%. This helps to deliver the aspiration that our customers have to ensure that every journey is supported and informed by the best quality data despite the volatile data landscape in transport.

## 1.4 Working as a single team

The inherent nature of data feed aggregation is that data is acquired from a very diverse range of sources, that often use different practices and tools. It is essential to the achievement of a very high level of execution quality that these parties cooperate effectively and that they are not defensive about their delivery. TransportAPI have found that achieving this objective requires orchestration of suppliers into a single team that is transparent internally about data quality.

This orchestration may require working practice modifications to drive transparency or delivery quality, and we would expect to work with customers to achieve this. This process takes time and leadership, and our practice is to follow the lead of our customer when setting the tempo and ambition of this orchestration.

After a decade of leading this orchestration of suppliers in various settings we have the experience to develop highly effective single team working. On the rare occasions that a problem emerges e.g. a corrupted upstream dataset, we now have the communication and support mechanisms in place to fix the problem and recover quickly.

## 1.5 Target-driven delivery using agile processes

Our experience in delivering large scale digital projects is that agile processes offer the most effective delivery framework. However, the challenge of agile delivery is to ensure that outputs are available on the date that the UX needs to consume them. We have a track record of successfully delivering through comprehensive design, active backlog management and early access to prerequisites.

It is also necessary to plan the implementation by building in sympathy with the target architecture rather than by slavishly following a conceptual workpackage subdivision, which may lead to implementation in the wrong sequence or repetition of work. Our work with customers implementing their vision always reflects this agile-driven approach to delivery.

## 1.6 Continuing value enhancement

Data-driven platforms are constantly evolving due to changing inputs, constituent service evolution and standards activity. In this kind of environment there cannot ever be one delivery with periodic updates…there must be continuous delivery and ongoing value enhancement. TransportAPI bundles all general upgrades to the API into the pricing of our managed services to ensure that obsolescence is avoided. We do not engage in managed obsolescence to drive captive upgrades.

TransportAPI has found that value enhancements fall into two broad categories. Firstly, the management of change in the data delivery environment, for example, ranging from server whitelisting to data standard updates. This change is always included in our managed services: that is the value they are designed to bring.

The second type of value enhancement is the functional enrichment of the delivery. TransportAPI's model is to invest in enrichment of our platform to meet market expectations. By charging for continuously upgraded managed services through an opex model we can also deliver continuous service enhancements over the short, medium and long term. We retain IPR in the enhancements and the upgrades are added to our asset base. This model incentivises us to invest in long-term relationships as opex funding is not front loaded, though there can be some specific setup activity requiring capex funding at the start of a contract.

# 2 Operating model

This section of the Service Definition document focusses on the TransportAPI operating model, which is suitable for integration into a wide variety of customer implementation settings.

## 2.1 Operating principles

TransportAPI's key operating principles include the relationships, managed services and standards that underpin our operations.

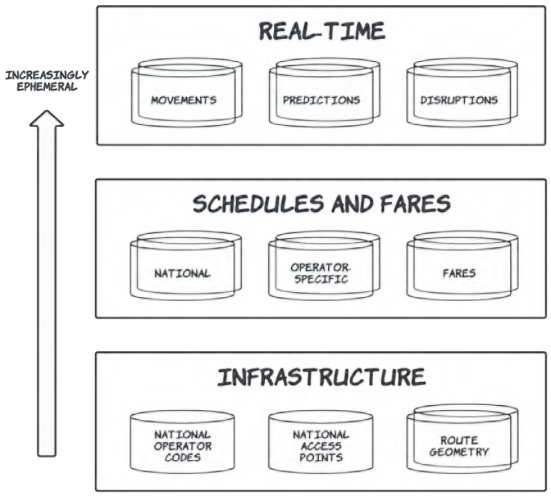
### 2.1.1 Relationships

TransportAPI has working relationships with all the key bodies in the transport data ecosystem including Department for Transport (DfT), Traveline, National Rail, Bus Open Data Service, Ordnance Survey, OpenStreetMap and Transport for London. We attend the Passenger Transport Information Committee (PTIC) and we are full members of the Real Time Information Group (RTIG), and we contribute to their standardisation activities as a working group member.

Our core role is as an aggregator of stops, routes, journey planning, bus schedule/ live information and disruptions to compile the single version of operating truth. These relationships divide into two:

- Open data sourced from Traveline, Bus Open Data Service (BODS), Transport for London (TfL) and Dept. for Transport (DfT) where the data is released scalably on an Open Government Licence (OGL) where TransportAPI undertake validation, harmonisation and cleansing;
- Commercial data sourced from Silverrail, Ticketer and CycleStreets where the data is commercially licensed and TransportAPI have integrated these sources into the platform.

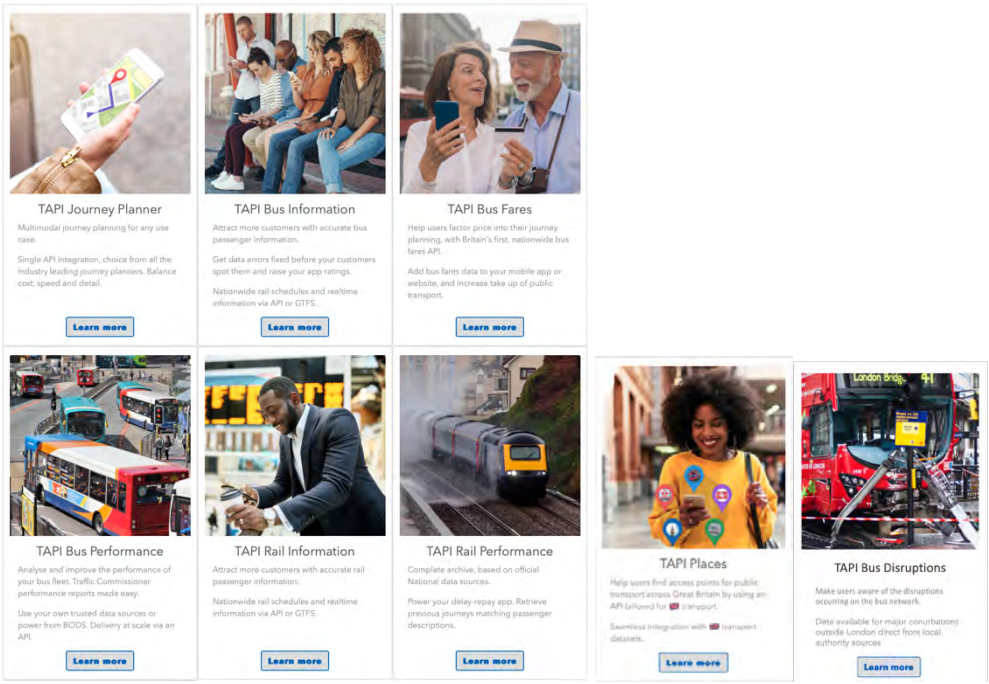
Our aggregation spans a wide range of sources, which become increasingly ephemeral as you move from infrastructure to planning and finally live operations with post-operation archiving.



### 2.1.2 Managed service focus

TransportAPI operates a set of eight national managed services covering all modes of public transport (https://www.transportapi.com/managed-services/):

- Journey planner and Places
- Bus information, Bus fares, Bus performance and Bus Disruptions
- Rail information, Rail performance




Each managed service consists of set of GET (and some POST) endpoints delivering JSON formatted data, with extended features designed to meet the needs of specific market segments. There is a full integration across all the managed services and endpoints covering location referencing, date and time formats, authentication and operator details. Hence the journey planning response is enriched by full route geometries and hyperlinks to the included journey leg schedules, and live departures and disruptions is included in our Bus Disruptions managed service.

TransportAPI delivers full intellectual property rights (IPRs) in all of the content we deliver, and so there are no downstream licensing issues to consider like mapping costs or re-use rights. The TransportAPI managed services are documented in the industry standard OpenAPI format and we provide usage examples with code and an interactive API console at http://docs.transportapi.com.

### 2.1.3 Standards

A key component of TransportAPI's commercial model is our adherence to, and investment in, standards in the journey planning, bus information and disruptions domain. We believe the standards discussed below are especially important to the delivery of our customer's objectives.

Schedules are the foundation of journey planning, and TransportAPI supports the following DfT TransXchange standards across its managed services: 2.1, 2.4, 2.5 and PTI profile 1.1.

TransportAPI both consumes and produces SIRI formatted data across a range of managed services and we have engineered to full compliance with EN/TS 15531:2011-2016. Our experience with each of the parts of the standard is as follows:

- SIRI-VM: we consume this data directly from some operators/ authorities in its streaming form and we access the data through the BODS polling interface. We publish billions of live bus-on-a-map updates per year from Ticketer into the First Bus app using websockets to deliver low latency service updates from SIRI-VM;
- SIRI-SM: we consume this data directly from Ticketer and uTrack across the country and we use it to publish departure boards in customer apps at scale;
- SIRI-SX: we formerly operated the Transport for the North Open Data Hub where we delivered SIRI-SX as a filtered polling service so local transport authorities (LTAs) could integrate disruptions into their journey planner solutions. We now offer this service from the BODS SIRI-SX data in our new Bus Disruptions managed service , which is used by the South Yorkshire Mayoral Combined Authority (SYMCA).

TransportAPI also aggregates rail data from the National Rail Darwin DWS/ Pushport services and the Network Rail ITPS, (V)STP and CORPUS services. TransportAPI powers the schedule and live data in all of the First Group train operating company apps as well as the LNER colleague app and the Heathrow Express website.

## 2.2 Development process

We set out below how TransportAPI delivers on product enhancements to specification, to budget and to time given a rich and comprehensive set of requirements. TransportAPI is also used by customer-appointed developers working with our platform who we support with their design, integration and deployment work.

### 2.2.1 Kanban

We have successfully used a Kanban approach in large scale agile development to ensure that development work is optimised for delivery in the forecasted time. Key services delivered in this way include the Open Data Hub Disruption Messaging service for Transport for the North in 2022 and the national streamed Bus-on-a-Map service for First Bus in 2023, now delivering billions of customer information requests per year into the FirstBus app as well as the Nextbuses live departures service in 2026.

### 2.2.2 Existing platform

TransportAPI can offer a comprehensive project delivery guarantee because the existing platform already has a high percentage of the functionality required by customers. This means that TransportAPI can plan a migration as a series of upgrades and integrations rather than as a new build with the advantage that this offers in the reduction of delivery risk. Both projects cited in 2.2.1 were delivered successfully into complex, multi-party environments at the required pace to meet delivery expectations while requirements also evolved.

### 2.2.3 Developer experience

TransportAPI offers comprehensive support to developers working with our platform. Our developer site offers interactive documentation with an API console and an extensive set of code examples at https://developer.transportapi.com/examples to shorten development timelines.

## 2.3 Supplier Relationship Management Model

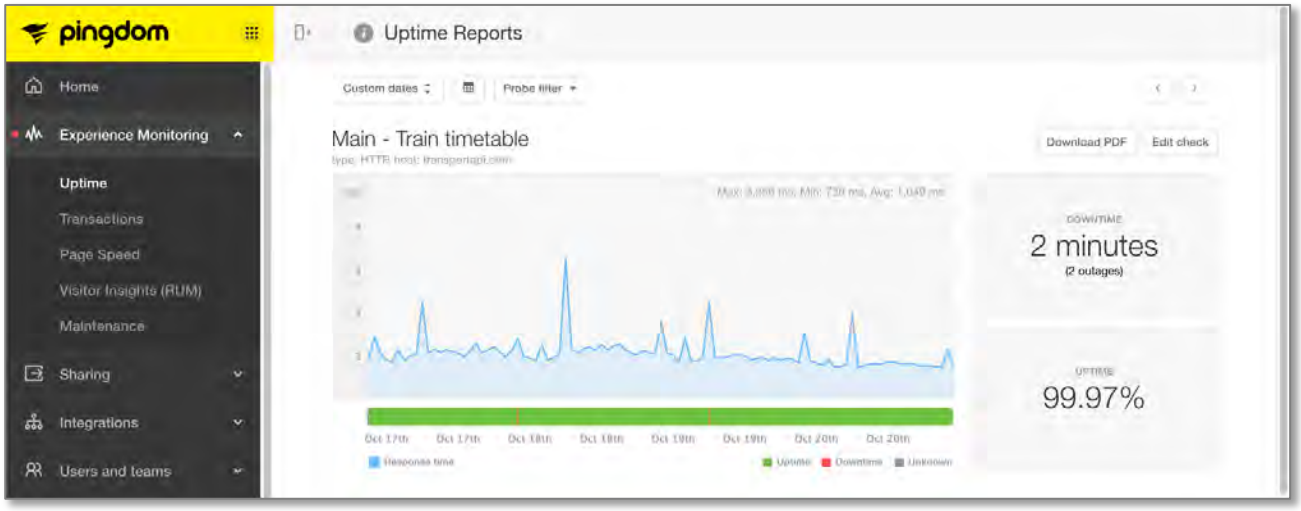
Our supplier relationship model is made up of the following eight elements.

### 2.3.1 Establishing a framework for risk monitoring and minimisation

The TransportAPI managed service platform is managed according to the principles of ISO27001, and hence our risk framework is based on this quality management scheme. Our risk register sets out the threats and vulnerabilities to all of our information and processing assets and defines the risks we accept and those requiring enhancements to contain.

### 2.3.2 Monitoring supplier performance against service level agreements

TransportAPI has suppliers in both the public and private sector. Public sector suppliers are generally the only source of supply and act as head aggregators in each area e.g. National Rail. These suppliers usually operate on the best-efforts basis and use the Open Government License to distribute their content. We monitor these suppliers closely and offer them assistance when they need it. Our private sector suppliers all operate on commercially negotiated agreements with service level agreements. We monitor the delivery of these SLAs using our external monitoring system Pingdom.




### 2.3.3 Tracking and quantifying benefits against contract rates and prices

The transactional pricing model we use for managed services is transparent, and usage is reported and tracked as part of our Supplier Relationship Management regime overseen by the account manager. Usage of the managed services is communicated to customers with Full support in a monthly usage report showing rates of consumption compared to contract allowances.

### 2.3.4 Monitoring performance against an agreed sustainability plan

TransportAPI is now a distributed organisation based in the UK and Bulgaria, and post-pandemic we have reduced our carbon footprint by giving up our offices with our staff working at home and meeting periodically for Team meetings. TransportAPI has a Sustainability policy, and we track our commitments on reducing carbon and waste, which we will share with Customers annually.

### 2.3.5 Business continuity

The TransportAPI Business Continuity Plan aims to pro-actively reduce the risks of exposure to disruptive events and to ensure that the service is able to recover quickly in the event of a crisis.

Risk exposure reduction is focused on the following areas:

- Loss of key premises: TransportAPI is a distributed organisation but can set up a project office as necessary;
- Loss of key IT infrastructure: TransportAPI is built automatically from a set of backed up and protected scripts that allow programmatic re-builds of the entire service;
- Loss of key data: most of the data held by TransportAPI is generated from other providers in government or private sector and a combination of replication, backup and the ability to re-download from external sources mitigates risk;
- Loss of upstream suppliers: short term outages of data suppliers upstream are mitigated by caching and failover to alternative sources, as well as maintenance of multiple suppliers;

- Intrusions/denial of service attacks: TransportAPI is intensively monitored for DDOS attacks and attempted intrusions and we work to minimise the attack surface;
- Security breaches: TransportAPI has its services regularly pentested by Kroll Redscan to assure information security at the highest level and has excellent outcomes.

### 2.3.6 Establishing escalation procedures

Our priority system for issue escalation is set out in SLA section 3.6 along with our service standards for restoring services if there are outages. As an aggregator a key aspect of our role is to monitor all services contributing to TransportAPI's platform and to orchestrate continuing high-quality delivery.

### 2.3.7 Establishing dispute resolution processes

TransportAPI seeks to resolve issues with customers and has an excellent record of customer relations, having never had a customer dispute. We can give a commitment to engage positively with the dispute procedures in the unlikely event they are ever needed.

### 2.3.8 Implementing a structured governance process through regular review meetings

For customers on Full Support would expect to have a regular cycle of meetings to check delivery and deal with issues (weekly) and to review progress and requirements (monthly) for customers with Full Support. TransportAPI will assign an account manager to be an ongoing point of contact with customers to deliver on requirements. The point of contact will be a member of the Management team during the roll out of the service and will transition to an account executive as the service enters BAU.

For customers on Basic and Standard support the supported delivery is set out in Schedule 1 of the Supplier Terms.

# 3 Platform and delivery

This section of the Product and Service outline sets out how we deliver the TransportAPI service into customer environments and manages it for efficient delivery.

## 3.1 Cloud native service

TransportAPI is a 'cloud-native' service that runs in the AWS cloud in the Europe London region. The managed services platform uses our own tech stack running on EC2 instances and integrating AWS services like RDS and Lambda that we manage in-house. Key elements of the platform are as follows:

- Development of our own services documented in GIThub
- Use of key open-source technologies Linux, Ruby on Rails, sidekiq, puma, OpenJDK, Kotlin, PostgreSQL and OpenSearch
- Use of the open source RabbitMQ message broker to enable low latency and high throughput data processing
- Deployment using Infrastructure-as-a-Service tools such as Chef and Terraform
- Monitoring and alerting services based on Instana and Elastic Stack
- Use of supporting platforms such as 3Scale for API management and Stripe for payments

TransportAPI is a fully auto-scaled service in all layers of the data pipeline to ensure that we can scale out all of the constituent services at the same pace when needed. TransportAPI has also developed protective 'circuit breakers' from outages in upstream services and has tuned them by working with mobile app backend providers at scale.

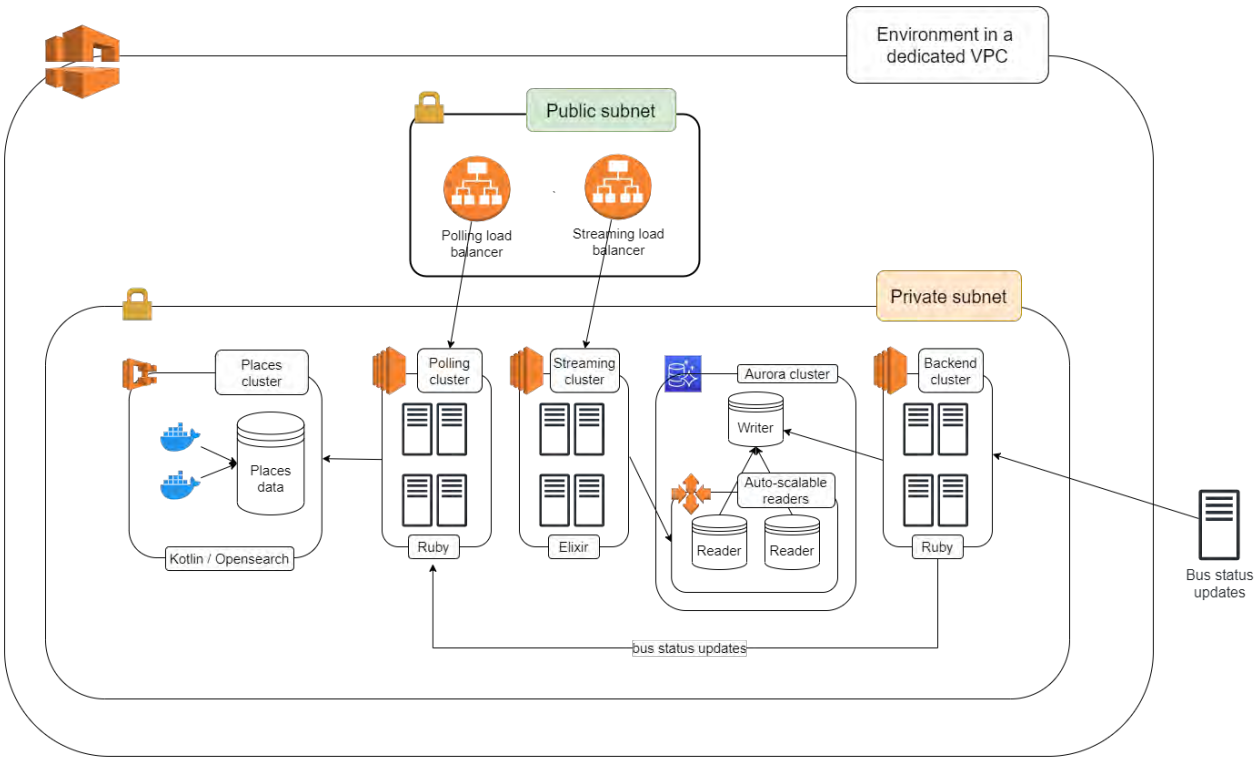
For the operation of the platform we rely on a best-in-class observability platform supporting Application Performance Monitoring (APM), infrastructure monitoring, distributed tracing and alerting.

## 3.2 Platform design

TransportAPI operates as a multi-tenant SaaS service for our small and medium size customers at http://transportapi.com/ (referred to as TransportAPI Core) and is deployed as the dedicated TransportAPI Platinum single-tenant service for enterprise customers. The platform as a whole is architected for security, efficiency and stability on AWS and has the following key features:

- The TransportAPI environment is in an AWS Virtual Private Cloud;
- Most of the processing is located in a private subnet that is not exposed on the public internet;
- Only the load balancers are in a public subnet and they run on bastion hosts.

The entire TransportAPI platform delivers around one billion GET and POST requests for multimodal journey information in JSON form every month. The generic design is shown in the diagram below.



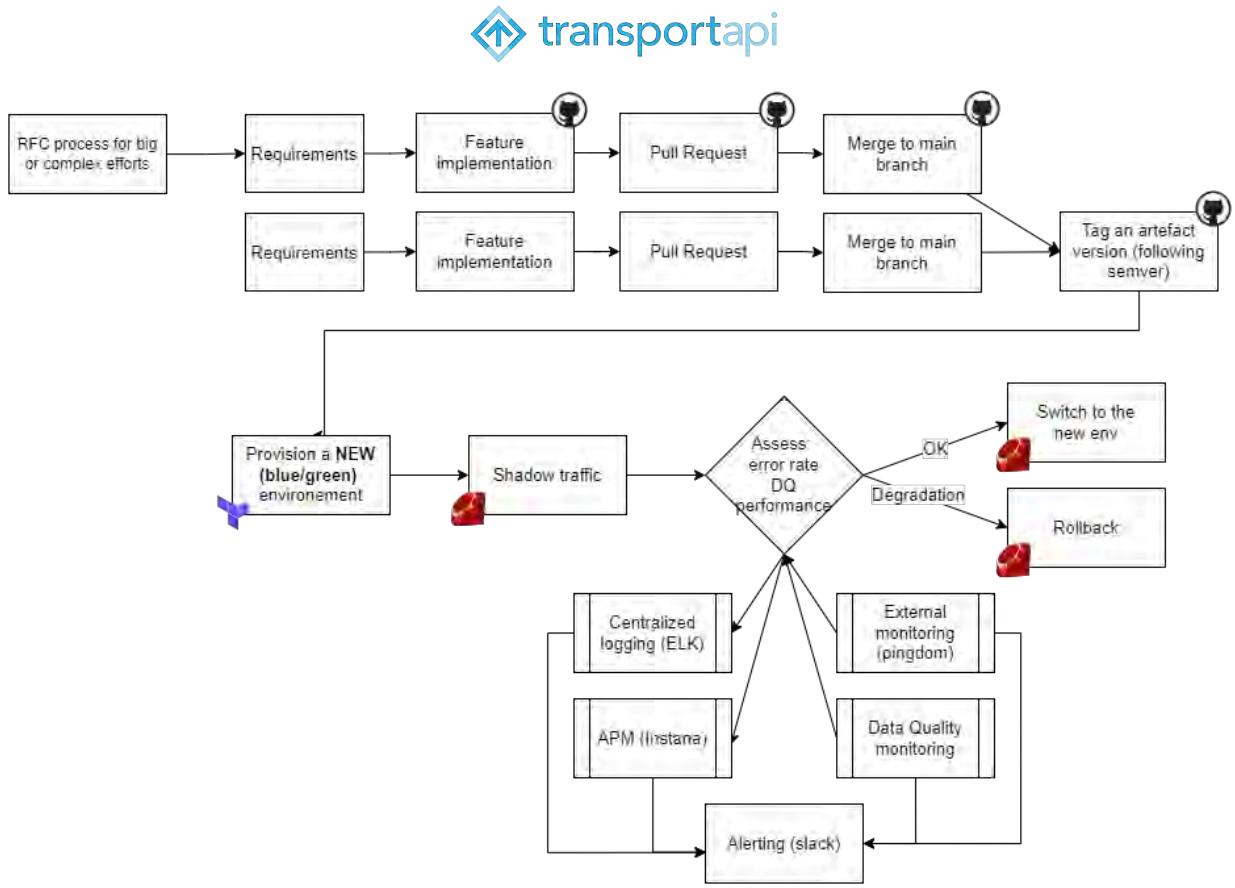

The TransportAPI Core service at http://transportapi.com/ operates identical managed services to the dedicated TransportAPI Platinum single-tenant service instances for enterprise customers that only differ in terms of scale, enterprise data integrations and the isolation from other customers.

We have experience of running dedicated Platinum services for First Bus, First Rail, Transport for West Midlands and Transport for the North, and the setup process is mostly automated using Infrastructure-as-Code services like Terraform and Chef.

Given that the platform is an array of service instances, this means that development work can be undertaken on the multi-tenant service at http://transportapi.com/ at low throughput volumes and migrated into production on a dedicated TransportAPI Platinum instance when scale is required.

## 3.3 Development and deployment process

TransportAPI produces features for the platform using a value stream process outlined in the diagram below:




TransportAPI code is managed, reviewed, tested and deployed using open-source tools including Github, CircleCI and Chef, all of which have commercial support agreements. Agile development consists of parallel development of features from the backlog which are merged into the main code branch after automated testing using linters and team review through pull requests.

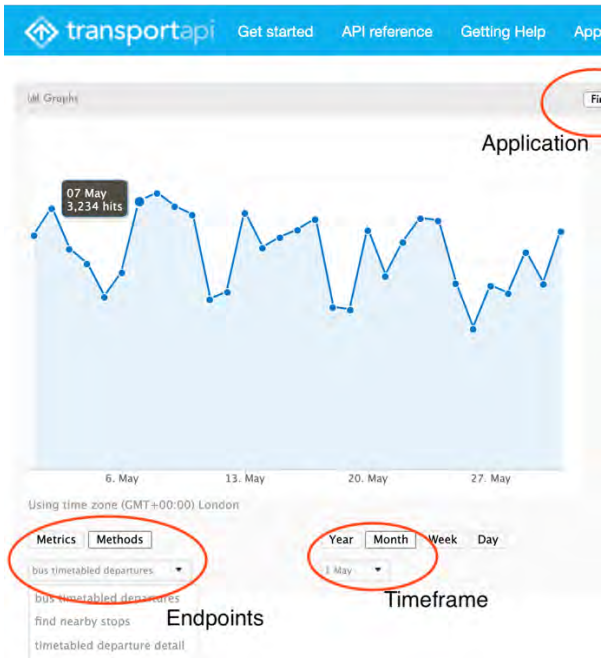
When a new artefact version is tagged then we can provision a new blue/ green environment and begin shadowing traffic in the new version. We assess performance, data quality and error rates for the shadowed traffic using our monitoring infrastructure in ELK, Instana and Pingdom until we consider that the feature can be deployed fully. Once we have deployed new code we also monitor in production for regressions, and we can roll back to the previous version if need be. This means that TransportAPI does not require service downtime to make update deployments.

Where product enhancements are developed to deliver to a customer, separate development, UAT and production environments are employed by TransportAPI for each stage of development and deployment.

## 3.4 API Management

TransportAPI managed services are delivered through RESTful endpoints using GET and POST methods. The endpoints are accessed through an API management platform based on 3Scale (by IBM Redhat). This platform supports access rights and traffic differentiation, account management, usage reporting and monitoring, rate limits and billing. The reporting options are set out in our

Service Management Guide supplied to customers and cover reports on user details and status, plan and access entitlements, and platform usage. The example below shows the reporting of usage of a specific endpoint used by a specific application over a defined timeframe.




## 3.5 Service monitoring

### 3.5.1 Performance monitoring protocol

TransportAPI approaches performance monitoring by a combination of upstream management, high-quality integration, and downstream delivery monitoring.

By upstream management we mean respecting the sources of data we receive, but also ensuring by reporting and review that we drive the sources to improve to the point where they can sustain customer expectations. We have a decade of experience of working with key upstream organisations in the industry and within enterprises to drive these outcomes.

By high quality integration, we mean that we hold ourselves to the highest standards of data integration that is rigorously internally verified through automated and reviewing checks.

By downstream delivery monitoring, we mean independent monitoring of our output that is shared with customers. TransportAPI monitors its endpoints using the external monitoring system Pingdom by SolarWinds.

The standard terms that TransportAPI offers in its managed services are set out in Schedule 3 of the Supplier Terms.

TransportAPI endpoints typically have a 99.9% uptime over the last 3 years in Pingdom reporting excluding upstream outages. This figure does not exclude maintenance windows as we don't have them: this is a 24/7 uptime.

### 3.5.2 Logging

TransportAPI logs usage of our managed services using our 3Scale API Management system and we log system events into our Elastic Stack service, exposing analytics through Kibana. System errors are tracked by IBM Instana and alerts are published to the devops team for action.

## 3.6 SLA management

TransportAPI operates several internally facing services to monitor the availability and response time for managed services endpoints, including Pingdom and Instana. If alerts are detected by these monitoring services, the TransportAPI devops team will notify customers, take action to determine the root cause of a service incident and bring the managed service back within the Support service resolution period.

As TransportAPI acts as a massive real-time integration environment between many real-time sources, service incidents can be complex to diagnose. If there is no internal failure within TransportAPI infrastructure, the TransportAPI devops team will check upstream inputs for:

- Responsiveness
- Syntactic correctness
- Semantic content

This triage process can be complex since a service that is providing data to us (responsive), with the right data structures (good syntax), but invalid content (incorrect semantics) may not be recognised as failing by the upstream provider. TransportAPI will contact upstream providers if they appear to be responsible for a service incident to provide details and seek recognition of the problem. When this assessment has been completed the triage process can allocate responsibility for a service incident mitigation. The results of this triage process are available to customers on the Standard Support tier.

The TransportAPI service uses the following standards for delivery:

- P1: all TransportAPI managed services are unavailable and/or inaccessible to users (Response: detection and report to customer within 30 minutes. Service standard for resolution: 4 hours);
- P2: one or more TransportAPI managed services endpoints fail 10 consecutive one-minute SLA checks or where the response time rises 50% above the SLA metric for 10 consecutive one minute checks (Response: detection and report to customer within two hours. Service standard for resolution: 1 business day);

- P3: service performance is affected by behaviour of an upstream or downstream service outside TransportAPI managed services (Response: detection and report to customer within 4 hours. Best efforts for resolution: 2 business days).

TransportAPI P3 incidents relate to upstream outages as we usually cannot do any engineering to resolve the issue after we have activated any applicable circuit-breaker, and in the case of open government data there is no commercial SLA that can be used to drive resolutions. In this case we devote resources to helping the upstream service diagnose the outage and validate solutions to speed up resolution. However, this is necessarily on a 'best efforts' basis as it is outside our infrastructure.

## 3.7 Service desk integration

TransportAPI has a support ticketing system provided by Freshdesk that offers help by email. Issues are triaged by support agents and assigned to a specific member of the TransportAPI team. For customers on a Full support tier an Extended hours support option is available to cover extended early mornings and evenings, bank holiday and weekend hours as set out in Schedule 3 of the Supplier Terms.

## 3.8 Information security

TransportAPI operates its platform to the principles of ISO27001 on Information Security based on Plan, Operate, Evaluate, Improve quality management model. Information security is driven from the company leadership who receive a briefing on key issues at each Board meeting and who delegate implementation to the management team. The management team have developed an information security policy that makes a commitment to secure engineering, risk assessment, appropriate controls and staff awareness.

TransportAPI uses Kroll Redscan to conduct pentests on our infrastructure and results have demonstrated only ‘Low’ priority issues.

## 3.9 Documentation

TransportAPI offers full documentation of its managed service endpoints at http://developer.transportapi.com. On this site, you will find the following resources:

- Developer documentation for request and response structures associated with each endpoint in OpenAPI form;
- An API console that allows developers to experiment with requests directly on the developer site and to see the live response produced;
- Examples of code that can be used to call TransportAPI and produce specified outputs as exemplars to developers.

# 4 TransportAPI Managed Services

TransportAPI provides managed services that power all the key product and retail services for customer information that are required by apps, screens and analytics. Ensuring that the single version of the operating truth is propagated into these applications, used to deliver journey plans and matched to fare options for bus and rail is crucial to ensuring the best customer experience.

The following sections of this Service Definition document sets out how each of the TransportAPI Managed Services can deliver on these objectives.

## 4.1 Places Managed Service

The single version of operating truth rests heavily on stops/ stations data as the starting points for customers seeking bus and rail information or requesting journey plans. However, this data is derived from national datasets, such as NaPTAN and CORPUS from the Department for Transport and Network Rail, and outside any operator's exclusive control. Local authorities have a prime role in the management of bus stops, along with certain significant hub operators like airports who are in the private sector. TransportAPI Places Managed Service maintains the most up-to-date data from NaPTAN and CORPUS and matches schedules and stops/ stations to ensure permanent alignment.

Bus stop/ rail station data along with all other public transport stopping places, postcodes, addresses and points of interest are available by querying the Places managed service endpoint by name or bounding box. There are a range of query options including one to find bus stops near a location filtered by the stops served by a particular bus route.

Due to the way that transport data is collected locally and regionally, Tube, Metro, Subway, Tram, River boat and Ferry stopping places and schedules are distributed with bus data. TransportAPI provides access to this data through the Places Managed Service and the Bus Information Managed Service.

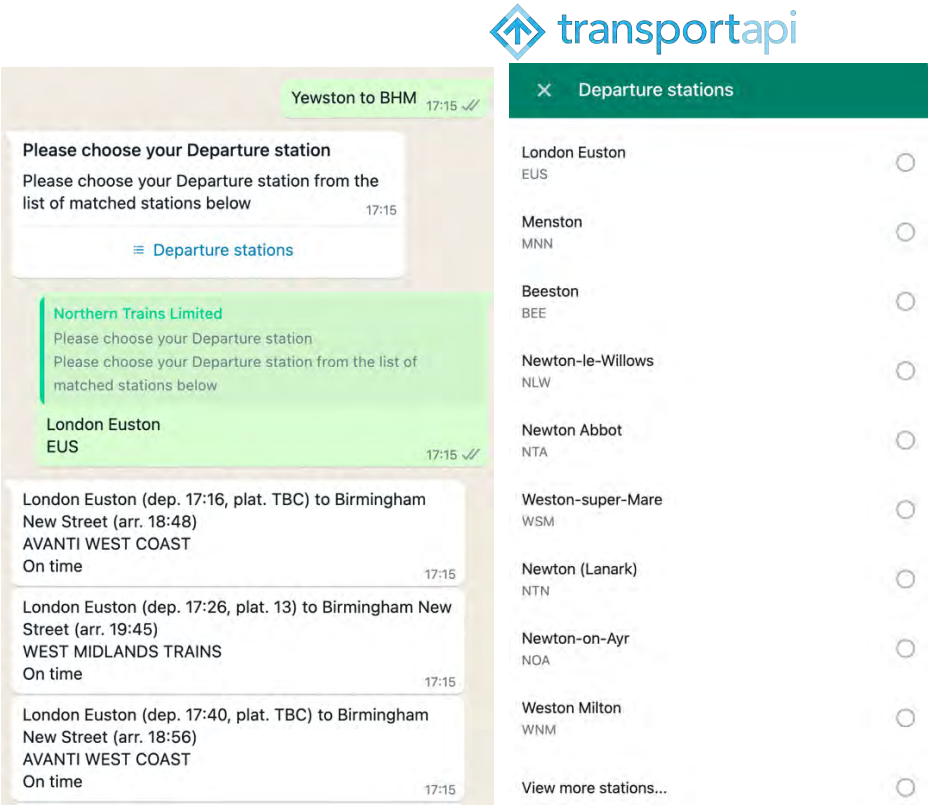
The TransportAPI Places Managed Service is a key companion to the TransportAPI Journey Planner Managed Service through the delivery of origin and destination reference data required by the journey plan. If the origin and destination are selected by the user typing input this must be efficiently translated into a form that can be sent to a journey planning engine. TransportAPI provide a massively scaled transport access, location and point of interest look up service using the Places managed service. The criteria used to select results are shown in the table below:

| Dimension | Act as Filter | Contribute to combined rank | Example |
|---|---|---|---|
| | | | exeter, exe, exteere, ecseter |
| **Geo** | Bounding box | - | min_lat, max_lat, min_lon, max_lon |
| | - | Distance from point | lat, lon |
| **Type** | Transport nodes, streets, POIs, etc | - | bus_stop, train_station, tram_stop, tube_station, settlement, streets, poi, event… |
| | - | Accounts for importance of type | train_station ranked above bus_stop |
| **Record Importance** | - | Accounts for importance of each record | e.g. Victoria station ranked above Sugar Loaf station (least used in GB!) |

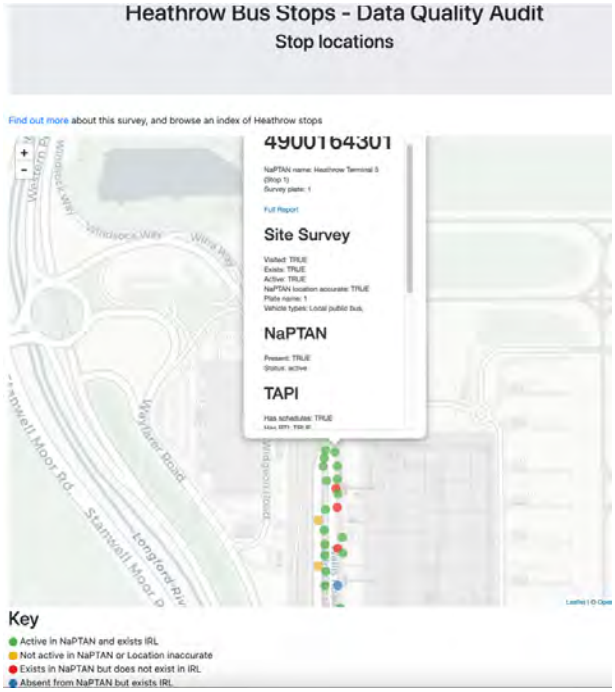
This service can support a wide range of customer input to help find the desired origin or destination point including:

- Homophones (sound-alike) support: search for "Yewston Station" must find Euston Station
- Typographic errors: search for “Mnchestr” must find Manchester
- Text matching that prioritises codes used in transport e.g. National Rail CRS codes like VIC

TransportAPI have worked with Northern Rail to create a WhatsApp chat bot capable of fuzzy matching user input to appropriate stations on the rail network as shown in the following example:




TransportAPI undertakes stop and route data quality audits for complex locations if regular imports show that there are significant problems there. This work is undertaken using our Data Quality Audit tooling, as seen in the example below for Heathrow Airport.




Given the single tenant nature of the TransportAPI Platinum service it is also possible to undertake custom integrations with our stop database e.g. to support private zone stops that are not covered by NaPTAN. Such integrations require the additional data to be available as-a-service (e.g. download).

Although TransportAPI Platinum offers enterprise integrations, our philosophy as an aggregator is to 'respect the source' by problem solving through dialogue with national data source owners. For example, where a number of stops are used interchangeably at a bus station, we would recommend stops are grouped in NaPTAN, rather than to associate all departures with one stop and exclude the others from the database in a custom change. We can also support searching for stop groups by NaPTAN stopArea to return locality stop sets.

## 4.2 Bus Information Managed Service

### 4.2.1 Schedules

Accurate and up-to-date bus schedules are the foundation of customer information for product and retail in the customer experience. However, bus schedules are volatile and decisions on their content are distributed across the operator.

The TransportAPI platform uses our TransXchange Parser to import TXC version 2.1/2.4/2.5 files into our schedules database from customer, Traveline TNDS and BODS sources. This import is benchmarked against key national datasets such as National Operating Codes (NOC) to ensure compatibility of the schedules across all managed services. This enables the delivery of features like branding lookup for the UI using operating codes.

As part of our import workflow we run our TXC Data Quality Metrics reporting tool to assess the performance of the data against customer-defined metrics (as shown in the example below).


The TransportAPI reviewing process implements policies on:

- Frequency of updates depending on data currency;

- Decisions if data files are late or have major omissions/ errors;
- Alignment of all data importing processes with journey planner updates;
- Data quality thresholds triggering publish/ refer decisions;
- Special imports for holidays.

TransportAPI also run a number of internal data quality checks to ensure that the downstream managed services produce the best quality outputs.

TransXchange presents a bus registration-oriented view of the schedules, which is sometimes at variance with the customer experience. Hence, some individual vehicles continue across separate services or with different drivers and route cross links may need to be shown in the UI. In line with the TransportAPI policy to 'respect the source', the backend will keep the routes separate and app/ web application logic is required to show route cross links according to the required business logic.

Bus schedules once they have been imported from TransXchange are accessible through TransportAPI Bus Information Managed Service on a specific date as (with endpoint names in brackets):

- Bus service metadata and footprint (bus/services)
- Departures from a specific stop filtered by route, operator and time (bus/stop_timetables)
- Journeys between any pair of stops on a route (bus/route/timetable)
- Whole individual journeys (bus/service_timetables)

Bus schedules for a route or stop at the current time can be augmented by real-time information (see below for bus live departures aka real-time information- RTI). TransportAPI can also be used to present the schedule data in the form of a PDF timetable for main stops or all calling points, as shown in the example below produced by Travel South Yorkshire.

**Full Service Timetable**

**Main Calling Points**

| | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Mill Road/St Wandrilles Close | 05:00 | 05:30 | 05:45 | 06:00 | 06:30 | 07:00 | 07:30 | 08:00 | 08:25 | 08:50 | 09:20 | 09:50 |
| Moonshine Lane/Galsworthy Road | 05:13 | 05:43 | 05:58 | 06:13 | 06:44 | 07:14 | 07:44 | 08:14 | 08:39 | 09:04 | 09:34 | 10:04 |
| Pitsmoor Road/Pinfold Lane | 05:23 | 05:53 | 06:08 | 06:23 | 06:55 | 07:25 | 07:55 | 08:25 | 08:52 | 09:17 | 09:47 | 10:17 |
| Snig Hill/CG9 | 05:31 | 06:01 | 06:16 | 06:31 | 07:03 | 07:33 | 08:03 | 08:34 | 09:01 | 09:26 | 09:56 | 10:26 |
| Ecclesall Road/Neill Road | 05:39 | 06:09 | 06:24 | 06:39 | 07:14 | 07:46 | 08:19 | 08:48 | 09:15 | 09:40 | 10:10 | 10:42 |
| Crimicar Lane/Westminster Crescent | 05:53 | 06:23 | 06:38 | 06:53 | 07:29 | 08:01 | 08:34 | 09:03 | 09:30 | 09:55 | 10:26 | 10:58 |
| | | | | | | | | | | | | |
| Mill Road/St Wandrilles Close | 10:20 | 10:50 | 11:20 | 11:50 | 12:20 | 12:50 | 13:20 | 13:50 | 14:20 | 14:50 | 15:20 | 15:50 |
| Moonshine Lane/Galsworthy Road | 10:34 | 11:04 | 11:34 | 12:04 | 12:34 | 13:04 | 13:34 | 14:04 | 14:34 | 15:04 | 15:34 | 16:04 |
| Pitsmoor Road/Pinfold Lane | 10:48 | 11:18 | 11:48 | 12:18 | 12:48 | 13:18 | 13:48 | 14:18 | 14:45 | 15:15 | 15:45 | 16:15 |
| Snig Hill/CG9 | 10:58 | 11:28 | 11:58 | 12:28 | 12:58 | 13:28 | 13:58 | 14:28 | 14:56 | 15:26 | 15:56 | 16:26 |
| Ecclesall Road/Neill Road | 11:15 | 11:45 | 12:15 | 12:45 | 13:15 | 13:45 | 14:15 | 14:50 | 15:15 | 15:45 | 16:15 | 16:50 |
| Crimicar Lane/Westminster Crescent | 11:31 | 12:01 | 12:31 | 13:01 | 13:31 | 14:01 | 14:31 | 15:07 | 15:32 | 16:02 | 16:32 | 17:07 |
| | | | | | | | | | | | | |
| Mill Road/St Wandrilles Close | 16:25 | 16:55 | 17:25 | | | | | | | | | |
| Moonshine Lane/Galsworthy Road | 16:39 | 17:09 | 17:39 | | | | | | | | | |
| Pitsmoor Road/Pinfold Lane | 16:50 | 17:20 | 17:50 | | | | | | | | | |
| Snig Hill/CG9 | 17:01 | 17:30 | 18:00 | | | | | | | | | |
| Ecclesall Road/Neill Road | 17:23 | 17:50 | 18:20 | | | | | | | | | |
| Crimicar Lane/Westminster Crescent | 17:38 | 18:04 | 18:34 | | | | | | | | | |

**All Calling Points**

| | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Mill Road/St Wandrilles Close | 05:00 | 05:30 | 05:45 | 06:00 | 06:30 | 07:00 | 07:30 | 08:00 | 08:25 | 08:50 | 09:20 | 09:50 |
| Church Street/Mill Road | 05:01 | 05:31 | 05:46 | 06:01 | 06:31 | 07:01 | 07:31 | 08:01 | 08:26 | 08:51 | 09:21 | 09:51 |
| St Marys Lane/Church Street | 05:01 | 05:31 | 05:46 | 06:01 | 06:32 | 07:02 | 07:32 | 08:02 | 08:27 | 08:52 | 09:22 | 09:52 |
| High Street/St Marys Close | 05:02 | 05:32 | 05:47 | 06:02 | 06:32 | 07:02 | 07:32 | 08:02 | 08:27 | 08:52 | 09:22 | 09:52 |
| High Street/Sycamore Road | 05:03 | 05:33 | 05:48 | 06:03 | 06:33 | 07:03 | 07:33 | 08:03 | 08:28 | 08:53 | 09:23 | 09:53 |
| Tunwell Avenue/Wordsworth Avenue | 05:04 | 05:34 | 05:49 | 06:04 | 06:35 | 07:05 | 07:35 | 08:05 | 08:30 | 08:55 | 09:25 | 09:55 |
| Wheata Road/Tunwell Avenue | 05:05 | 05:35 | 05:50 | 06:05 | 06:35 | 07:05 | 07:35 | 08:05 | 08:30 | 08:55 | 09:25 | 09:55 |
| Colley Crescent/Rokeby Drive | 05:06 | 05:36 | 05:51 | 06:06 | 06:37 | 07:07 | 07:37 | 08:07 | 08:32 | 08:57 | 09:27 | 09:57 |
| Colley Road/Colley Crescent | 05:07 | 05:37 | 05:52 | 06:07 | 06:37 | 07:07 | 07:37 | 08:07 | 08:32 | 08:57 | 09:27 | 09:57 |
| Colley Road/Turie Avenue | 05:08 | 05:38 | 05:53 | 06:08 | 06:38 | 07:08 | 07:38 | 08:08 | 08:33 | 08:58 | 09:28 | 09:58 |
| Holgate Avenue/Holgate Crescent | 05:08 | 05:38 | 05:53 | 06:08 | 06:39 | 07:09 | 07:39 | 08:09 | 08:34 | 08:59 | 09:29 | 09:59 |
| Holgate Avenue/Margetson Road | 05:09 | 05:39 | 05:54 | 06:09 | 06:40 | 07:10 | 07:40 | 08:10 | 08:35 | 09:00 | 09:30 | 10:00 |
| Adlington Road/Deerlands Avenue | 05:10 | 05:40 | 05:55 | 06:10 | 06:41 | 07:11 | 07:41 | 08:11 | 08:36 | 09:01 | 09:31 | 10:01 |
| Adlington Road/Falstaff Road | 05:11 | 05:41 | 05:56 | 06:11 | 06:41 | 07:11 | 07:41 | 08:11 | 08:36 | 09:01 | 09:31 | 10:01 |
| Southey Hill/North Hill Road | 05:12 | 05:42 | 05:57 | 06:12 | 06:43 | 07:13 | 07:43 | 08:13 | 08:38 | 09:03 | 09:33 | 10:03 |
| Moonshine Lane/Galsworthy Road | 05:13 | 05:43 | 05:58 | 06:13 | 06:44 | 07:14 | 07:44 | 08:14 | 08:39 | 09:04 | 09:34 | 10:04 |
| Moonshine Lane/Southey Rise | 05:14 | 05:44 | 05:59 | 06:14 | 06:45 | 07:15 | 07:45 | 08:15 | 08:40 | 09:05 | 09:35 | 10:05 |
| Herries Road/Herries Drive | 05:15 | 05:45 | 06:00 | 06:15 | 06:46 | 07:16 | 07:46 | 08:16 | 08:41 | 09:06 | 09:37 | 10:07 |
| Herries Road/Boynton Road | 05:15 | 05:45 | 06:00 | 06:15 | 06:47 | 07:17 | 07:47 | 08:17 | 08:43 | 09:08 | 09:37 | 10:07 |
| Herries Road/Longley Lane | 05:16 | 05:46 | 06:01 | 06:16 | 06:48 | 07:18 | 07:48 | 08:18 | 08:44 | 09:09 | 09:38 | 10:08 |
| Norwood Avenue/Fairbank Road | 05:17 | 05:47 | 06:02 | 06:17 | 06:48 | 07:18 | 07:48 | 08:18 | 08:44 | 09:09 | 09:39 | 10:09 |
| Longley Avenue West/Musgrave Drive | 05:18 | 05:48 | 06:03 | 06:18 | 06:49 | 07:19 | 07:49 | 08:19 | 08:45 | 09:10 | 09:40 | 10:10 |
| Shirecliffe Road/Musgrave Road | 05:18 | 05:48 | 06:03 | 06:18 | 06:50 | 07:20 | 07:50 | 08:20 | 08:46 | 09:11 | 09:41 | 10:11 |
| Shirecliffe Road/Standish Drive | 05:19 | 05:49 | 06:04 | 06:19 | 06:50 | 07:20 | 07:50 | 08:20 | 08:46 | 09:11 | 09:41 | 10:11 |

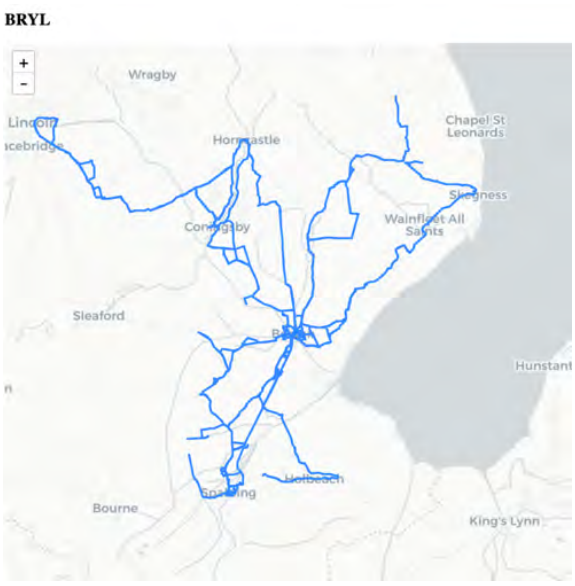
### 4.2.2 Route geometry

TransportAPI offers bus route geometry information through the Bus Information Managed Service to support customer enquiries, bus tracking and fare calculation. Bus route geometry is available by individual route and for every journey operated on that route. This data can be accessed at an aggregated level by bus services, operators or geofences such as local authority boundaries. Journey patterns for each route variation can be displayed from data in Bus Information Managed Service through the bus/services endpoint:

**To Keele (outbound)**

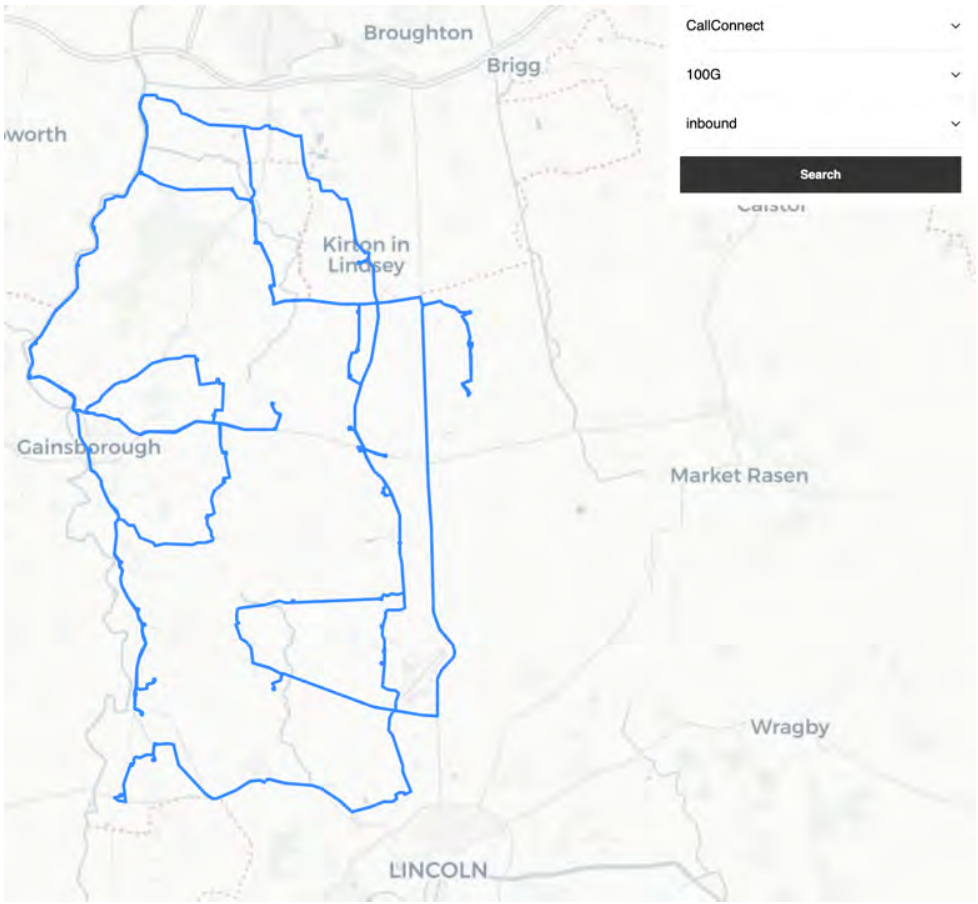
| 34 Occurrences | 6 Occurrences | 15 Occurrences | 85 Occurrences | 1 Occurrence | 3 Occurrences |
|---|---|---|---|---|---|
| Bus Station (stand V) | Bus Station (stand A) | Bus Station (stand D) | Bus Station (stand V) | Bus Station (stand D) | Bus Station (Stand 12) |
| Harding Road | Harding Road | Harding Road | Harding Road | Harding Road | Merrial Street (Stop C) |
| Wellesley Street | Wellesley Street | Wellesley Street | Wellesley Street | Wellesley Street | Queen Elizabeth Park |
| Stoke-on-Trent College | Stoke-on-Trent College | Stoke-on-Trent College | Stoke-on-Trent College | Stoke-on-Trent College | Drayton Street |
| Seaford Street | Seaford Street | Seaford Street | Seaford Street | Seaford Street | St Mary's Drive |
| Film Theatre | Film Theatre | Film Theatre | Film Theatre | Film Theatre | Rotterdam Road |
| Stoke-on-Trent Railway Station | Stoke-on-Trent Railway Station | Stoke-on-Trent Railway Station | Stoke-on-Trent Railway Station | Stoke-on-Trent Railway Station | House No. 220 |
| Seven Arches | Seven Arches | Seven Arches | Seven Arches | Seven Arches | Wain Avenue |
| Lytton Street | Lytton Street | Lytton Street | Lytton Street | Lytton Street | Gallowstree Lane |
| stop Q | stop Q | stop Q | stop Q | stop Q | Home Farm Drive |
| stop K | stop K | stop K | stop K | stop K | Barnes Hall |
| St Andrew's Square | St Andrew's Square | St Andrew's Square | St Andrew's Square | St Andrew's Square | Darwin Building |
| Higson Avenue | Higson Avenue | Higson Avenue | Higson Avenue | Higson Avenue | Horwood Hall |
| Richmond Street | Richmond Street | Richmond Street | Richmond Street | Richmond Street | Lindsay Hall |
| Quarry Road | Quarry Road | Quarry Road | Quarry Road | Quarry Road | Holly Cross Hall |
| Holy Trinity Church | Holy Trinity Church | Holy Trinity Church | Holy Trinity Church | Holy Trinity Church | Sneyd Arms PH |
| Nelson Road | Nelson Road | Nelson Road | Nelson Road | Nelson Road | |
| main entrance link | main entrance link | main entrance link | main entrance link | main entrance link | |
| Chapel | Chapel | Chapel | Chapel | Chapel | |
| The Avenue | The Avenue | The Avenue | The Avenue | The Avenue | |
| Cherry Tree PH | Cherry Tree PH | Cherry Tree PH | Cherry Tree PH | Cherry Tree PH | |
| Bus Station (Stand 12) | Bus Station (Stand 12) | Bus Station (Stand 12) | Bus Station (Stand 12) | Bus Station (Stand 12) | |
| | Merrial Street (Stop C) | Merrial Street (Stop C) | Merrial Street (Stop C) | | |
| | Queen Elizabeth Park | Queen Elizabeth Park | Queen Elizabeth Park | | |
| | Drayton Street | Drayton Street | Drayton Street | | |
| | St Mary's Drive | St Mary's Drive | St Mary's Drive | | |
| | Rotterdam Road | Rotterdam Road | Rotterdam Road | | |
| | House No. 220 | House No. 220 | House No. 220 | | |
| | Wain Avenue | Wain Avenue | Wain Avenue | | |
| | Gallowstree Lane | Gallowstree Lane | Gallowstree Lane | | |
| | Home Farm Drive | Home Farm Drive | Home Farm Drive | | |
| | Barnes Hall | Barnes Hall | Barnes Hall | | |
| | Darwin Building | Darwin Building | Darwin Building | | |
| | Horwood Hall | Horwood Hall | Horwood Hall | | |
| | Lindsay Hall | Lindsay Hall | Lindsay Hall | | |
| | Holly Cross Hall | Holly Cross Hall | Holly Cross Hall | | |
| | Sneyd Arms PH | Sneyd Arms PH | Sneyd Arms PH | | |

Individual journey route geometry (aka occurrences of a journey pattern) can be accessed for a route number and operator along with the full schedule for that journey. The journey route geometry (and stops) is snapped to OpenStreetMap and follows the 'bus-drivable' route for the journey in both inbound and outbound directions as shown in the example below. Bus route geometry can also be served for each individual stop-to-stop segment of each route in GeoJSON format. Bus performance data is available for all bus route segments (see section 4.5 below).


Bus route geometry can be accessed at an aggregated level by sending the required boundary in coordinate form in a POST request. This can return all bus routes in an area with summary information about each route and operator as in the example above.

This output can be used to draw full route network maps for an operator or city. Where Demand Responsive Transport networks publish details of their service areas this is included in an aggregated request as shown for Lincolnshire Call Connect in the example below.



### 4.2.3 Real-time information

TransportAPI managed services deliver real time information (RTI) to customers at scale. TransportAPI is responsible for the complex data integration required to compose accurate bus live departure boards nationwide by integrating SIRI-SM and custom RTI data from multiple sources including TfL, Nextbuses, uTrack and Ticketer. This involves navigating a source tree for the selected stop in order to define which services are running at the time of interest and which live sources are available to check. TransportAPI Bus Information Managed Service then blends the results together, returning timetables where live is not available for any specific departure.

TransportAPI is configurable for the sources that are used in the RTI output either in the request that is made (for Nextbuses inclusion or exclusion), using the source configuration definition that controls access to specific sources. RTI results are also managed geographically in TransportAPI, for example to return results from TfL Countdown for the Greater London area.

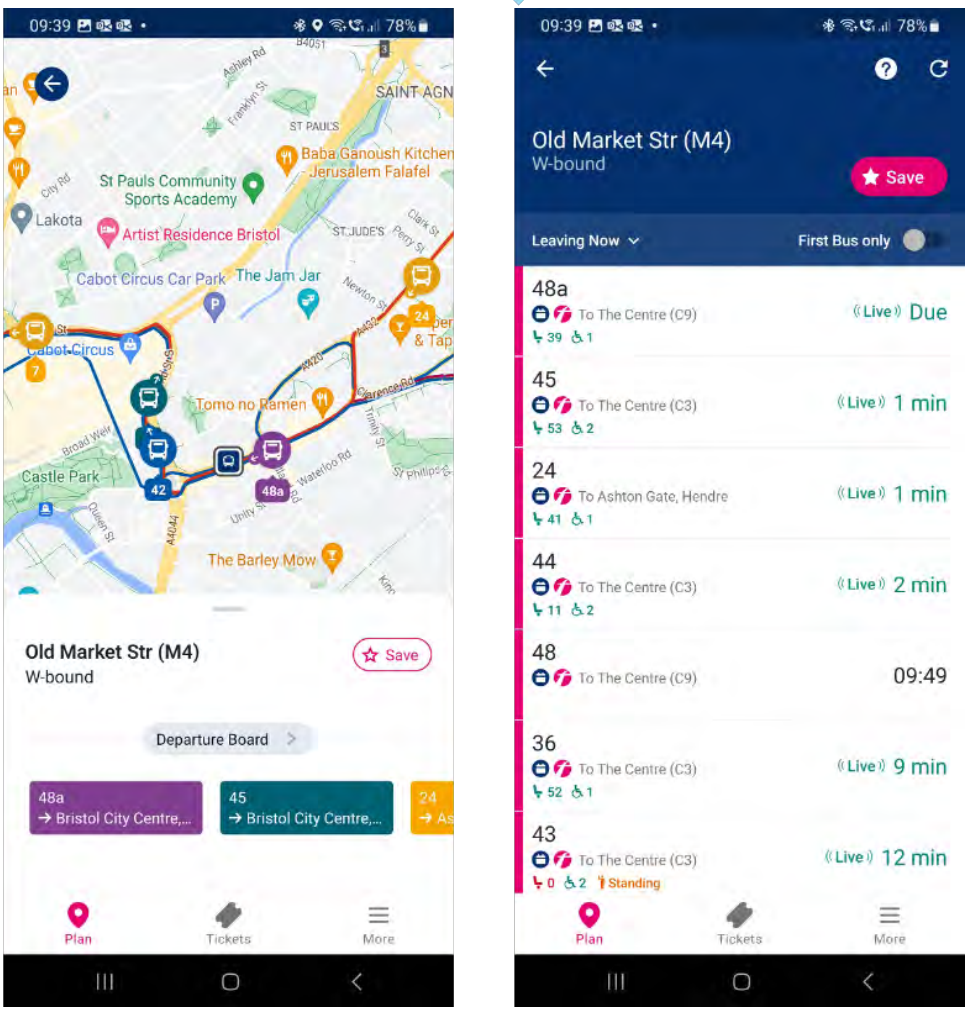
RTI data can be acquired from stop prediction sources such as SIRI-SM or to track buses using SIRI-VM. TransportAPI has a full operational integration with the industry-standard Ticketer SIRI-SM and SIRI-VM feeds. TransportAPI SIRI-SM feeds power the following features (with endpoint names in brackets):

- Live departure boards for all bus stops nationally (bus/stop_timetables)
- Service cancellations when available (bus/stop_timetables, bus/service_timetables)

Bus RTI data is served from the same stop_timetables endpoint as schedule data so that all the options available to configure schedules are also available for RTI. This is achieved by adding an additional query string parameter is used to add ‘expected’ times to the ‘aimed’ (i.e. scheduled) time of departure.

Despite the longstanding use of departure boards, users are increasingly favouring the use of bus tracking as a way to assess how long the bus will take to arrive at their stop. The example below shows the two alternative ways to present the same RTI information for a stop shown in the lower centre of the map. Buses-on-a-map (BOAM) also shows the buses that have recently departed.

The delivery of bus tracking data from the vehicle to the app using SIRI-VM data and websockets streaming technology with a latency of 10 to 15 seconds from bus to app is an industry leading delivery in this area, and highly rated by customers as shown by app scores for the First Bus app. For busy stops in city centres dozens of vehicles are being tracked and their positions streamed into the app in real time, both before and after departure from the stop. Currently, this websocket bus tracking delivery is integrated between backend and app through a very low latency pipeline to ensure it is always interactive.




At the heart of the delivery of buses-on-a-map is the cross-referencing service linking schedules and real-time positions received from the bus. TransportAPI tracks the rates of cross-referencing success to detect mismatches or cancellations so that every service is shown.

By using the Buses-on-a-map (BOAM) feature of the Bus Information Managed Service users of an operator app can view live bus tracking streamed into their app with low latency from the on-bus tracking (including real-time occupancy information- if available in the SIRI-VM feed).

TransportAPI BOAM service requires access to the SIRI-SM and SIRI-VM feeds from the operator's provider. If the provider is Ticketer TransportAPI already has a production interconnection. A one-off integration process is required to define how the feeds map onto opcos/ routes, and to arrange whitelisting of TransportAPI servers. Usage of the SIRI-SM feed is accounted for by the requests made by app/ web users and is thus variable by customer usage. Usage of the SIRI-VM feed is accounted for in two stages:

- Ingest of the raw SIRI-VM messages into TransportAPI infrastructure at the back end, which depends on how many buses are running and varies very little week by week
- Delivery of the websockets payloads directly to the customer app, which is variable by customer usage

Delivery of the TransportAPI BOAM service requires infrastructure engineering with the app provider in order to set up the OAuth 2.0 authentication and deliver the JSON-RPC requests through the low latency websocket connection. Cancellations will be used to update the service when they are made available by the operator's provider.

## 4.3 Bus Disruption Managed Service

Disruption to bus services can be divided as follows:

- Either Planned or immediate changes
- One of Schedules, Stops or Routes disruption

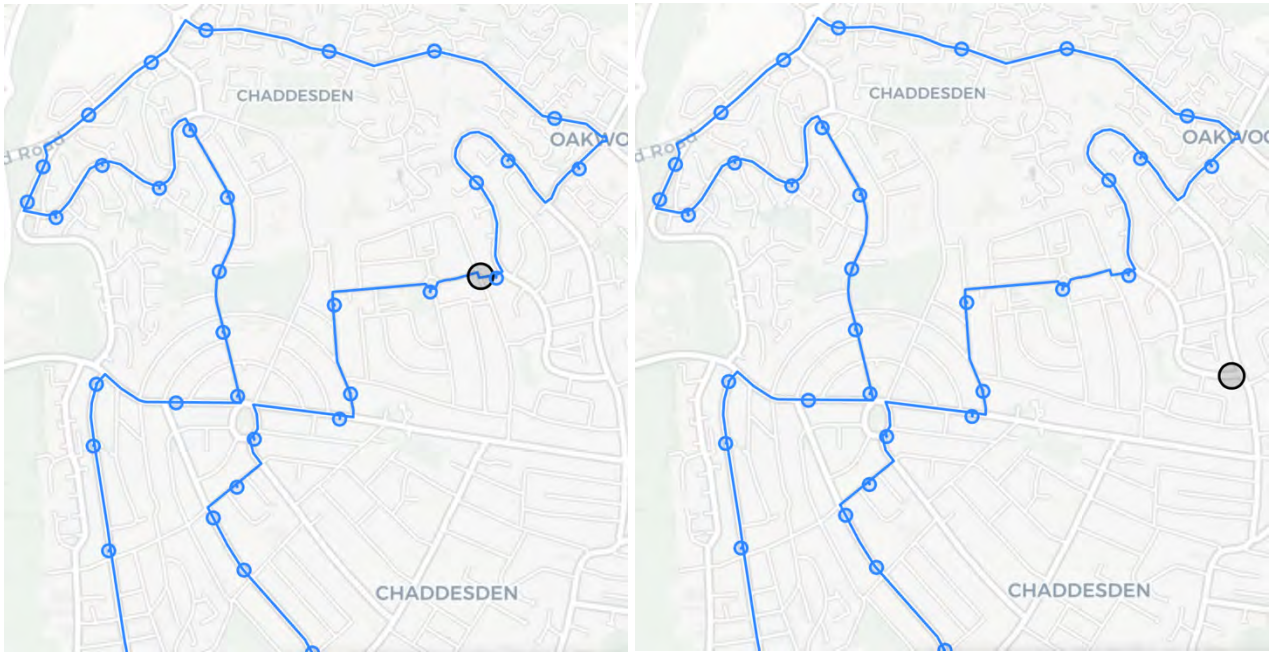
| Disruption management | **Schedules** | **Stops** | **Route** | **RTI** | **Performance** |
|---|---|---|---|---|---|
| **Planned changes** | TransXchange update | NaPTAN update | TransportAPI bus information | Live data | Actual behaviour |
| **Immediate change** | TransportAPI Bus disruptions | TransportAPI Bus disruptions | TransportAPI Bus disruptions | Live data | Actual behaviour |

Planned schedule change is generally managed through the weekly or sub-weekly TransXchange updates. The TransportAPI import process manages this change ranging from overnight road closures to long term diversions. The Bus Information Managed Services will always present the up-to-date schedules at the Bus Information Managed Service endpoints.

Planned stop change is managed by local authorities, operators and Dept. for Transport NaPTAN team under national guidance. If stops are being added, removed or moving more than 20m then they will be updated and the change will be reflected in TransportAPI managed services.

Planned route change is a consequence of planned schedule and stop change, and this change is propagated into the bus route geometry database held by TransportAPI from TransXchange, operator and NaPTAN sources.

Note that RTI data intrinsically reflects the current live bus behaviour, and bus tracking can be used to detect disruption as in the example below from the Bus information managed service RTI endpoints where a bus in the right-hand map (shown as a grey circle) is running off-route:




In the Bus Performance Managed Service (see section 4.5) actual movements of each bus journey are logged into the TransportAPI bus performance archive.

Immediate schedule, stop and route change is now being collected and disseminated by the Dept. of Transport. Currently only certain local transport authorities including SYMCA, WYCA, TfGM, West of England, Merseytravel, and Cornwall contribute data although it is planned that all authorities will eventually use this service. This data is now being published on the Department for Transport BODS Disruption service in SIRI-SX, although the service does not offer an SLA on availability.

The TransportAPI Bus Disruption Managed Service caches, validates and publishes this data in the universally supported JSON format with the following resources:

- A source of individual disruption events with full description and alternative travel options;
- A resource that returns collections of disruption events, matching search criteria (e.g. lists of stops, operators, services, timescales, etc);
- Augmentation of existing Bus information managed service endpoints to provide a summary of the relevant disruption (with a short description, and link to the full disruption resource).

In the DfT service coverage area Immediate disruption will in future automatically appear in Bus information results, and it will be possible to drill down to the full details.

## 4.4 Bus Fares Managed Service

The TransportAPI Bus Fares managed service is based on the Netex data produced by a fully qualified source. Although around 75% of the operators nationally published data to BODS in 2025 this data is not sufficiently populated for operational use in TransportAPI, see: https://www.transportapi.com/blog/2024/12/deprecation-of-tapi-bus-fares/)

However, the Bus Fares Managed Service is available for use with local or operator specific data. This managed service requires access to a source of Netex data produced by an operator or local authority.

The Bus Fares Managed service provides a point-to-point simple fare endpoint to enable a fare lookup. This source can be used to add multi-operator fares to the Journey Planner Managed Service results or to carry out fare competition comparisons to be pushed into enterprise analytics services. This service has been integrated with the Sojo multimodal journey planner service (see screenshot on p35).

TransportAPI requests for a point-to-point fare return JSON response tagged with operator, route and listing all applicable products (singles, returns and passes), concessions and journey types. The query URL to the bus fares managed service, filtered by operator, route and product name functions as a deep link to every individual fare and pass that is available in the source Netex. This deep link can be used to link into the retail offer or used to indicate fare options to the customer.


```
{
      "from": {
        "atcocode": "2400A008460A",
        "bus_stop": {
          "live": { "id": "http://transportapi.com/v3/uk/bus/stop/2400A008460A/live.json" },
          "timetable": { "id": "http://transportapi.com/v3/uk/bus/stop/2400A008460A/timetable.json" }
        },
        "place": { "id": "http://transportapi.com/v3/uk/places.json?query=2400A008460A&type=bus_stop" }
      },
      "to": {
        "atcocode": "2400102559",
        "bus_stop": {
          "live": { "id": "http://transportapi.com/v3/uk/bus/stop/2400102559/live.json" },
          "timetable": { "id": "http://transportapi.com/v3/uk/bus/stop/2400102559/timetable.json" }
        },
        "place": { "id": "http://transportapi.com/v3/uk/places.json?query=2400102559&type=bus_stop" }
      },
      "service": {
        "id": "http://transportapi.com/v3/uk/bus/services/GOCH:S14.json",
        "line_name": "S14",
        "operator": { "code": "GOCH" }
      },
      "concession": "fxc:none",
      "price": "5.0",
      "trip_type": "single"
          "product_type":null,
          "duration":null
    },
```


data from TXC + NaPTAN
hyperlinks to other APIs
data from NeTEx source

In future functionality demonstrated in TransportAPI Labs we have populated a fare zone stop membership database to define zones in stop terms as these are the explicit points establishing an entitlement to board a bus. Maps can be drawn showing the outer limits of a set of stops by using the convex hull and stops that do not form part of the zone can be highlighted within it.

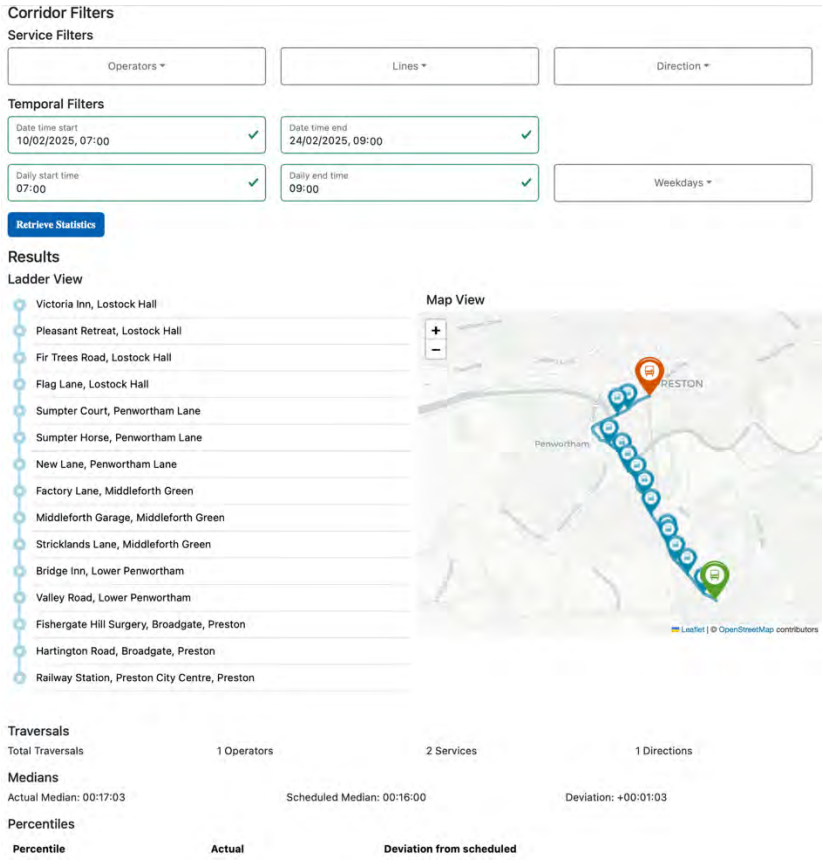
## 4.5 Bus Performance Managed Service

TransportAPI offers access to the Bus performance managed service, which archives all bus movements from SIRI VM sources fully cross-referenced to current schedules.

Performance can be measured at:

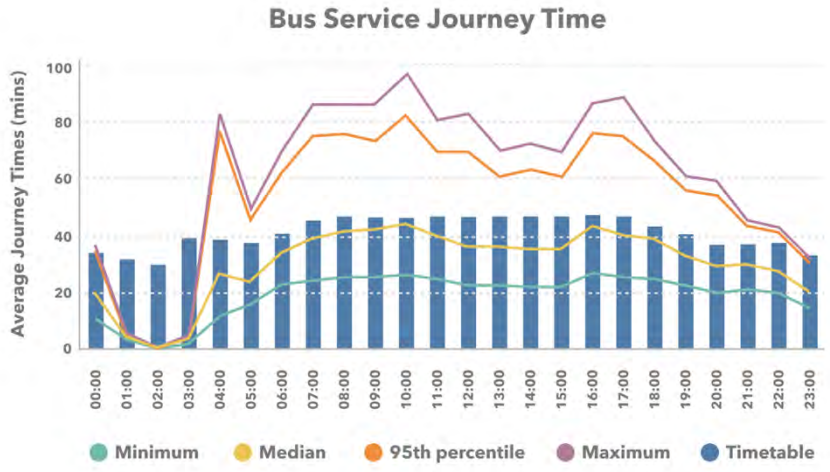
- Stops for punctuality and dwell time whether a timing point or not
- Route stop-to-stop segments referred to as edges for actual traversal times
- Full journey records including actual end to end journey times based on tracking data

The processing of this data to produce reliability statistics for both customer services and their competitors will provide a source of additional confidence in journey planning results for the user, and competitive analysis for other operators. This information also facilitates challenge to regulatory interventions by providing access to the raw data for bus journeys.



TransportAPI offer the Bus Performance Corridor Explorer web application with the Bus Performance Managed Service (see https://performance.bus.transportapi.com).

The production of bus performance indicators behind an API also allows the forecasting of real journey times for customers based upon the most recent or the typical journeys taken by buses. This information can be shared with app and web users to right-size their estimates of travel times for any given point to point journey.




## 4.6 Journey Planning Managed Service

The TransportAPI Journey Planner Managed Service offers unprecedented access to journey planning sources.

### 4.6.1 Journey planning in TransportAPI

The TransportAPI Journey Planner Managed Service aggregates a range of journey planners into a single source accessed by common requests and delivering to a common JSON response structure. The TransportAPI Journey Planner Managed Service currently provides integrated access to four journey planning engines including TfL (Mentz), National PTI SW England service (Mentz), the Great Britain Silverrail service and the customisable TransportAPI hosted Open Trip Planner. It is therefore possible to submit the same request to each of the engines by changing a single endpoint query string parameter in the public_journey endpoint. This allows the best engine to be used for each purpose.

TransportAPI adds a few hundred milliseconds to the delivery of the public_journey output when the underlying engines take between 3 and 5 seconds to respond depending on the complexity of the query. However, TransportAPI enriches the journey planning delivery in the following ways:

- The standard request structure allows comparison of journeys across different providers
- Changes in the underlying engines are managed by TransportAPI and require no changes in application logic in apps/ on web;
- All stop, route and operator national references are included in the response in standard form to enable linkage between the journey plan and other TransportAPI managed services
- Public_journey offers journey leg next timetable departures in case a connection is missed

TransportAPI can be used to undertake bulk access to journey planning for travel time analysis using real journeys rather than isochrones.

TransportAPI offers journey planning that is primarily based on Silverrail's multimodal journey planner covering Great Britain, reproducing a solution that we currently operate for the First Group website and Travel South Yorkshire (by SYMCA). Silverrail's journey planner operates two of the largest UK journey planner sites at National Rail Enquiries and Traveline.info and is recognised as the best in its class for UK transport.

For London journey plans we automatically supply journey plans from the TfL Journey Planner by Mentz, which is available as an open data service to whitelisted services such as TransportAPI. The TfL option provides the earliest access to alternative routes when London disruption closes stations, stops or routes.

TransportAPI is also offering a tight integration with the Google journey planner at scale for First Bus and Heathrow. However, note that Google's journey planning is an internationalised product that lacks some UK localisation, for example, in its representation of coach services and its failure to return the key national references needed to establish a linkage with related data such as RTI. TransportAPI has developed in-depth expertise in Google journey planner integration where this service is required by corporate policies.

In our experience the TransportAPI public_journey service in the Journey Planner Managed Service is preferable to the Google Directions API, which is being deprecated in favour of the new Routes API.

### 4.6.2 TransportAPI support for rich journey planning experience

Delivery of the rich journey planning experience that customers expect involves the integration of a number of data sources in a real-time aggregation process from TransportAPI.

Firstly, the user experience must help the customer choose the origin and destination points for the journey. This can be done via a map presented in the user experience, designed with appropriate data layers such as points of interest and congestion.

The TransportAPI Journey Planning Managed Service can be queried directly by public transport stop, station, pier or postcode, which means that a separate location look up is not required, saving time and bandwidth for the user. Where a geolocation service provides latitude/longitude values or

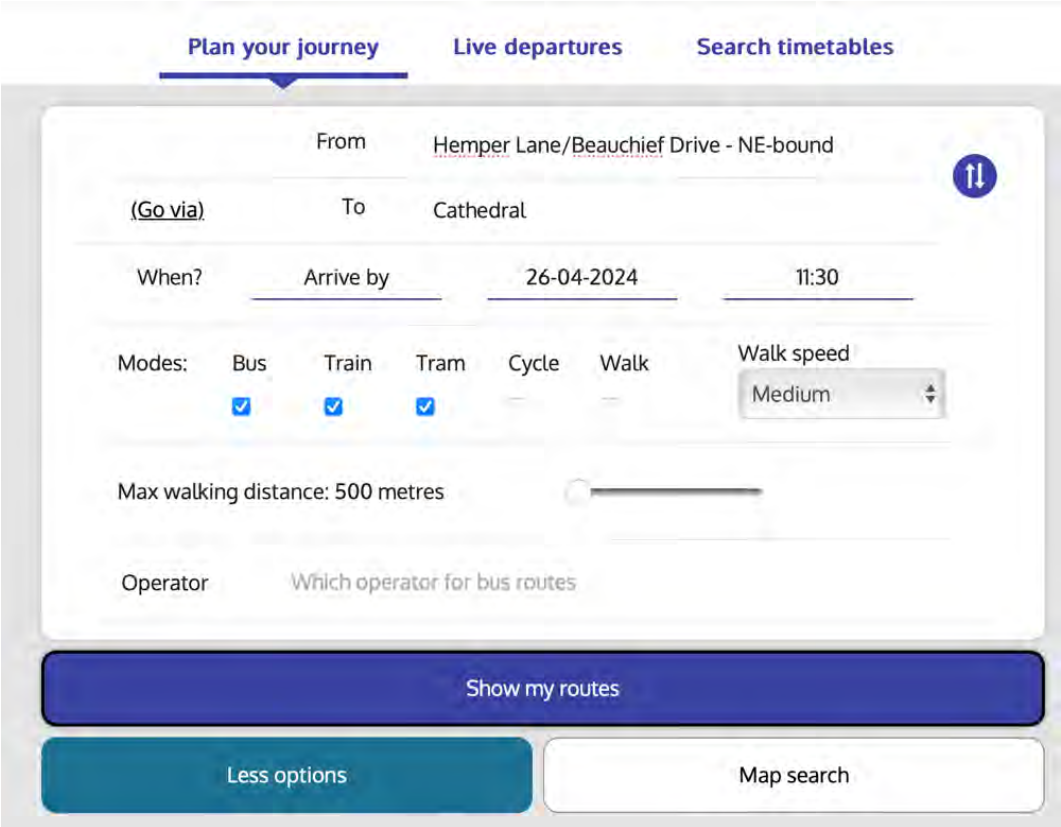
if a user enters and address or postcode the TransportAPI Places Managed Service can be called to return lists of nearby stops/ stations or those meeting semantic search criteria.

Secondly, once the origin and destination are available the core journey plan must be submitted to the best journey planning engine for any given journey. Each journey planner offers specific advantages in coverage or search options. Through public_journey it is possible to switch requests between these planners by a single keyword change, and thus each planner can be used where appropriate or as failover for each other. TransportAPI can switch automatically between Silverrail and the Mentz TfL service in London for journey planning delivery.

TransportAPI provides the journey planner on the First Bus website at:
https://www.firstbus.co.uk/planning-your-journey/journey-planner/
This journey planner permits the user to select the modes of transport they wish to use in their journey and time and date after which they wish to leave from the origin. A number of configurable features of the journey planner are defaulted in the First Bus implementation, however additional features are available, including:

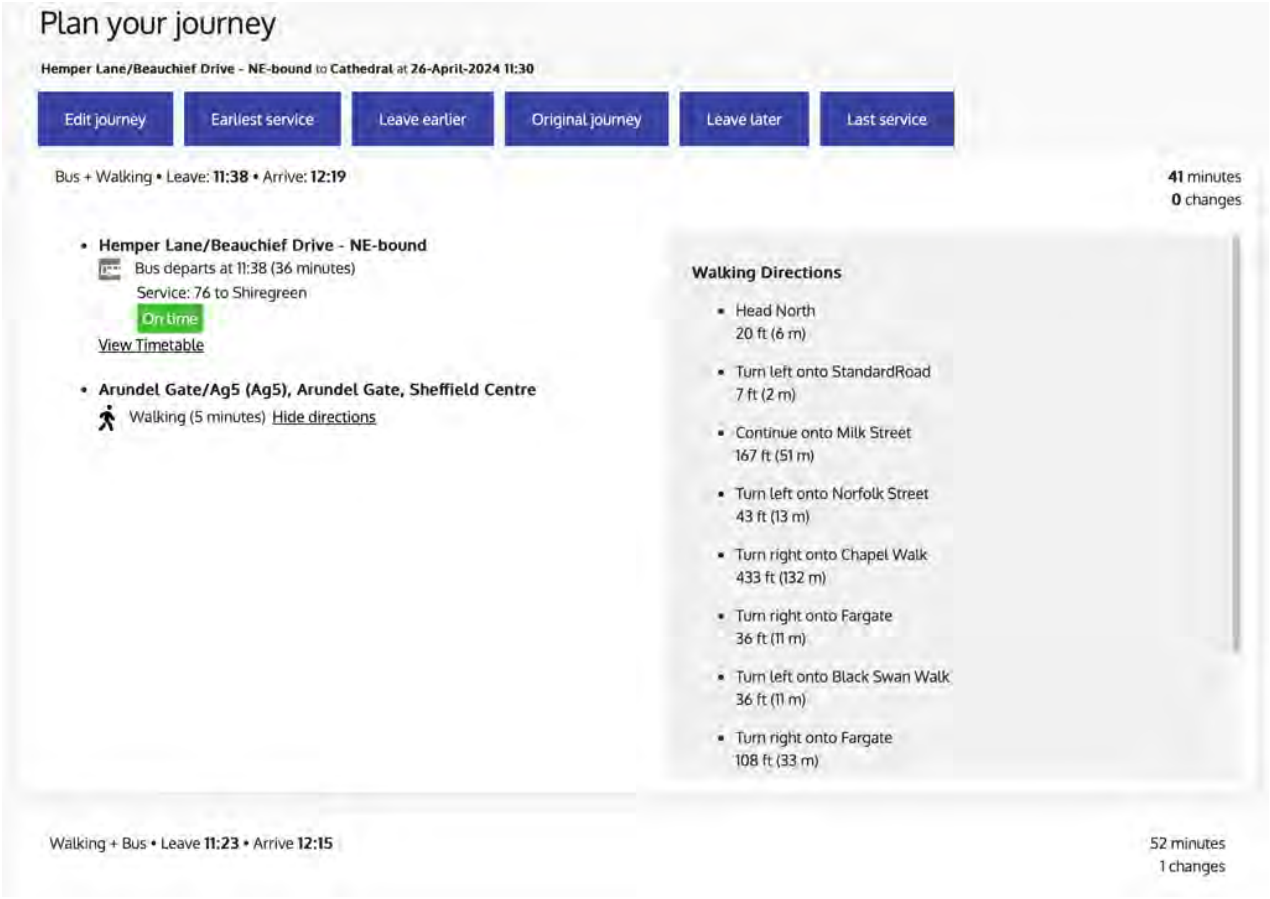
- Maximum walking distance
- Walking speed
- Minimum transfer time
- Fewer transfers

Walking controls are shown in the Travel South Yorkshire implementation powered by TransportAPI:

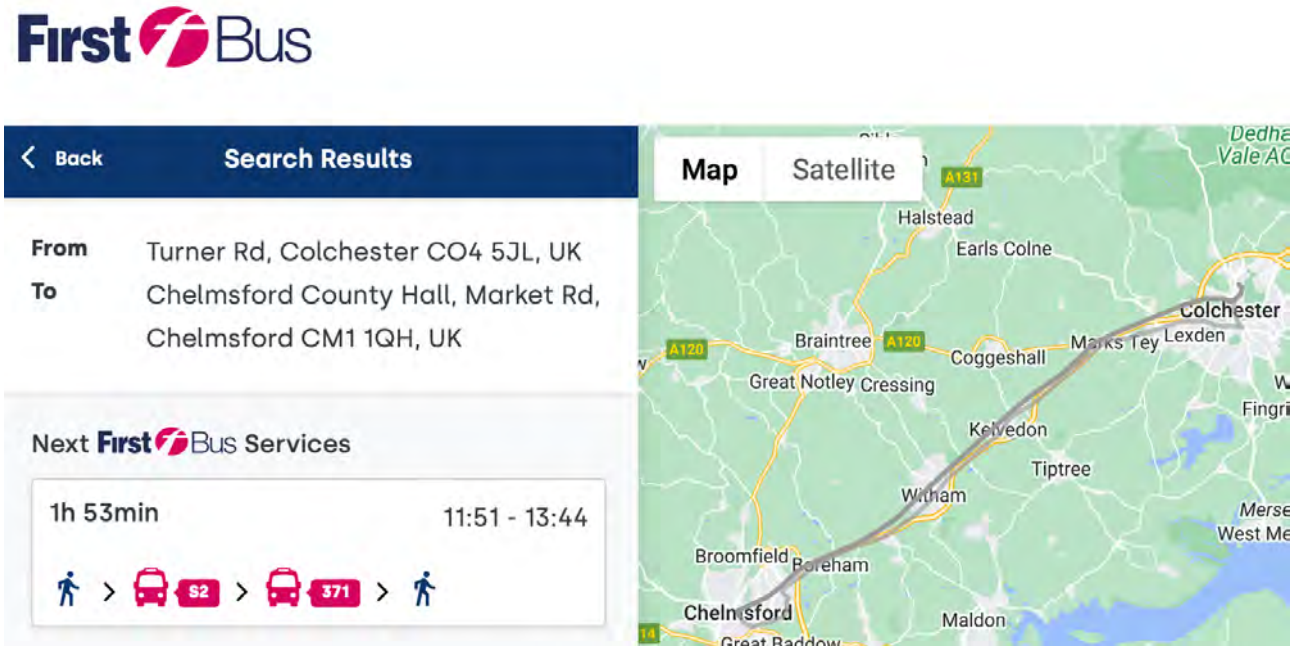


Thirdly, the journey response can be augmented by data of various kinds to enhance the options for customers. The current TSY implementation presents a map and journey schematic, however, additional elements can be provided from the current managed service (as in Travel South Yorkshire) such as:

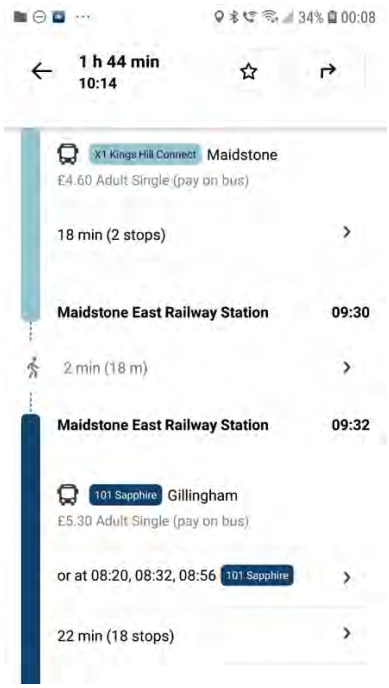
- Subsequent departures on any given journey leg
- Known delays for current journey plans
- Walking directions
- First and last services




Fourthly, the journey plans returned may be re-ordered according to business rules like operator preference for their own providers or eliminated if cancellations render the journey plan infeasible. For national bus and coach operators like First, Stagecoach or Arriva, choices also need to be made about how to show bus and coach in competition with train. TransportAPI implements operator preference in results ordering for the First Bus website journey planner as shown below:




Fifthly, the journey plan legs can be decorated with point-to-point fares where this data is available. TransportAPI powers this functionality within the Sojo multimodal journey planning app from TransportAPI Bus Fares Managed Service, where the bus stop identifiers provided in the journey planner response are used as an input to a point-to-point fare look up for the applicable traveler type.




Finally, the journey plan can be linked to live departures, route geometry and disruption to enhance the results. As the TransportAPI Journey planner managed service response delivers all nationally standardised stop, station, operator and route references, where the vehicle providing the journey

leg is already running, the UX can be decorated with live status, disruption and cancellations from the TransportAPI Bus and Rail Information Managed Services where appropriate.

Implementing this vision involves treating the journey plans and their constituent legs as rich scenarios that can be evaluated and presented as appropriate for the customer's requirements. The range of options available for personalisation to the customer depend upon the attributes that are stored in the customer record, and this will depend upon the user consents and GDPR constraints.

### 4.6.3 Owning the journey plan through to fulfilment

A journey plan should be more than a set of schedules that might inter-connect. By validating data sources and aggregating a range of information to support the customer's decision making, a journey plan can become a validated plan of campaign with alternatives and fallback positions. This allows the operator to offer a comprehensive fulfillment solution to match customer expectations.

One key benefit of Account-based ticketing (given GDPR consent) is that it is possible to follow a selected journey as the plan executes e.g. by sending links to departure boards as the journey progresses, giving bus station bay numbers or re-calculating the plan when a connection becomes infeasible or is cancelled. Owning the fulfillment allows the development of a richer customer relationship by continuing to offer support during the journey. It also allows the execution to be scored and archived to feed back into the business rules for the service.

With the addition of pick up/ drop off zone geometries to the TransportAPI stop database accessed by the Places endpoint, Demand Responsive Transport can also be added to journey planning origin/ destination options.

## 4.7 Rail Information Managed Service

The TransportAPI Rail Information Managed Service offers access to the schedules and live rail data for services in Great Britain. There are two principal organisations involved in producing rail data:

- Network Rail with its Corpus, ITPS & TRUST services (published through National Rail Open Data -NROD)
- Rail Delivery Group with its DWS, Pushport & RSP services (published through Darwin)

The Network Rail CORPUS service holds all the rail infrastructure details including station locations and references to which the schedules and live data are linked. The ITPS service is the master timetabling service for the GB rail network and is the source of all rail schedules. Rail schedules are updated nationally in May and December of each year, however, changes are made every night (Short Term Plan- STP) and during every day (Very Short Term Plan- VSTP). The TransportAPI Rail Information Managed Service provides access to rail schedules from ITPS, and Heathrow Express use this data in their timetables on their website (see below):

transportapi

From Heathrow Airport | To London Paddington | Date of travel Today, 30 Jan

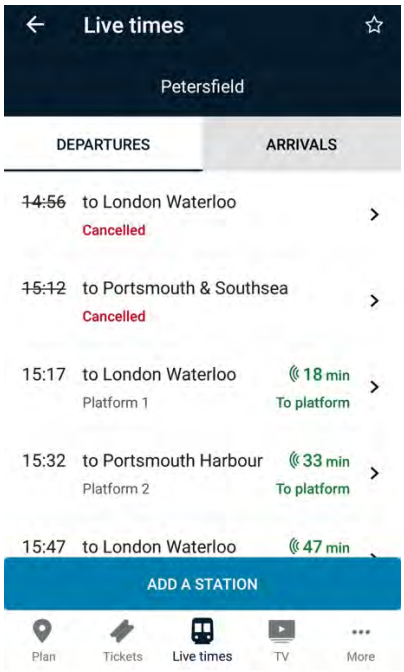
| Departs Heathrow Airport Terminal 5 | | Departs Heathrow Airport Terminal 2 & 3 | | Arrives London Paddington |
|---|---|---|---|---|
| 07:42 | → | 07:47 | → | 08:03 |
| 07:57 | → | 08:02 | → | 08:19 |
| 08:12 | → | 08:17 | → | 08:32 |
| 08:27 | → | 08:32 | → | 08:47 |
| 08:42 | → | 08:47 | → | 09:03 |
| 08:57 | → | 09:02 | → | 09:17 |
| 09:12 | → | 09:17 | → | 09:32 |
| 09:27 | → | 09:32 | → | 09:47 |
| 09:42 | → | 09:47 | → | 10:02 |
| 09:57 | → | 10:02 | → | 10:17 |

Real-time train running data is produced by both NROD and Darwin and TransportAPI gives access to both sources. TRUST from NROD is a service designed to record train operations and so live running data is not always reported in real time. However, the schedules and the live data are updated by STP and VSTP data. TransportAPI retrieves live running data from Darwin through DWS (for Rail Information Managed Service) and Pushport (for Rail Performance Managed Service).

On TransportAPI we have engineered access to both the Network Rail and National Rail ecosystems with as similar API response as possible.

- TRUST station live departures (train/station_timetables) with no source configuration parameter
- DWS station live departures (train/station_timetables) includes the source configuration parameter

For passenger information purposes developers should always use the Darwin source WITH THE SOURCE CONFIGURATION parameter in the requests. TransportAPI delivers live data to all the First Group Train Operating Companies using the Darwin DWS service as in the SWR app shown below:



## 4.8 Rail Performance Managed Service

TransportAPI has built a national scale rail service archiving infrastructure to power the TransportAPI Rail performance managed service. This data is available at API for apps, screens and analytics to consume and display, and it is also available to browse in our Rail Performance Explorer web application (see below).

Data is available from the TransportAPI Rail Performance Managed Service by using the following endpoints:

- Station_actual_journeys: returns a historic departure board from a specified day within the last 90 days, including all the scheduled departures and their live status at the specified 'point of view time' for a given time range.
- Actual_journeys: returns the recorded progress (or cancellation) of individual services according to various filters by rail referencing including train_uid, RID, headcode and RSID

This data can be browsed in the Rail Performance Explorer (https://rail-performance.transportapi.com):