

Annex B – TI API Technical Specification

Contents

Consultation Question.....	1
Introduction	1
TI API Technical Specification	2
Updates to the May 2026 baseline	2
Splitting tariff pricing data into static and dynamic rate types (CHG-015)	2
Regional identifiers for gas tariffs (CHG-018)	4
Usage-band fields for block tariffs (CHG-019)	5
Governance	6
Appendix A: Technical background and navigation guide	7

Consultation Question

Question B1: Do any of the proposed changes to the TI API Technical Specification, baselined in the May 2026 Government Response to the November 2025 consultation, introduce any defects, inconsistencies or blockers that could prevent the successful implementation of TI Phase 1a? If yes, please describe the issue and the change needed.

Introduction

1. This annex identifies the proposed TI API Technical Specification for Phase 1a. The specification defines the physical API standard, including the permitted API error response codes. The TI API Technical Specification will be used by TI Energy Suppliers and TI Users to exchange market messages and data items set out in Annex C.
2. The TI API Technical Specification will primarily be used by technical experts, such as developers and software architects, to build and maintain the TI API. Reviewers with this skill set should focus their review on this artefact.
3. In addition to describing the proposed changes to the TI API Technical Specification, that was baselined in the May 2026 Government Response to the November 2025

consultation ('May 2026 baseline'), we describe the governance arrangements for the TI API Technical Specification once it is implemented into the REC, and Appendix A sets out how the TI API Technical Specification is hosted and a navigation guide.

TI API Technical Specification

4. The proposed TI API Technical Specification for TI Phase 1a is an OpenAPI specification hosted on the REC Developer Portal and can be found at the following link:

[APIs | Developer Portal](#)

Updates to the May 2026 baseline

5. As noted above, the TI API Technical Specification for TI Phase 1a was baselined in May 2026. The following enhancements have been made to this baseline following industry engagement, detailed technical review and pilot testing activity with industry volunteers:
 - a. a split of tariff price information into two clearly defined types — fixed-schedule (static) and time-varying (dynamic). This will allow systems to automatically check whether tariff data is valid. (*CHG-015*);
 - b. Revised conditionality for regional identifiers. Electricity tariffs continue to use GSP Group Id. For gas tariffs, GSP Group Id or LDZ Region is populated only where the Supplier uses that identifier to distinguish regional variants. (*CHG-018*);
 - c. a clear rule that the usage-band fields (*min_kWh*, *max_kWh*) apply only to block tariffs and are otherwise omitted (*CHG-019*).
 - d. a minor update to the description of *sellable_to* so that this remains aligned to the EMDS. An update has been made to data item 'DIYYY19 – Sellable To' in the EMDS and this is described further in Annex C rather than in this annex.
6. The changes noted in paragraph 5a to c are described in more detail below.

Splitting tariff pricing data into static and dynamic rate types (*CHG-015*)

7. This change is proposed in response to industry feedback, raised at the second Implementation Working Group (IWG) and discussed through direct participant engagement following publication of the May 2026 baseline. Concerns were raised that the specification described tariff pricing data ('rates') using a single general-purpose

structure, when in practice it takes different shapes for static and dynamic tariffs (the 'polymorphism' issue).¹

8. The proposal to split rates into two clearly defined types (*rate_static* and *rate_dynamic*) means that tariff pricing data can be validated automatically against the specification itself.
9. This change does not amend the payload returned to TI Users i.e., the same rates object is returned, with the same fields, populated according to the same logic. However, the change enforces the existing rules in the machine-readable YAML, rather than in prose, so that systems can automatically check that the data supplied is valid.
10. Suppliers that have started building against the current baseline will need to adjust how rate data is populated. In return, tariff data can be validated automatically, preventing technically valid but semantically broken payloads e.g. populating both *time_from* (associated with a static tariff) and *valid_from* (associated with a dynamic tariff) on the same rate.
11. This change has been reflected in an update to the description of the 'rates' object and *rate_static* and *rate_dynamic* schema within the YAML/API Technical Specification. See updated extract below:

```
rates:
  type: array
  description: >
    List of rate segments. Each segment must conform to exactly one of
    two structural variants, discriminated by which effectivity fields
    are present: rate_static (time_from / time_to, used where
    tariff_type is static) or rate_dynamic (valid_from / valid_to,
    used where tariff_type is dynamic). Fields belonging to the other
    variant must not be included in the segment (rather than being
    supplied as null) - this is enforced structurally via oneOf with
    additionalProperties set to false on each variant.
  items:
    oneOf:
      - $ref: '#/components/schemas/rate_static'
      - $ref: '#/components/schemas/rate_dynamic'
```

¹Static tariffs are tariffs where the unit price applicable at a specific time of day does not change over time. Dynamic tariffs are tariffs where the unit price applicable at a specific time of day changes over time. (See Annex C, Energy Market Data Specification, *DIYYY20 – Tariff Type*).

```
rate_static:
  type: object
  description: >-
    Rate segment variant for static tariffs (tariff_type = static). Time
    of day effectivity is expressed via time_from / time_to. The dynamic
    rate fields valid_from / valid_to must not be included on this
    variant - additionalProperties is set to false so their presence
    fails validation, rather than relying on prose convention.
  additionalProperties: false
  required:
    - type_of_use
    - tariff_days
    - time_from
    - time_to
    - unit_price
```

```
rate_dynamic:
  type: object
  description: >-
    Rate segment variant for dynamic tariffs (tariff_type = dynamic).
    Effectivity is expressed via valid_from / valid_to datetimes. The
    static rate fields time_from / time_to must not be included on this
    variant - additionalProperties is set to false so their presence
    fails validation, rather than relying on prose convention.
  additionalProperties: false
  required:
    - type_of_use
    - valid_from
    - valid_to
    - unit_price
```

Regional identifiers for gas tariffs (CHG-018)

12. This change amends the geographic region code applicable to gas tariffs. Under the May 2026 baseline, LDZ Region was required for all gas tariffs. Supplier engagement through testing has confirmed that gas tariff regionalisation is not uniform: Suppliers may use data item DI50028 – GSP Group Id or DIYYY24 – LDZ Region, depending on their existing pricing methodology.
13. The proposed change is to allow TI Energy Suppliers to differentiate their gas tariffs using GSP Group Id or LDZ Region, depending on the TI Energy Supplier's existing pricing methodology. Where GSP Group Id is used to distinguish gas tariff variants, *gsp_group_id* must be populated. Where LDZ Region is used, *ldz_region* must be populated. Where a gas tariff is not regionally differentiated, neither field is required. The TI Arrangements do not require Suppliers to change how they regionalise or price their tariffs.
14. The change does not add, remove or rename an API endpoint or payload field and does not alter the JSON structure. It does update the OpenAPI/YAML descriptions and the associated EMDS conditionality so that GSP Group Id and LDZ Region are populated consistently with the supplier's regional pricing methodology. This is shown in the

extract from the API Technical Specification descriptions below. This conditionality is also reinforced by changes to the conditionality rules in the EMDS for when these data items are used in the MMXXX5 and MMXXX7 (see Annex C).

```
gsp_group_id:
  type: string
  description: >-
    to be populated for electricity tariffs. To be populated when used to differentiate gas tariffs.
```

```
ldz_region:
  type: string
  description: |
    to be populated when used to differentiate gas tariffs
```

15. We have also taken the opportunity to correct a discrepancy in the use of the LDZ Region YAML Name. This was applied inconsistently as either *ldz_region* or *ldz_regions*. As only one value (enumeration) can be used in the market messages that contain this data field, we have updated to use the YAML Name *ldz_region*. This clarification has also been applied to the EMDS (see Annex C).

Usage-band fields for block tariffs (CHG-019)

16. This change clarifies when the usage-band fields should be populated for block tariffs. Example payloads produced during testing showed block tariff fields being applied inconsistently. Updates have been made to the field descriptions (see API Technical Specification screenshots below) so that they now state explicitly that *min_kWh* (DIYYY48 – Tariff from kWh) and *max_kWh* (DIYYY49 – Tariff to kWh) are only used for block tariffs and tariffs where the unit price changes once a defined amount of energy has been used, and are omitted for all other tariff types. For *max_kWh*, the field may also be omitted on the final block of a block tariff where there is no upper limit. There is no structural change to the specification; only the field descriptions are updated. This has also been updated in the EMDS as described in Annex C.

```
min_kWh:
  type: string
  description: >-
    indicates the energy usage (kWh) from which the unit rate in this rate
    segment applies. Populated only where the rate forms part of a block
    tariff, where the unit rate changes once a defined amount of energy has
    been used. Omit this field for all other tariff types.
```

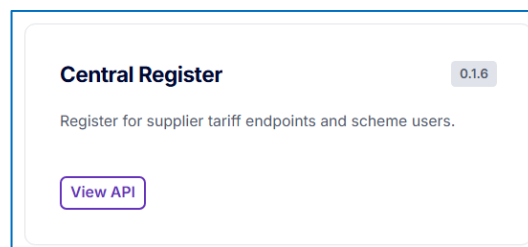
```
max_kWh:
  type: string
  description: >-
    indicates the energy usage (kWh) up to which the unit rate in this rate
    segment applies. Populated only where the rate forms part of a block
    tariff, where the unit rate changes once a defined amount of energy has
    been used. Omit this field for all other tariff types. May also be
    omitted on the final block of a block tariff where there is no upper
    limit.
```

Governance

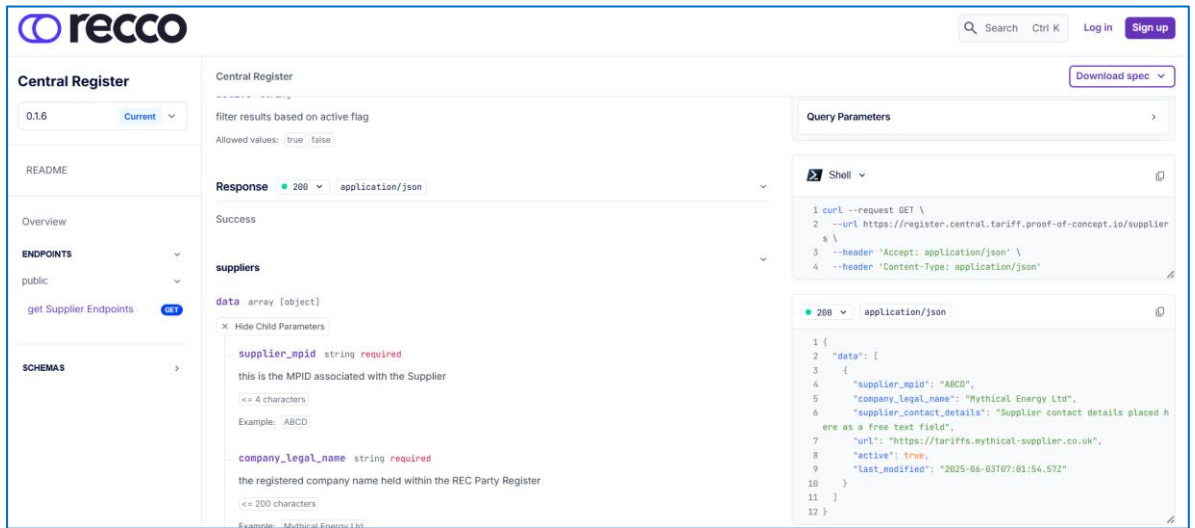
17. The TI API Technical Specification is proposed to be a Category 3 governance document within the REC. As set out in [REC Schedule 5: Change Management](#) (Section 29), this means that RECCo, as the 'Responsible Provider' will raise changes, consult with impacted parties where required, and determine if an update should be made.
18. The TI API Technical Specification references data items and market messages. These are part of the REC Data Item Catalogue and Market Message Catalogue which form part of the Energy Market Data Specification (EMDS). As these catalogues are REC Category 2 products, updates must be progressed via the standard REC change process set out in REC Schedule 5 (Section 7 to 22). Further information on the changes to the EMDS required to support TI Phase 1a is included in Annex C.
19. As the TI API Technical Specification is proposed as a Category 3 Product, RECCo will be responsible for maintaining the API specification and for keeping it aligned with the EMDS.
20. Implementation of the changes set out within this Annex will be delivered through the Secretary of State's section 245 Energy Act powers. Following implementation, any subsequent changes to the TI API Technical Specification will, unless the SoS decides to amend this document further, follow the standard Category 3 change process to enable delivery of change to support lower-level technical design, within a shorter timescale.

Appendix A: Technical background and navigation guide

21. The REC Developer Portal presents the TI API Technical Specification as an interactive web page and can be viewed without users needing to log in. Readers can browse each API endpoint (for example "get tariffs" and "get tariff details") with a plain description of what it does, the parameters it accepts and the possible values for each, alongside a working example of the request and an example of the data returned. The portal is subject to version control, and the version being issued for consultation is v0.2. The complete specification can be downloaded (in both YAML and JSON formats) in machine-readable form via the "Download spec" button.
22. An OpenAPI specification is an industry-standard way of describing an API in a single structured file (published in YAML or JSON format). It defines, precisely and completely, what requests can be made, what information must be provided, exactly what data comes back, and in what format. Because the format is standardised, it serves two audiences at once: people can read it as the formatted web pages, and computer systems can use the same file directly, to generate code, test an implementation, or automatically check that the data a supplier sends matches the rules. The pages on the REC Developer Portal are generated directly from the OpenAPI file, so the documentation and the specification are fully aligned.
23. Navigating to the Supplier Register (all screenshots are illustrative; REC Developer Portal layout may change if deemed appropriate): In the REC Developer Portal landing page, select the following:

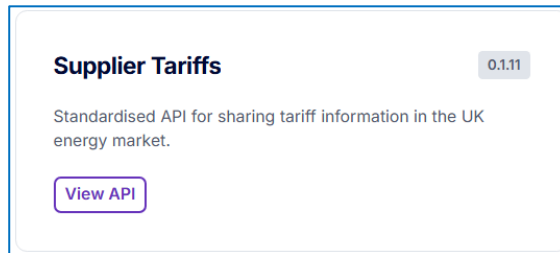


This will take the user to the part of the API Technical Specification on supplier endpoint details:



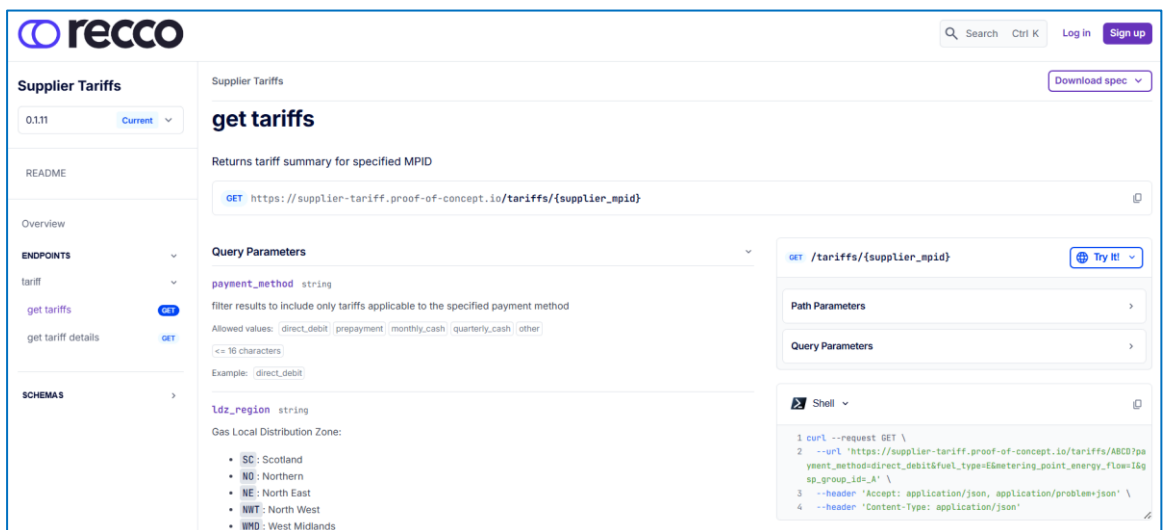
The screenshot shows the 'Central Register' API page on the recco platform. The left sidebar contains navigation links: Central Register (0.1.6), README, Overview, ENDPOINTS (public, get Supplier Endpoints), and SCHEMAS. The main content area displays the 'get Supplier Endpoints' endpoint with a 200 status and application/json content type. The response is a JSON array of objects, with an example showing a supplier with 'supplier_mpid' and 'company_legal_name'. The right sidebar shows the 'Query Parameters' section with a 'Shell' tab containing a curl command for the endpoint.

24. Navigating to the Supplier Tariffs: In the Developer Portal landing page select the following:



The card displays the 'Supplier Tariffs' API version 0.1.11. It describes it as a 'Standardised API for sharing tariff information in the UK energy market.' and includes a 'View API' button.

This will take the user to the API Technical Specification on supplier tariffs and tariff details:



The screenshot shows the 'Supplier Tariffs' API page on the recco platform. The left sidebar contains navigation links: Supplier Tariffs (0.1.11), README, Overview, ENDPOINTS (tariff, get tariffs, get tariff details), and SCHEMAS. The main content area displays the 'get tariffs' endpoint with a 200 status and application/json content type. The response is a JSON object containing a list of tariffs. The right sidebar shows the 'Query Parameters' section with a 'Shell' tab containing a curl command for the endpoint.