



Beckn Protocol

Core Working Group Meeting

19th August, 2022



Agenda



	Topic
1	Objectives and Progress Tracking
2	Action Items from last call
3	Topics selected for discussion today
4	Topic 1: Gateway search optimization (contd from last time)
5	Topic 2: Commercial Model agreements
6	Topic 3: Transaction Id as a session reference
7	Topic 4: Payments schema update : tracking multiple payments against an order
8	Topic 5: Review update / on_update API
9	Topic 6: Review of rating / on_rating API

beckn 1.x release : Objectives and progress summary



Objective	Status Update
Review ONDC RFC and extract feature requests for inclusion into core	In progress.
Review Local Retail specs and extract feature requests for inclusion into core	In progress
Review Logistics specs and extract feature requests for inclusion into core	In progress
Review DHP yaml and extract feature requests for inclusion into core	Not started yet
Review DSEP yaml extract feature requests for inclusion into core	Not started yet
Review mobility specs and extract feature requests for inclusion into core	Not started yet
Resolve issue on Gateway Optimization	In progress. Some of the issues have been discussed and points of view have been provided.
Review Registry API and proposals to spec	Not started yet. List of proposals can be viewed here
Review Group buy use case	In Progress

Action Items from last call



- ❑ Create schema for capturing additional information like people, things, and docs |

Complete

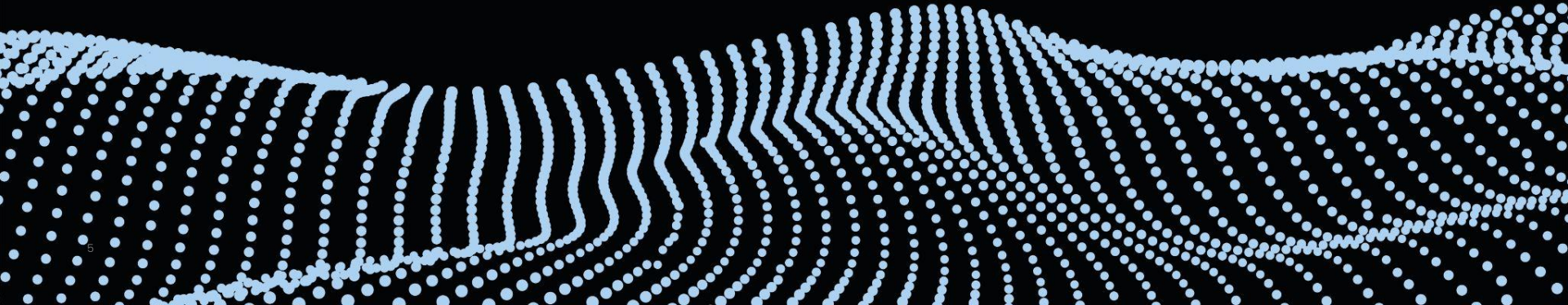
- ❑ Create PR to `draft` for updating communication protocol for Gateway search flow |

Ravi | **In Progress**

- ❑ Create PR to `draft` for adding new features to Ack schema | **Not Started**



Topics selected for discussion today



Topics selected for discussion today



- ❑ Gateway search optimization (contd from last time) | DONE
- ❑ Commercial Model agreements | DONE
- ❑ Transaction Id as a session reference | DONE
- ❑ Payments schema update : tracking multiple payments against an order
- ❑ Review update / on_update API
- ❑ Review of rating / on_rating API

Gateway Search Optimization (contd)



- ❑ Change ACK schema with `num_of_callbacks_expected` as additional fields | **Discussed**
- ❑ Gateway `on_search` needs to be sent to bap from bpp directly to avoid unnecessary hop | **Discussed**
- ❑ Add granular domains for better filtering of bpps | **Discussed**
 - ❑ BAP must send this domain while searching.
 - ❑ BPP may be registered against multiple granular domains.
- ❑ Registry of bpps to have granular area of service or operating region E.g pincode | **Pending**
 - ❑ Proposal to add a new API register/location on the registry
 - ❑ Provider location with a possible region

Commercial Model Agreements



❑ **Proposal**

- ❑ Commercial Model of ONDC - Click [here](#)
- ❑ ONDC wants transparency to buyers about settlement terms. Their comments : -
- ❑ *"ONDC commercial policy necessitates buyer apps sending the finder fee in the search request. Finder fee values will be dynamic and can change for different categories of products. How else can this be communicated over the protocol?"*

❑ **Recommendation:**

- ❑ Terms are network agreements between participants and has no need to be transmitted to buyer on every transaction
- ❑ For Transparency, The Quotation object can be used to show breakups of all kinds e.g. Paytm - 6 Rs, Network Fee - 1 Re , Seller - 112 Re
- ❑ The Network Policy can mandate certain line items to always be shown in the quote breakup

Commercial Model Agreements (contd)



❑ **Proposal**

- ❑ An example of a commercial model agreement in the “core” API is Payment “collection” endpoint being sent by the BPP to the BAP during an **on_init** API call
- ❑ This seems like a *negotiation* between the BAP and the BPP at a transaction level as BAP could want to collect payment first, and then settle to BPP, while BPP may want to collect payment directly by sending a collection endpoint. This can potentially lead to unending loops due to conflicting terms
- ❑ Should this and “other” commercial terms be negotiated “off” the transaction API set (search/select/...)

❑ **Recommendation**

- ❑ Payment.url should always be either a BPP driven value or a Network level decision.
- ❑ Network can decide who collects payments first. It can decide one of the following:
 - ❑ All payments must directly be made to BPPs, or
 - ❑ BAPs always collect first then settle to BPP, or
 - ❑ Either of them can collect, but BPPs payment terms will override BAP terms
- ❑ All commercial model agreements must take place outside the transaction APIs. There can be a different set of APIs to negotiate commercial model agreements.
- ❑ The resulting dual digitally signed agreement can be linked to transactions as a reference if full transparency is required

Registry API Spec and Proposals



❑ **Proposal**

- ❑ Should we have a /register api for onboarding purposes.
- ❑ We can review ondc/dhp/dsep onboarding process to abstract this.
- ❑ This is needed to have smoother onboarding from Accelerator/Agencies/Self
- ❑ Discussion <https://github.com/beckn/registry/discussions/15>

❑ **Recommendation**

- ❑ Agreed. We should have an /register API as part of the registry specs
- ❑ We can have a similar form-like input

❑ **Proposal**

- ❑ We should have a /register_location api to record BPPs location/region of service

❑ Recommendation

- ❑ Agreed. We should have an register_location API set as part of the registry specs

Transaction ID / Order ID / Message ID



Transaction ID

- ❑ Globally unique : Hence UUID
- ❑ Must be created post-search (select ->support)
- ❑ **Order ID : Transaction ID** can have **n:1** relationships at a BAP
- ❑ **Order ID : Transaction ID** can have **1:1** relationships at a BPP

Message ID and Transaction ID must be globally unique. Non-human-readable

Registry API Spec and Proposals



Breaking Changes:

1. Status API has order object instead of order_id
2. Track API has order object instead of order_id
3. Cancel API has order object instead of order_id
4. tracking_info added to Fulfillment object
5. Adding cancellation event inside Order object containing cancellation_reason_id, cancelled_by, cancelled_at

To discuss:

1. Should we have an event log inside the order object
2. Tracking information inside order object. Tracking url rules
3. Should we remove bpp/categories, bpp/offers, bpp/fulfillments
4. Meta APIs - should these APIs be moved to the network layer, or keep the APIs at the core layer and enumerate the values in the network layer.



Action Items

Action Items



ID	Task	Assignee	Date
1			
2			
3			
4			

Call Recording

