Pushed Client Registration

OAuth WG
IETF123, Madrid
July 25, 2025

Justin Richer
Aaron Parecki

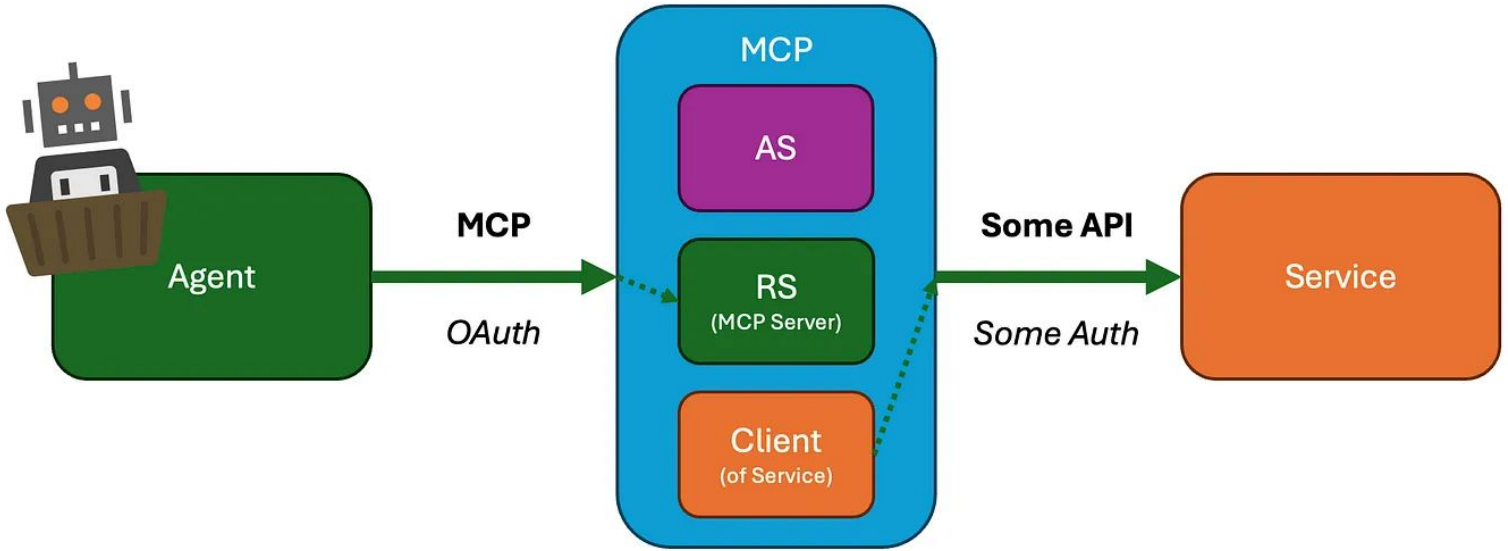
Why do we have client_id in the first place?

- OAuth 2 is built around the authorization code flow
- Start in the **front channel**, where we can’t authenticate the client
- *Identify the client*, without authentication, to kick off the process
- Later authentication is bound to that identifier

Not everything fits a pre-registered client_id model

- Ephemeral clients (SPA instances)
- Public clients (native app instances)
- Dynamic registration
  - Allows a client instance to register for a new client_id
- OIDFed
  - Allows registration to happen through a federation authority
- Client ID Metadata Document (BlueSky)
  - Allows AS to fetch a client's metadata from a URL
- SPIFFE Ids

MCP Agents: Pushing the Boundary



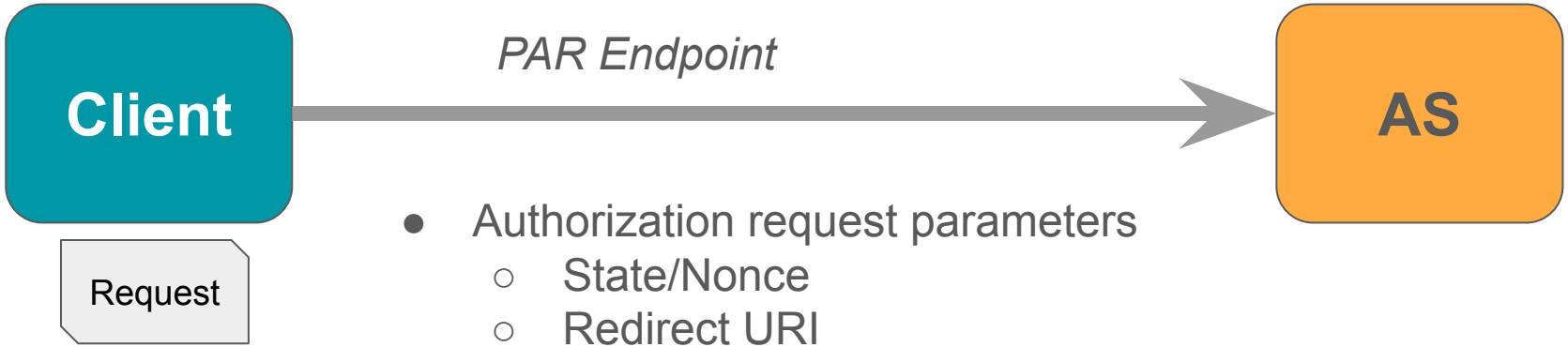
What if we could fold in client registration to the request?

- Start with pushed authorization request (PAR)
- Send all the required client metadata in that step
- Subsequent stages are traceable back to the original request
- Work around client_id required syntax with a special URI:

```
urn:ietf:oauth:parameters:dynamic
```

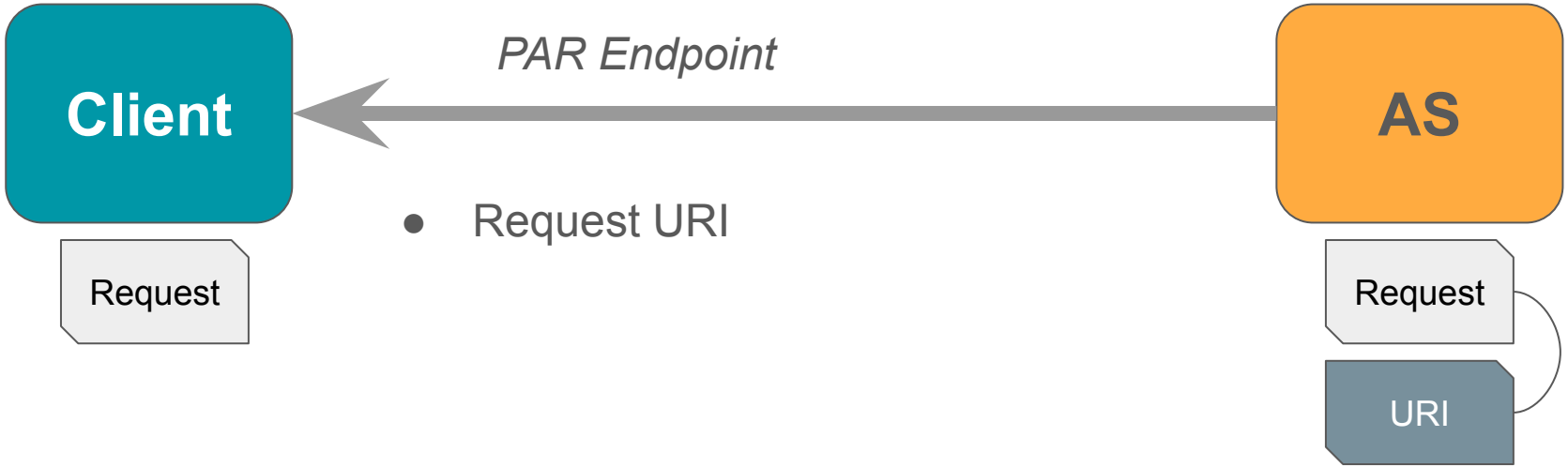
draft-richer-oauth-pushed-client-registration-00

Step 1: PAR

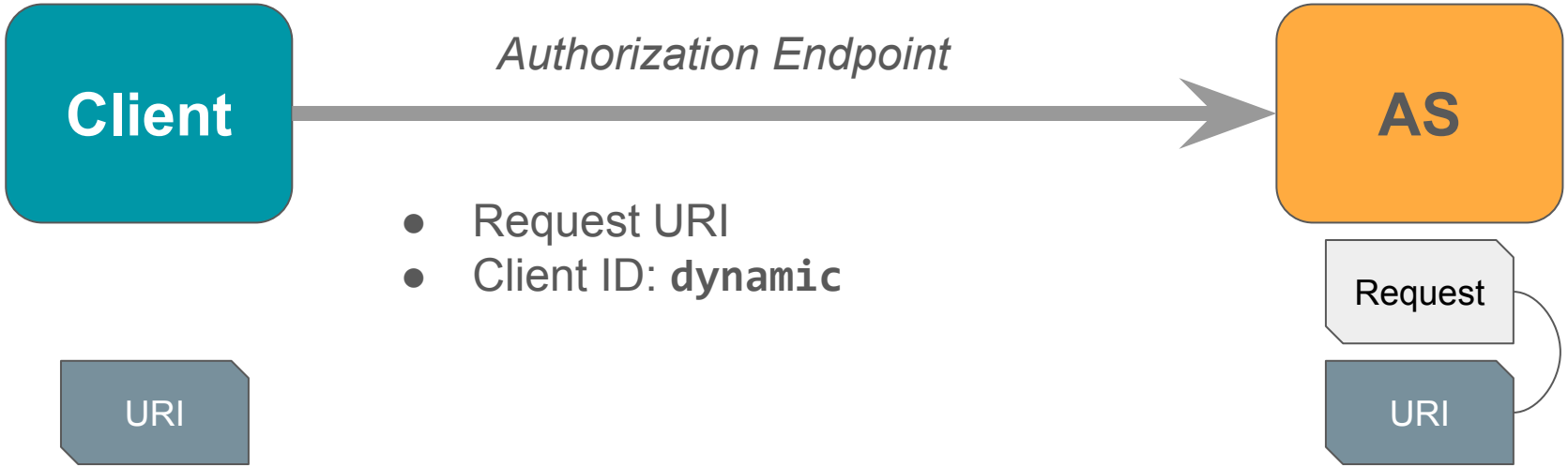


- Authorization request parameters
  - State/Nonce
  - Redirect URI
  - Scope
  - PKCE Challenge
- ClientID: **`dynamic`**
- Client metadata
  - Keys
  - Name

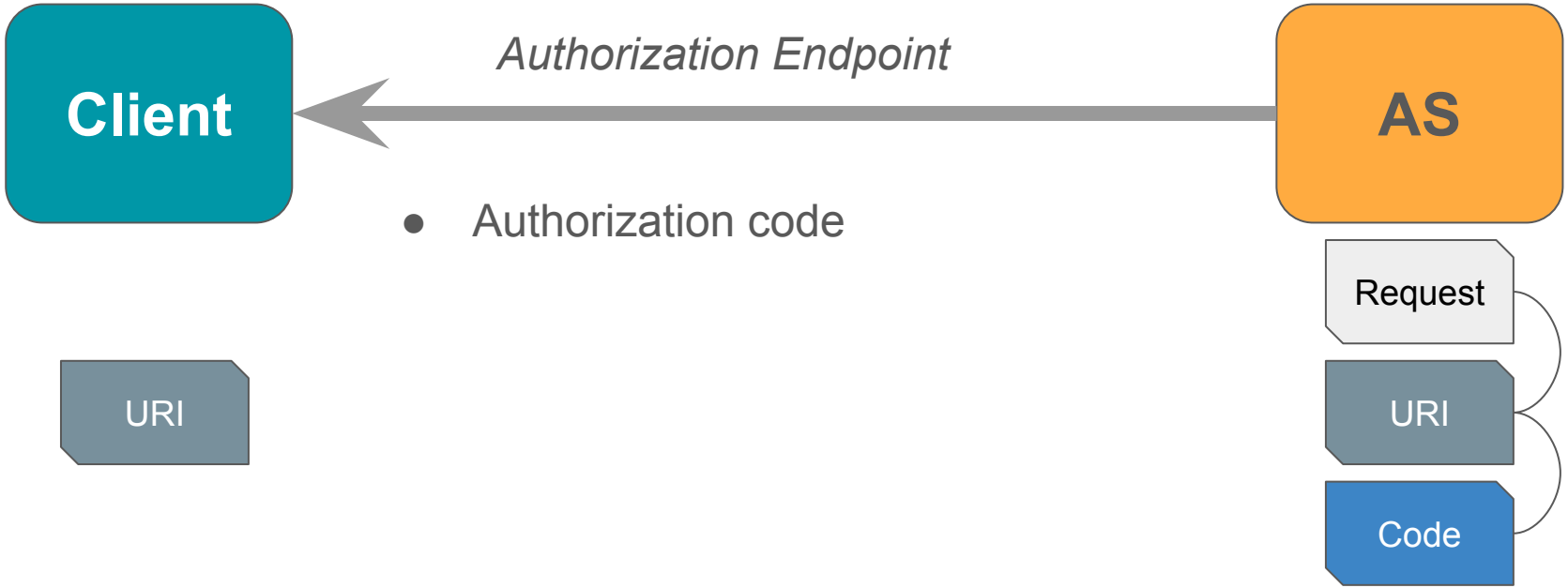
Step 2: Request URI



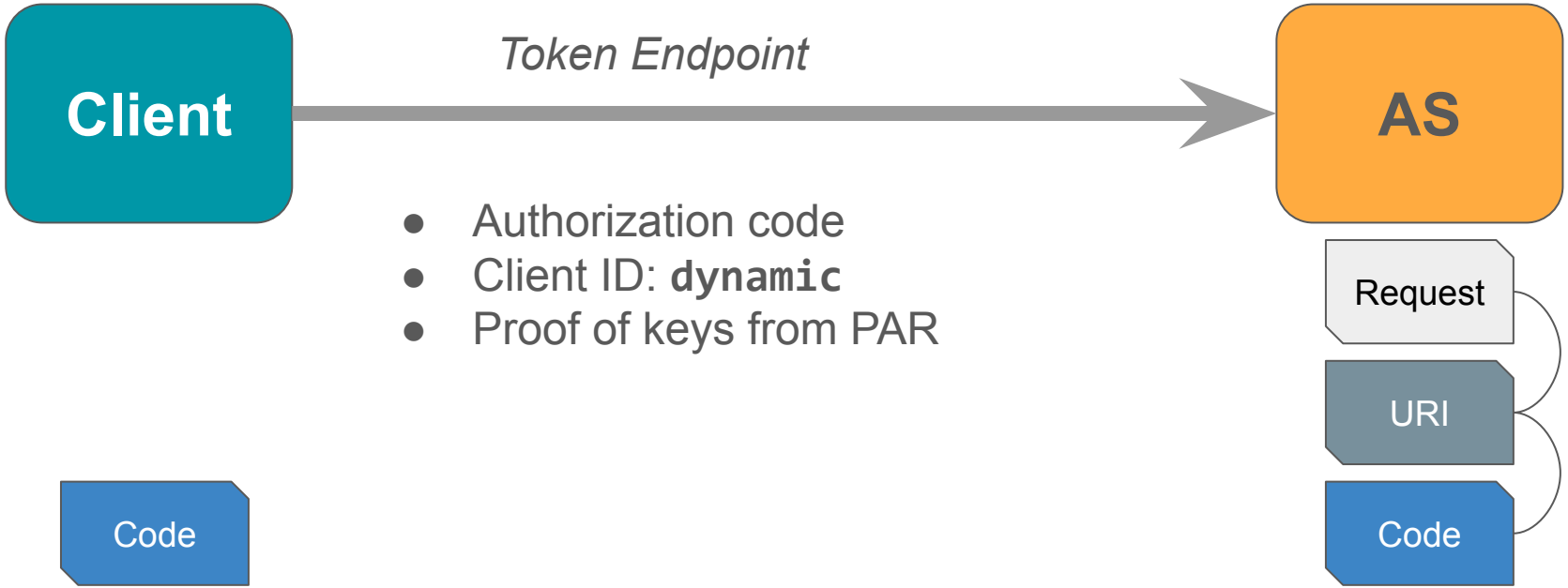
Step 3: Authorization Request



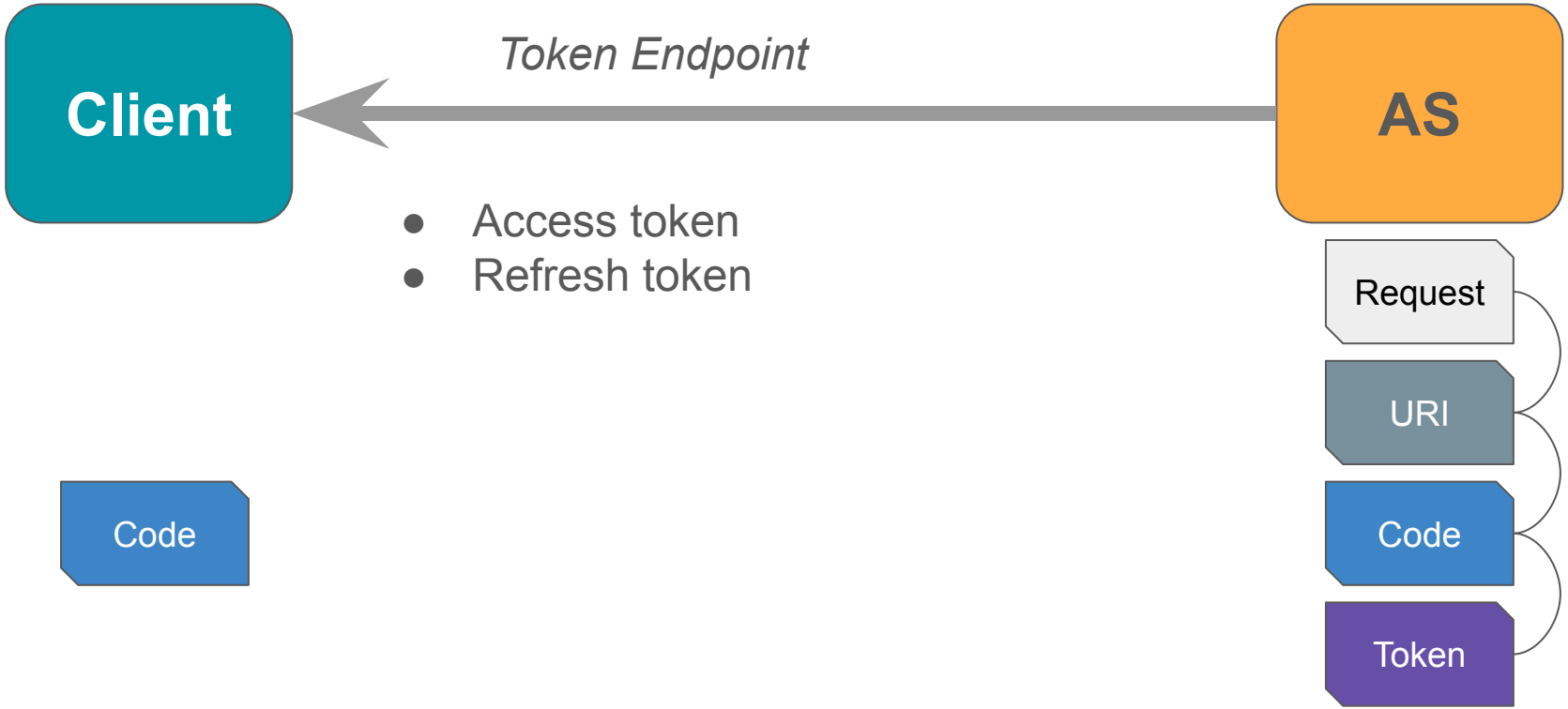
Step 4: Authorization Response



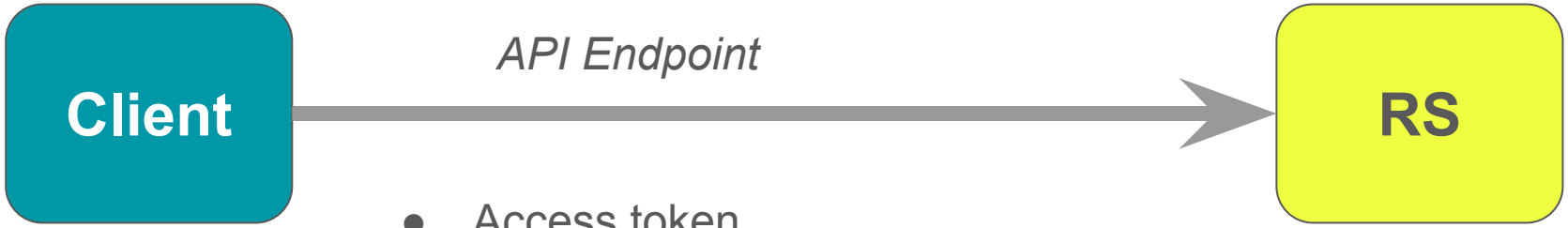
Step 5: Token Request



Step 6: Token Response



Step 7: Token Use



- Access token
- RS can dereference whatever it needs from the original request context just like a regular access token

Token

Token

Benefits of this approach

- No extra round trip for Dynamic Registration
- No fetch from the AS (with SSRF attack surface)
- No need to host metadata for clients (can be instance-specific)
- No vestigial client IDs hanging around to clean up
    - When the token goes away, the client information can go away
    - New client requests are unrelated to previous ones, and that’s a feature!

Do we want to solve this problem space?