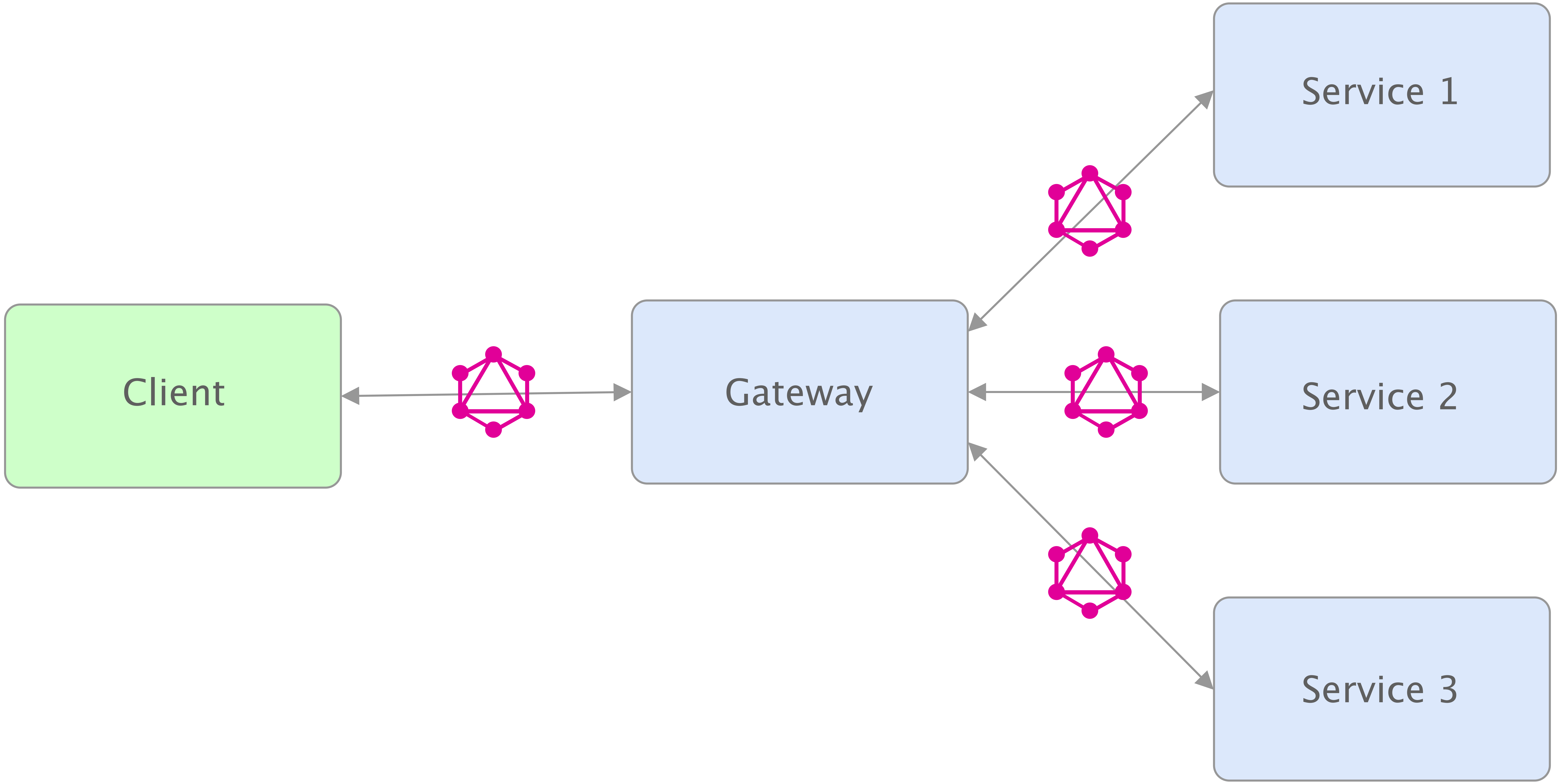
Schema Stitching

What?

Schema stitching is the process of creating a single GraphQL schema from multiple underlying GraphQL APIs.


Client
Gateway
Service 1
Service 2
Service 3

Why?

- Who owns the schema in a setup of multiple teams?
- One entry point for your API: building a monolith?

mergeSchemas

```
const { mergeSchemas } = require("graphql-tools");
const { ApolloServer } = require("apollo-server");
// import schemaOne and schemaTwo

const schema = mergeSchemas({
  schemas: [schemaOne, schemaTwo]
});

const server = new ApolloServer({ schema });
server.listen({ port: 4002 }).then(({ url }) => {
  console.log(`Gateway at ${url}`);
});
```

Remote Schema

```
const { HttpLink } = require("apollo-link-http");
const fetch = require("node-fetch");

const schema = async () => {
  const link = new HttpLink({ uri: "http://localhost:4000/", fetch });
  const schema = await introspectSchema(link);
  const executableSchema = makeRemoteExecutableSchema({
    schema,
    link
  });

  return executableSchema;
};

module.exports = schema
```

Exercise

Our application is super successful and the most requested feature is a search. The company decided to create a new team with it’s own service.

Create a Gateway that stitches together both Schemas.
Note: Do not care about the auth/headers/context.