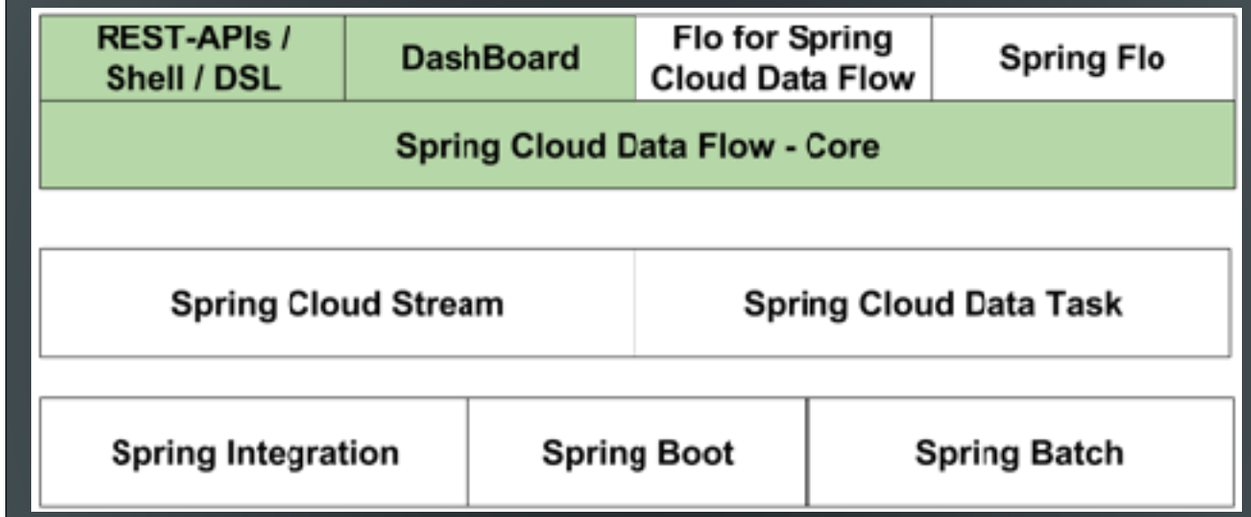
Spring Cloud Data Flow Architecture

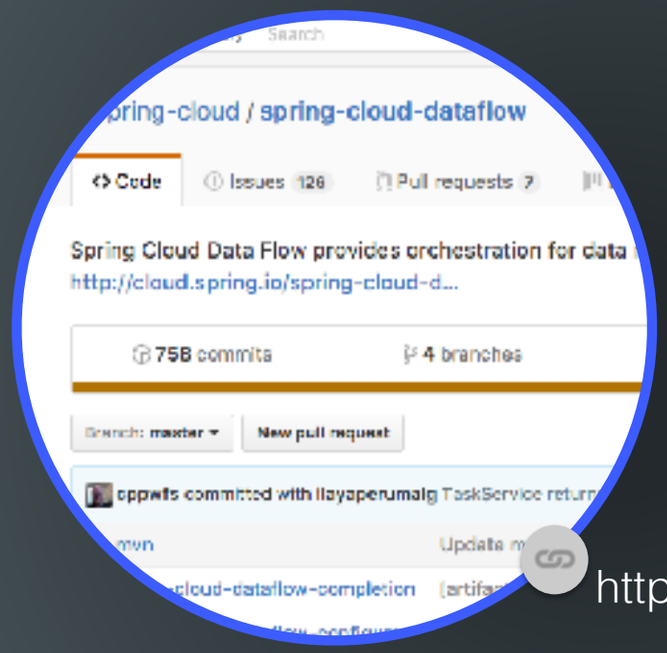
Glenn Renfro | Sabby Anandan

In this section we will cover
The major components of Spring Cloud Data Flow
Go into some detail of each and a quick overview of what they do.
Registering apps

The Big Picture



CODE IS ON GITHUB



https://github.com/spring-cloud/spring-cloud-dataflow

JAR LINK

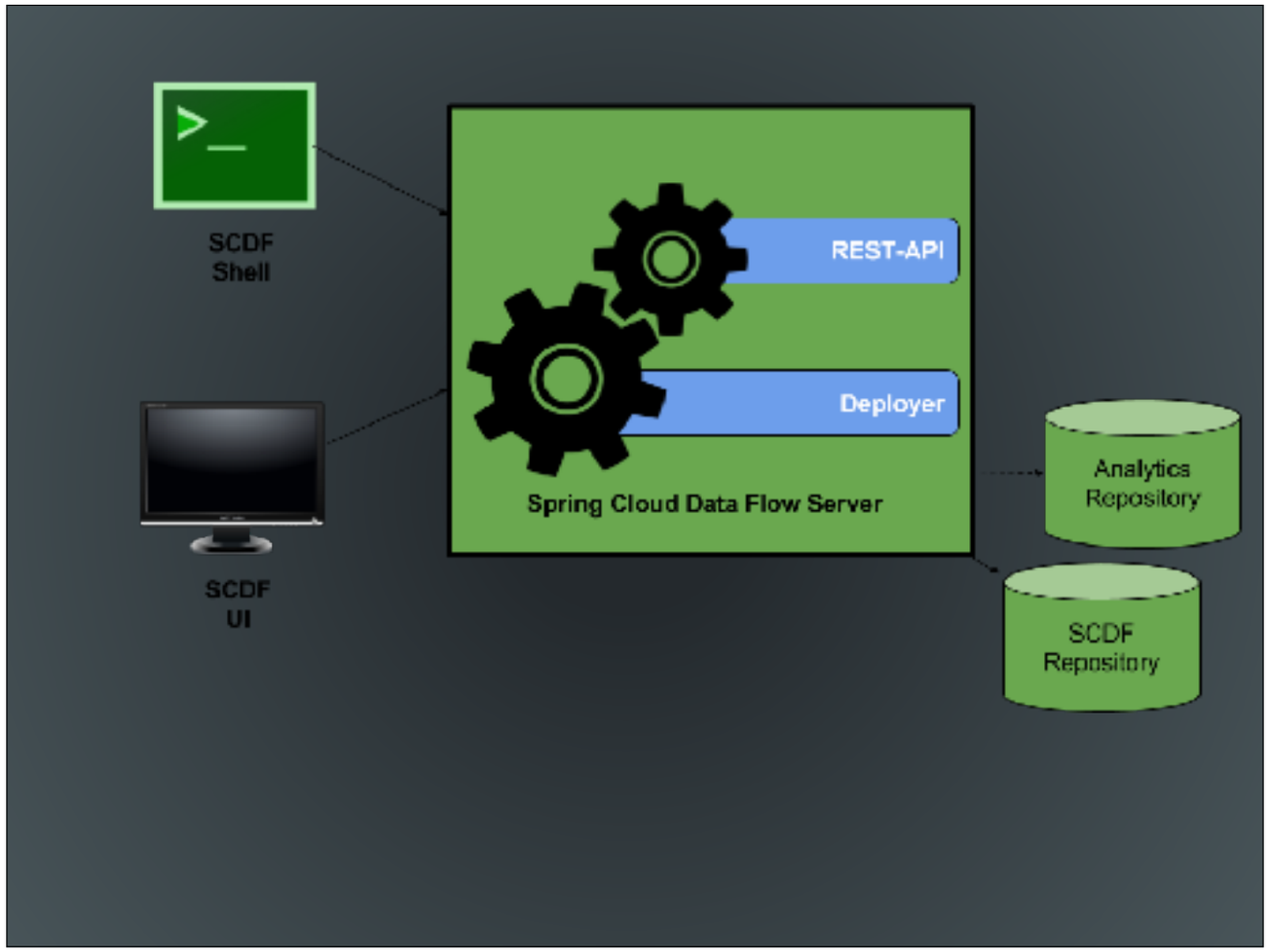
http://repo.spring.io/release/org/springframework/cloud/spring-cloud-dataflow-server-local/1.1.2.RELEASE/spring-cloud-dataflow-server-local-1.1.2.RELEASE.jar

https://flic.kr/p/9Bkxjo

JAR LINK

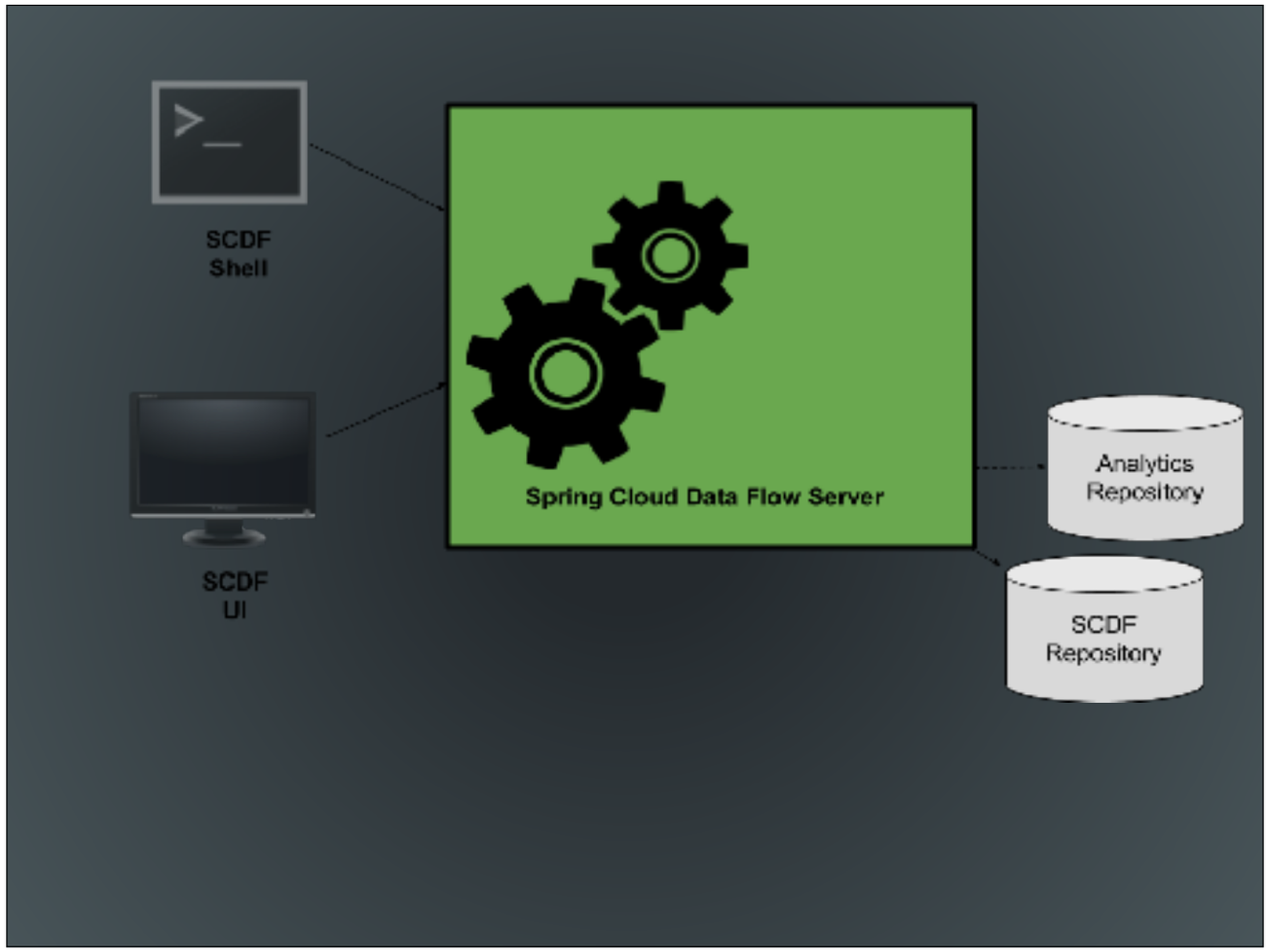
http://docs.spring.io/spring-cloud-dataflow/docs/1.1.1.RELEASE/reference/htmlsingle/

https://flic.kr/p/8MVYfc

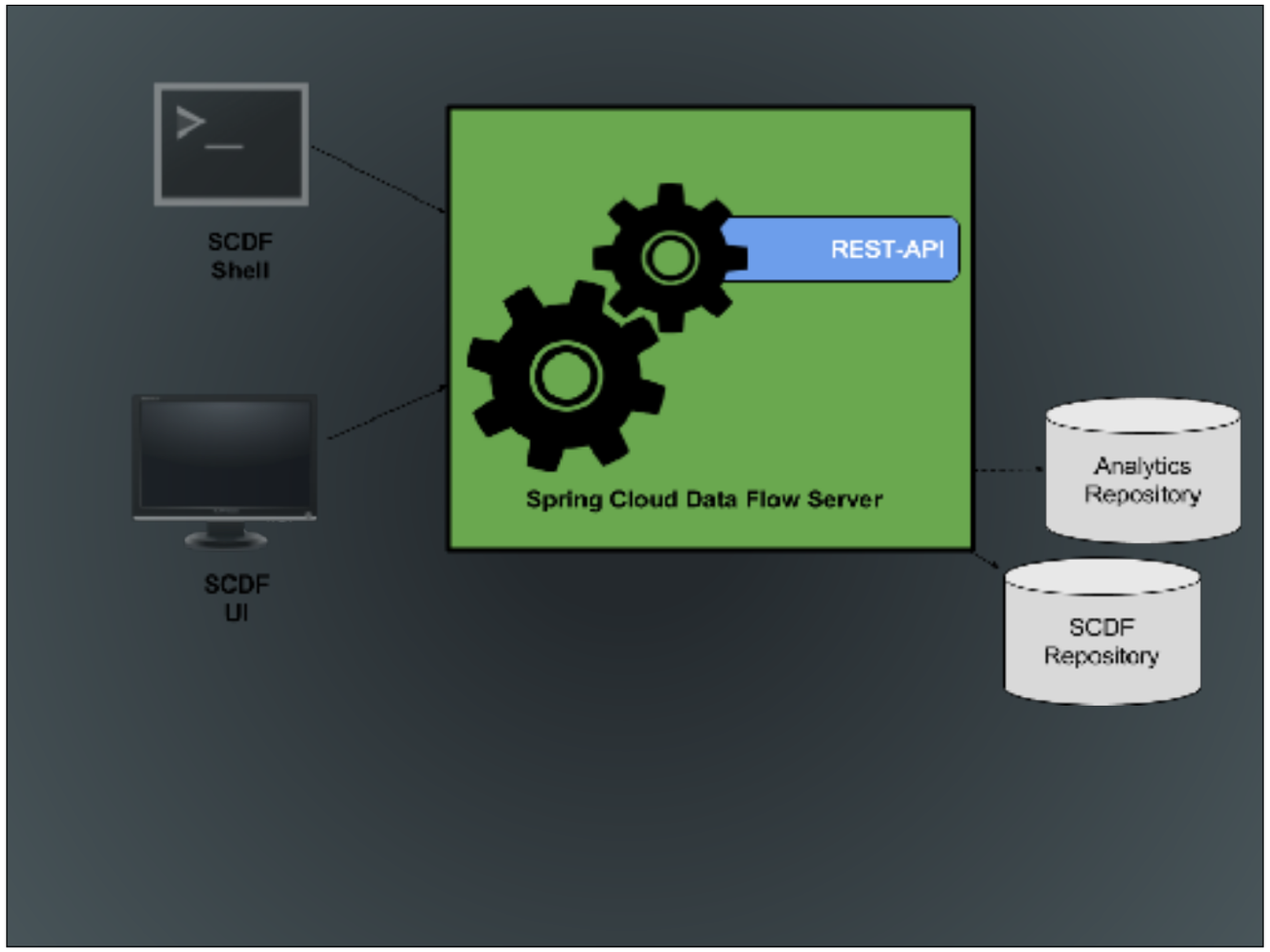


6 Components of the Spring Cloud Data Flow Server

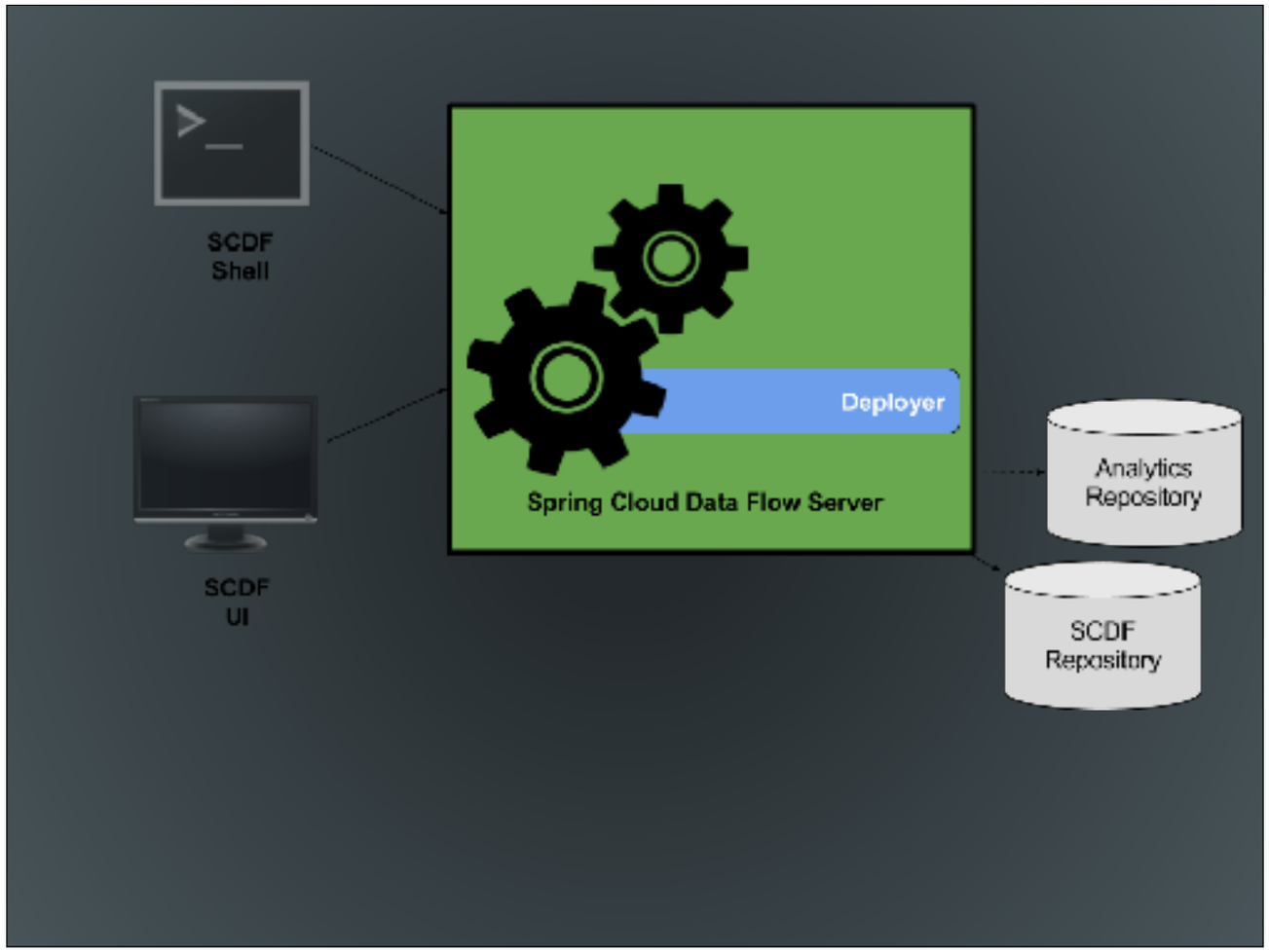
- Server -
- REST-API
- Deployer
- SCDF-UI
- SCDF Shell
- Repository



- What is the Data Flow Server
  - The Data Flow Server uses an embedded servlet container and exposes REST endpoints for creating, deploying, undeploying, and destroying streams and tasks, querying runtime state, analytics, and the like.
- Allows us to deploy the applications that compose a stream. Or an launch application(s) for a task
- Via a restful-API or UI allows users to retrieve the state of the apps of a stream or task.
- Offers the ability to view the current values of the analytics
- Stores the the URI's of where to obtain the application.
- Stores the definitions for all the tasks and streams

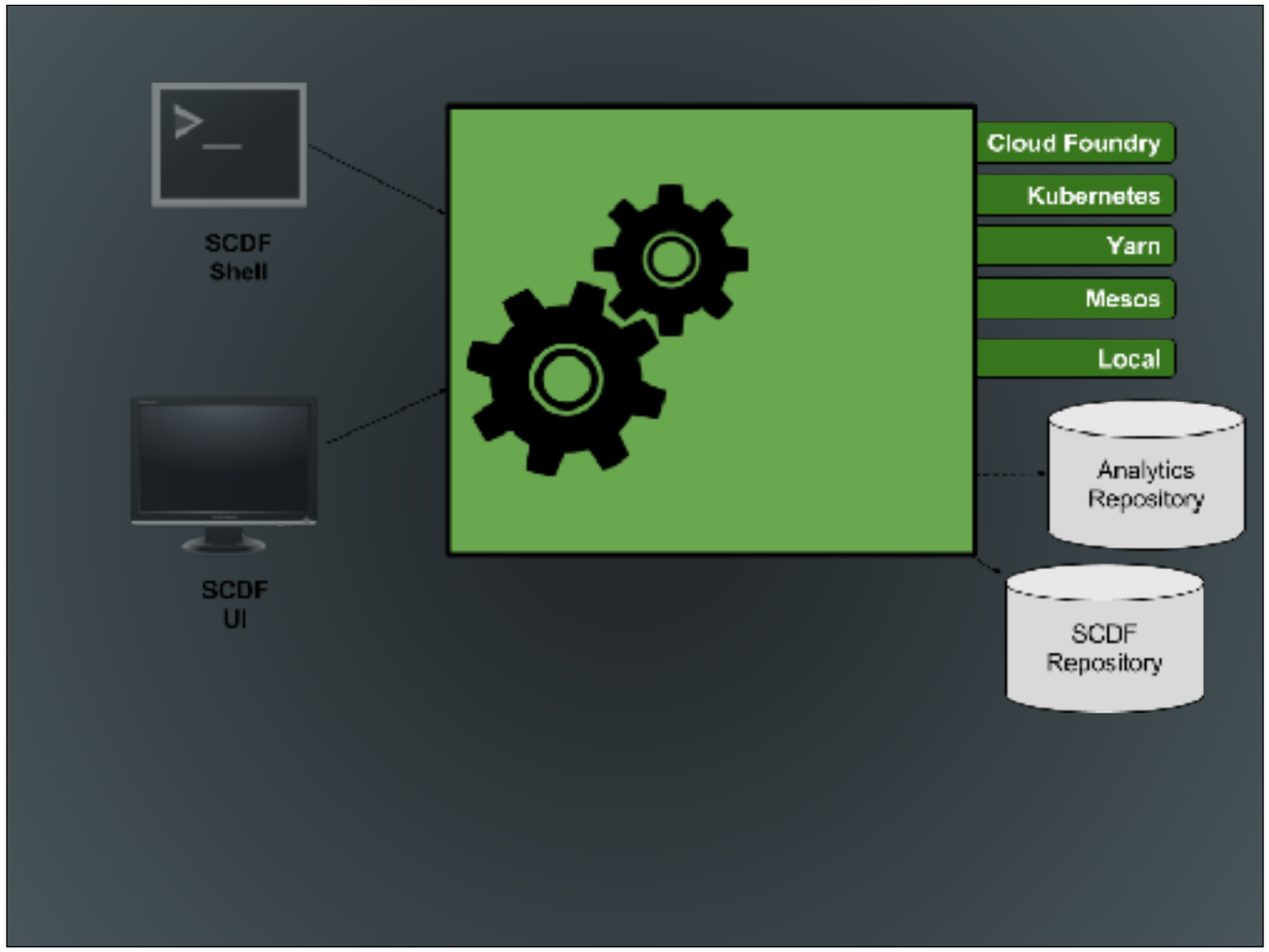


- Restful interface is offered from the SPI and thus all implementations support it.
- This is good for CI implementations
- Is build on Spring HATEOAS so it supports HATEOAS principles
  - Client doesn’t have to have prior knowledge of the server
- The endpoints are broken down into the following categories:
- Dashboard UI



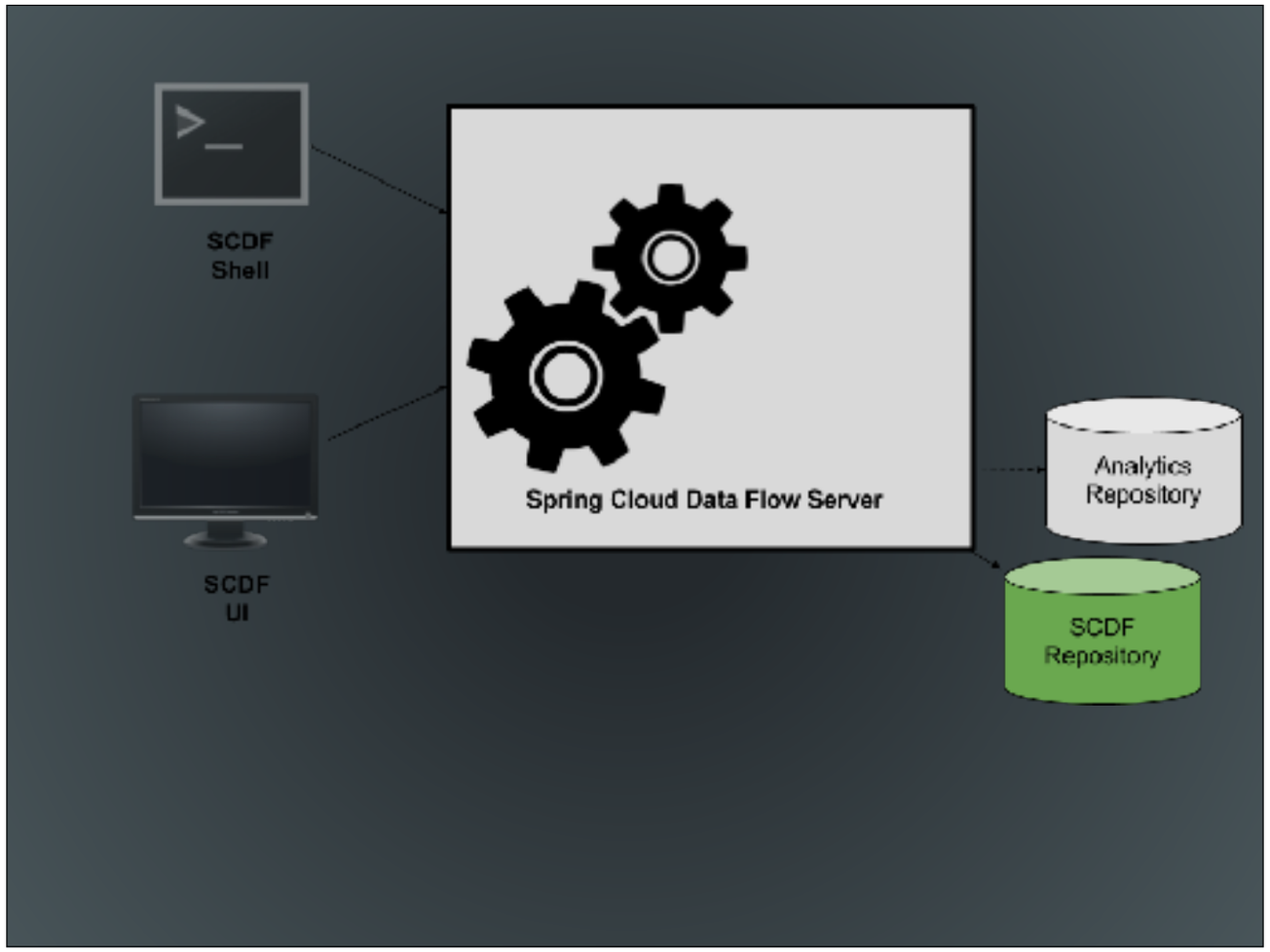
Spring Cloud Deployer

- Provides a common means to deploy applications to a platform
- Based on the Spring Cloud Deployer project https://github.com/spring-cloud/spring-cloud-deployer
- Each Spring Cloud Data Flow Server implementation uses one deployer. The current deployers that we support are:
  - CF
  - Mesos
  - Kubernetes
  - Yarn
  - Local
  - Others have been added by the community



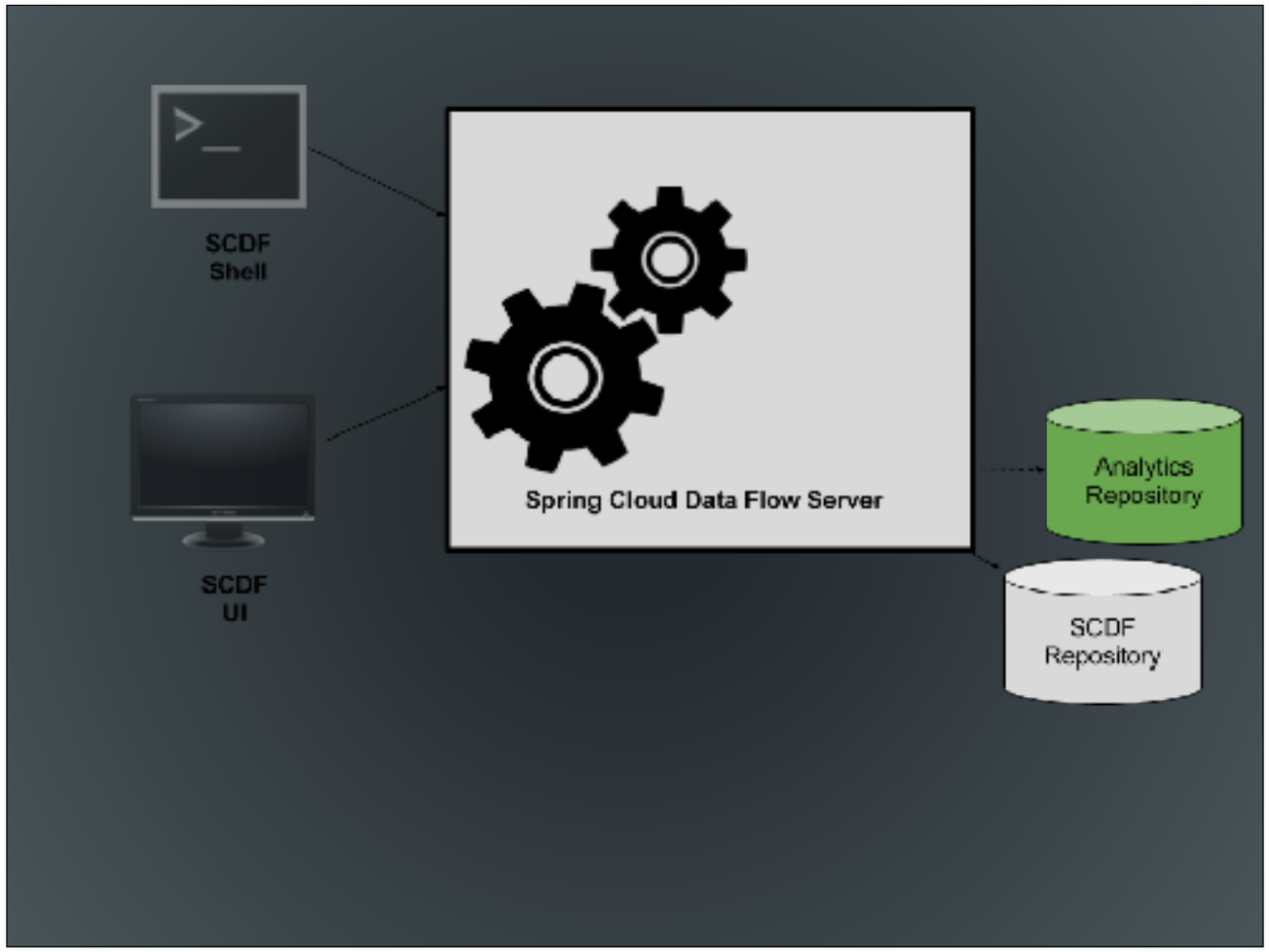
SCDF Server Types

- Spring Cloud Dataflow has a separate server for each deployment type.
- Cloud Foundry - https://github.com/spring-cloud/spring-cloud-dataflow-server-cloudfoundry
- Mesos - https://github.com/spring-cloud/spring-cloud-dataflow-server-mesos
- Yarn - https://github.com/spring-cloud/spring-cloud-dataflow-server-yarn
- Kubernetes - https://github.com/spring-cloud/spring-cloud-dataflow-server-kubernetes
- Local -SPI
- But all are based on the Spring Cloud Data Flow SPI - https://github.com/spring-cloud/spring-cloud-dataflow
- Can you support multiple platforms on a single SCDF instance.
- The answer is no. Each server supports only one platform.
- Why we are using Local
    - Meant for development purposes
    - Wanted to run it locally on your Machines
    - Wanted fast deployment to speed up labs
    - Wanted simple install (Yarn isn’t easy to install)



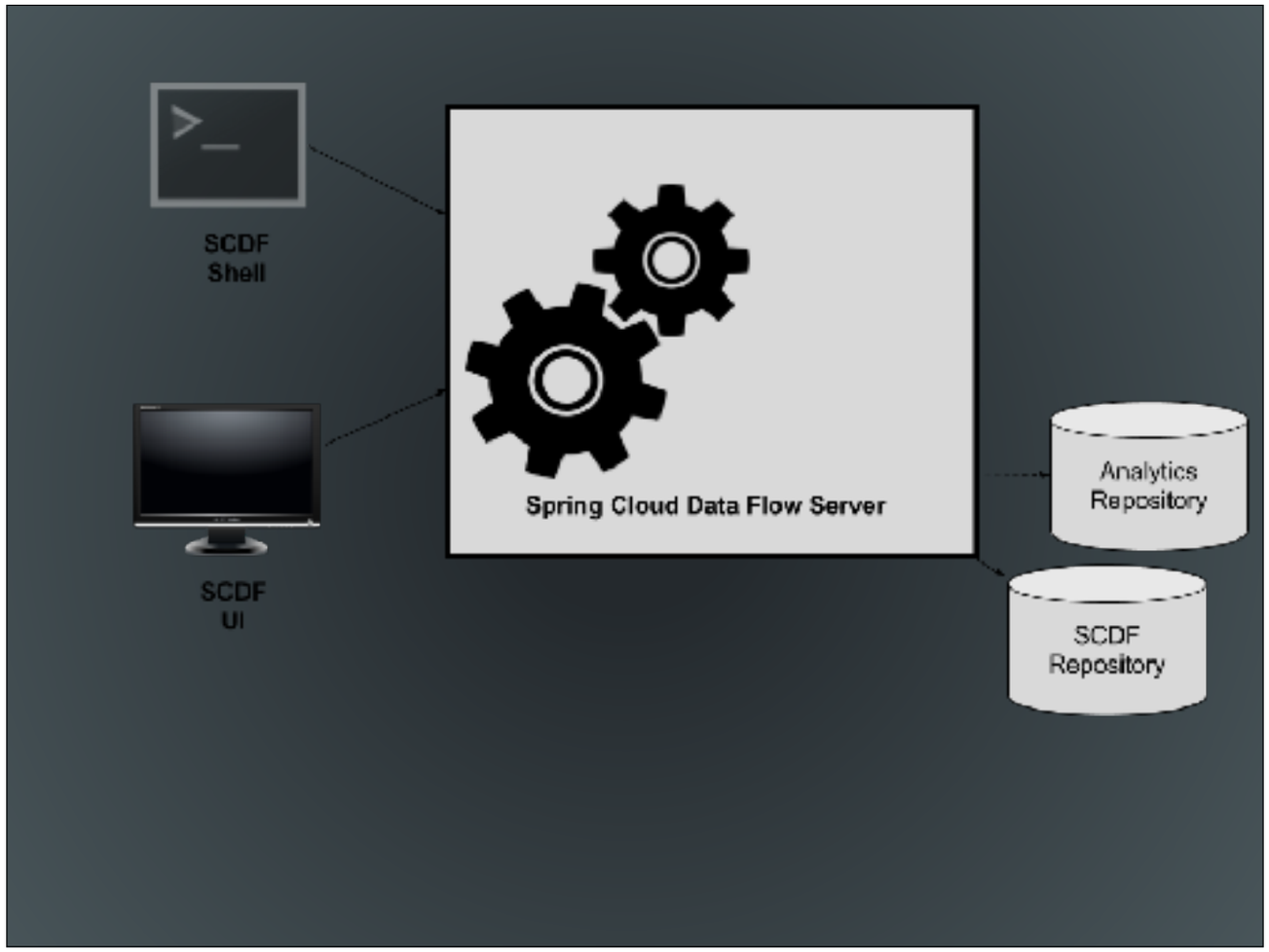
SCDF Repository

- Is an external relational database that stores:
  - Stream Definitions
  - Task Definitions
  - URI’s to the apps
  - Task Execution Statuses
  - Job Statuses
- By default Local uses an embedded H2 database
- Currently supported (out of the box) H2, HSQLDB, MySQL, POSTGRESQL
-



Analytics Repository

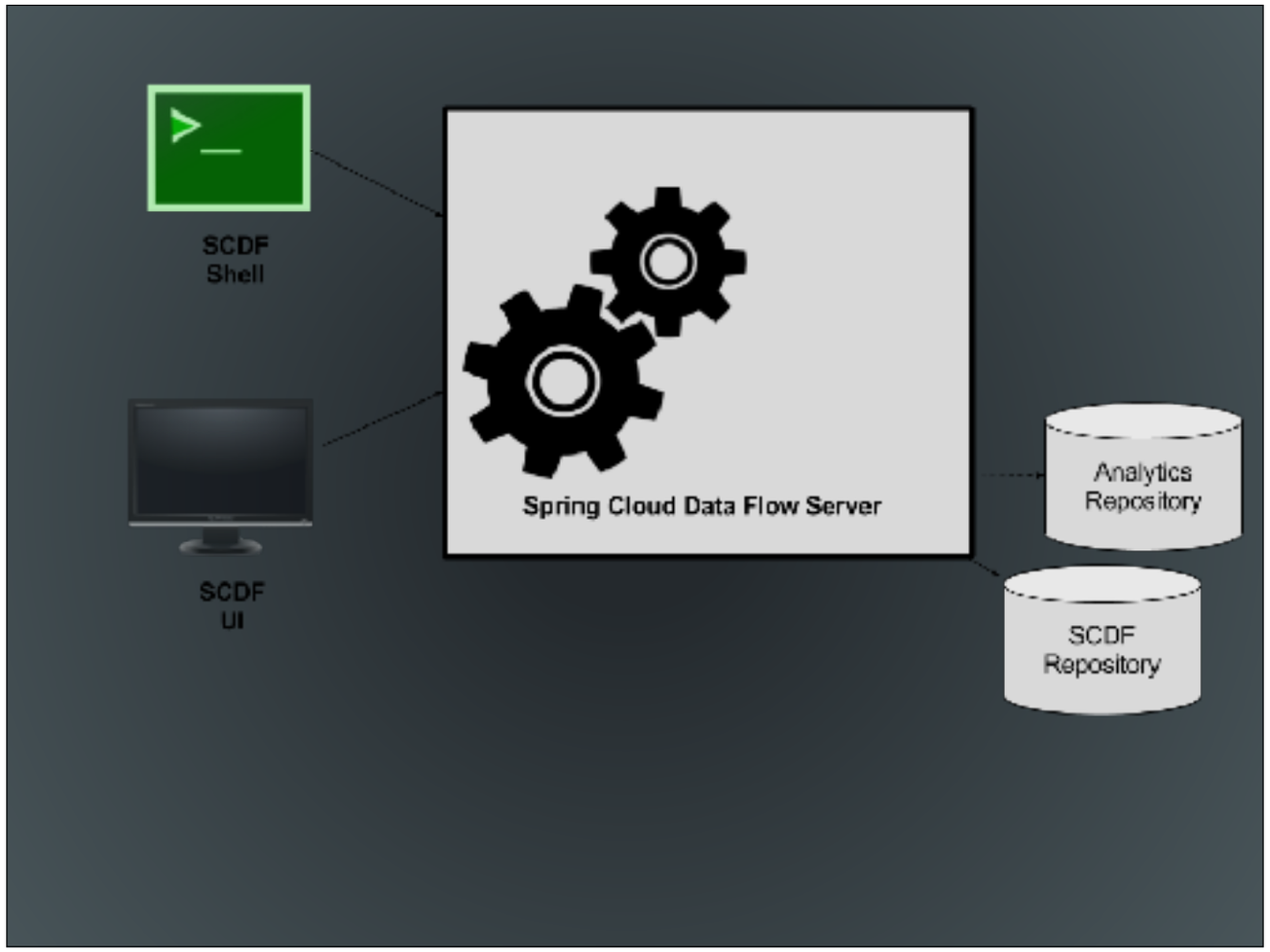
- Is an external Redis database that stores:
- Counts
- Aggregate-Counters
- Field Value Counters



SCDF UI

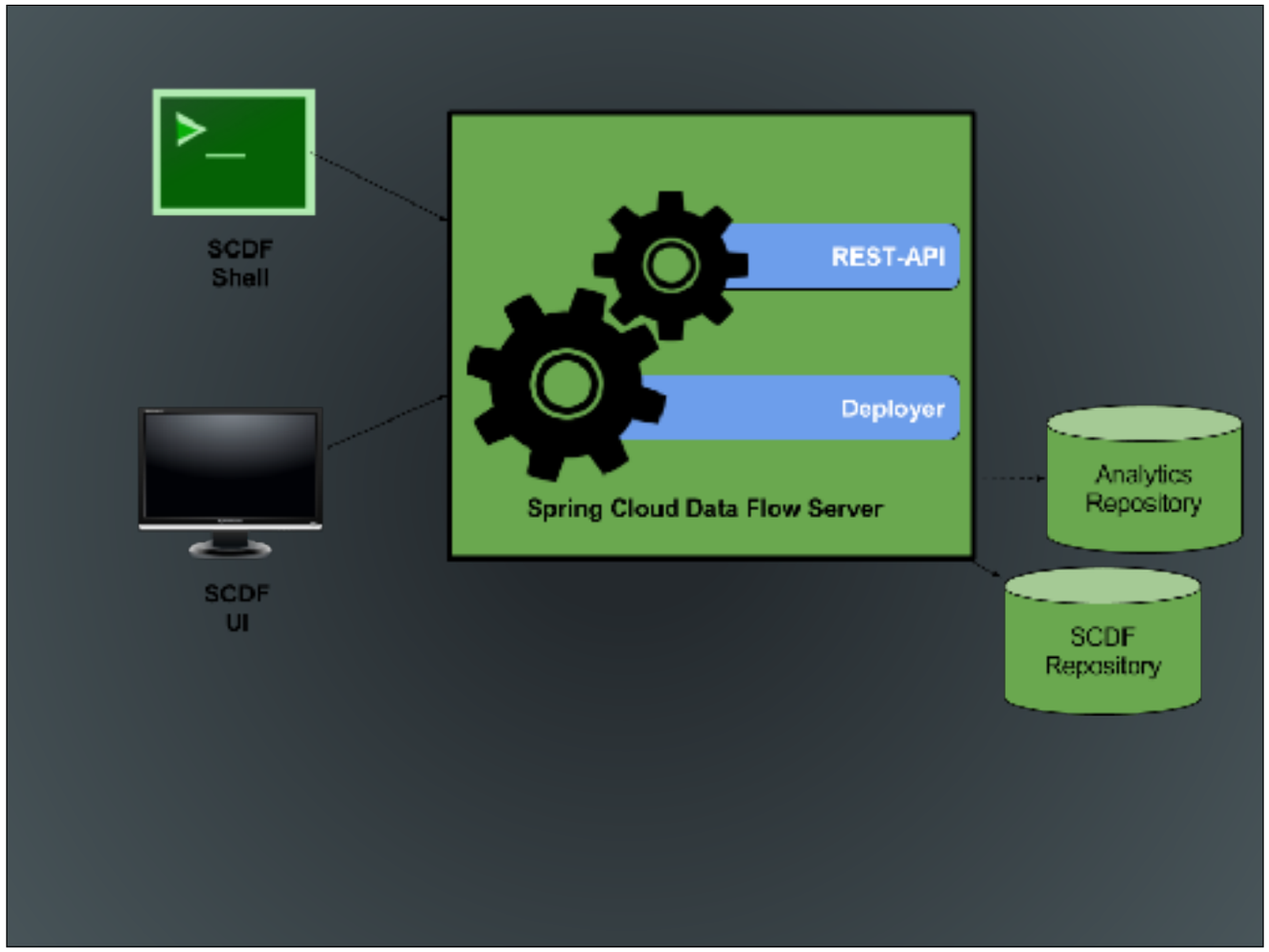
Offers an UI interface to the features offered by the RestfulAPI

- Accessible via the /dashboard endpoint ie. localhost:9393/dashboard
- Stream Creation, destruction and monitoring
- Task Creation, destruction and monitoring
- App registration
- Metrics montoring
- Job Monitoring



SCDF Shell

- Offers a command line interface to interface with the Restful API
- The shell has no real knowledge of the SCDF except what it gets from the Spring Cloud Data Flow Server
- Since it is a stand-alone app it can connect to any SCDF server that has its Restful API available for access
- By default it looks at localhost:9393/
-



Now we have the basic components that makeup a SCDF Server.

Security!

- Https
- Basic Authentication via LDAP or File based
- Single Sign On OAuth
  https://flic.kr/p/igAH3a

Startup Data Flow Server
Startup Shell
Startup Rabbit