

### Abstract

- Develop a cyberinfrastructure for supporting near-term ecological research.
- Use the GENI testbed as the “**edge**” cloud, and the Chameleon testbed as the “**core**” (back-end) cloud.
- Analyze and automate procedures to make ecological forecasts and update these forecasts over time as new data become available.
- Researchers can “compete” using different forecast models.
- Provide a state-of-the-art cloud-based infrastructure.
- User code runs as a “function” in the cloud.
- Highly scalable and easy to manage.
- Provide an interactive web-interface for users to submit/run code and view results.

### System Workflow

Figure 1 shows the EcoForecast workflow. To run an ecological model code on our EcoForecast system, the following steps are taken:

1. The user submits her code, along with dependencies, via the web interface. The code is sent to the Orchestrator.
2. The Orchestrator performs the following tasks:
  - Analyze the user’s code.
  - Install code dependencies.
  - Find the “best” place to run the code, e.g. Edge vs. Core and “best” configuration, e.g. amount of RAM.
3. The code runs as a “**Serverless**” function in a containerized environment in the cloud.
  - OpenWhisk supports the creation of code “*actions*” that can be “*triggered*” based on external factors.
4. The function processes its input data and stores its output (results) in the database.
  - It is faster to run the function closer to the data source and storage.
5. The output of the function is sent to the user and displayed on the webpage.

### EcoForecast on GENI & Chameleon

- **Web Interface:** The web interface is implemented using the Common Gateway Interface (CGI). It is hosted on GENI nodes. It provides a user with following functionalities:
  - Submit/run code
  - Retrieve results from the database
  - Plot results to compare different models
- **Orchestrator:** The orchestrator is implemented as a python module, and it co-resides with the web-interface. Upon successful submission of code, it packages the code as a serverless “action”. Depending on the requirements of the computation and available resources, it decides where to execute the code.
- **Cloud Infrastructure:** The EcoForecast’s cloud infrastructure is provided by the GENI and the Chameleon testbeds. We use OpenWhisk [1] as the “*serverless*” platform for running user code. To accommodate the varying needs for the computation resources, we use two different cloud resources:
  - **Edge Cloud (GENI Nodes)** is close to the data sources and the end-user, but it has limited resources.
  - **Core Cloud (Chameleon Nodes)** is far from the data sources and the end-user, but it has powerful computation resources that can run computationally intensive ecological models.
- **Databases:** EcoForecast uses MongoDB [2] to store the results of the computations. Moreover, the web interface provides easy GUI for accessing results from the database and plotting/comparing results across different ecological models.

### References

1. <https://openwhisk.apache.org/>
2. <https://www.mongodb.com/>

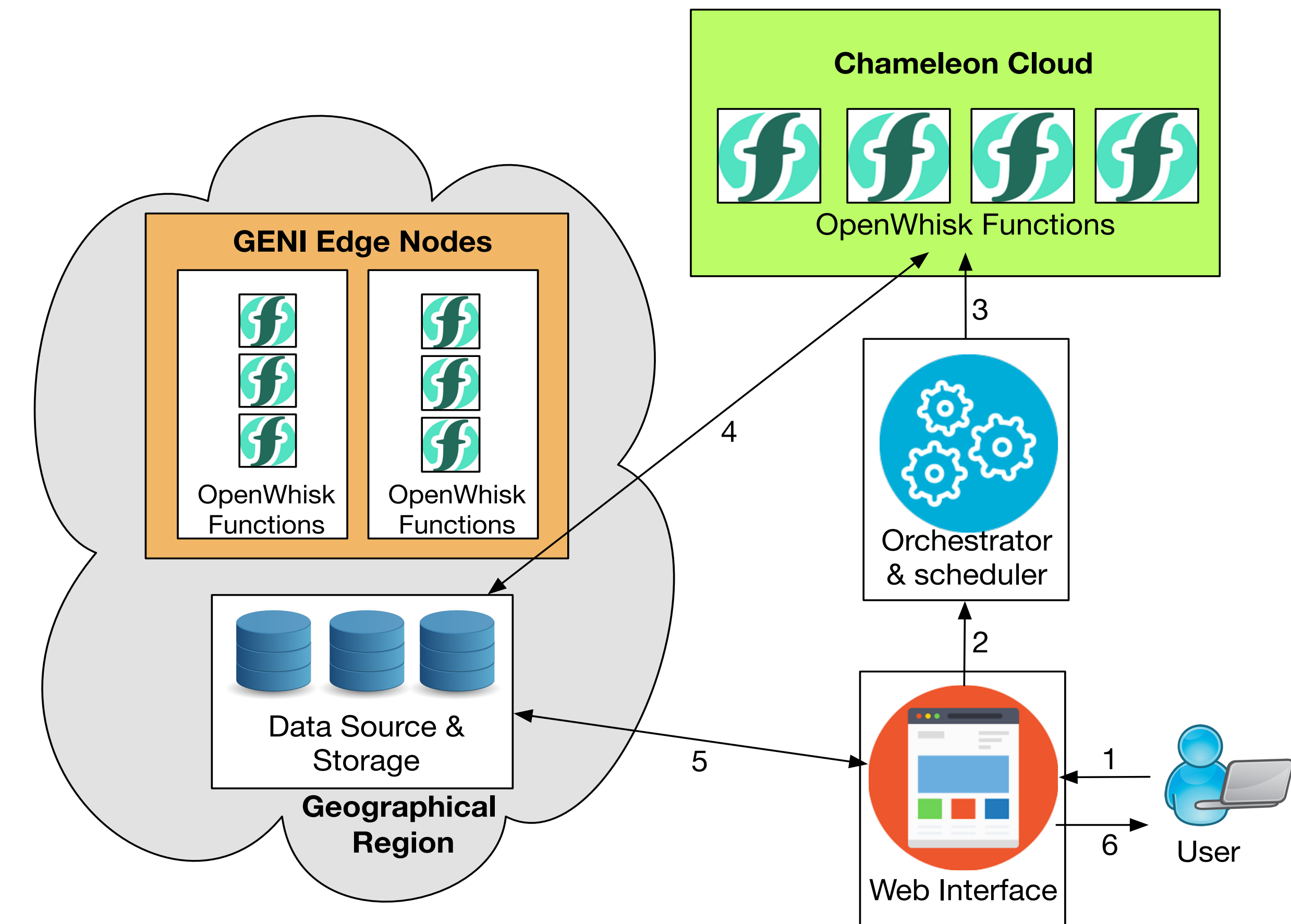


Figure 1: EcoForecast System Workflow

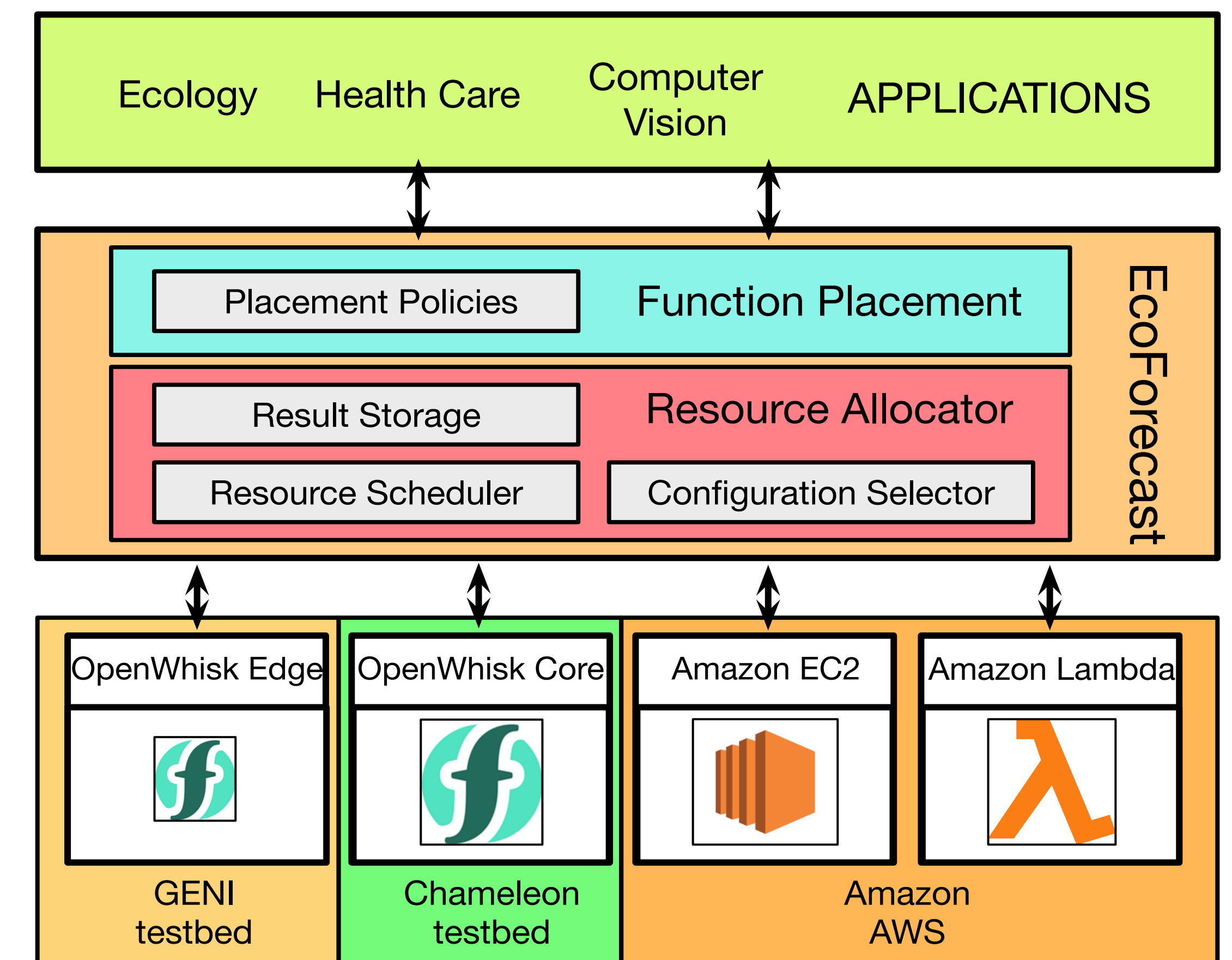


Figure 2: EcoForecast Architecture