

Draker Energy (now a BlueEnergy group company) is one of the largest providers of Solar Power Solution across the globe. It has built a reliable and robust software, a SaaS product for performance and asset management of integrated solar power systems. Draker has designed and installed more than 230 MW of solar and wind systems for commercial and utility applications.

The company provides monitoring and control for more than 2GW of renewable power systems as well as the world-class system life-cycle maintenance and support capabilities that its clients demand. Its solution is to maximize and control the efficiency and profitability of PV systems and enhance other BlueNRGY support and service capabilities tailored for asset owners and managers.

## Existing system:

Customer's application is a SaaS application which allows monitoring their customers solar panels to grid framework. The customer has 3 versions of the product

- ✓ IA6 (intelligent array 5) (legacy version)
- ✓ Sentalis (Current version in production)
- ✓ Intelligent Array 6 (the upcoming release)

Intelligent appliances deployed at customer sites pass on data via a web service to a Rack space cloud. The rack space cloud syncs back the data to a Cassandra DB and custom app servers. Rabbit MQ enables synchronization of the rack space cloud data to the customer's Amazon AWS instances. The AWS instances run BI processes on the data. After number crunching, the data is presented at the frontend and backed up on a MYSQL cluster.

## Key business requirements were:

- Leverage the legacy system for business continuity, but migrate to a new platform to AWS
- Automate the process of creating project type environment for each client on a multi-tenant architecture
- Faster synchronization between development and deployment servers
- Monitor the applications for uptime in all the three versions till the migration is completed

## Choice of Cloud platform:

The choice of the hosting platform and optimizing the various components on that was the biggest decision as a SaaS provider to give (7X24X365) access. The robustness and scalability of the platform, options for compute, memory, storage, bandwidth, multiple zones and open APIs were critical factors. Newt Global recommended that among all the

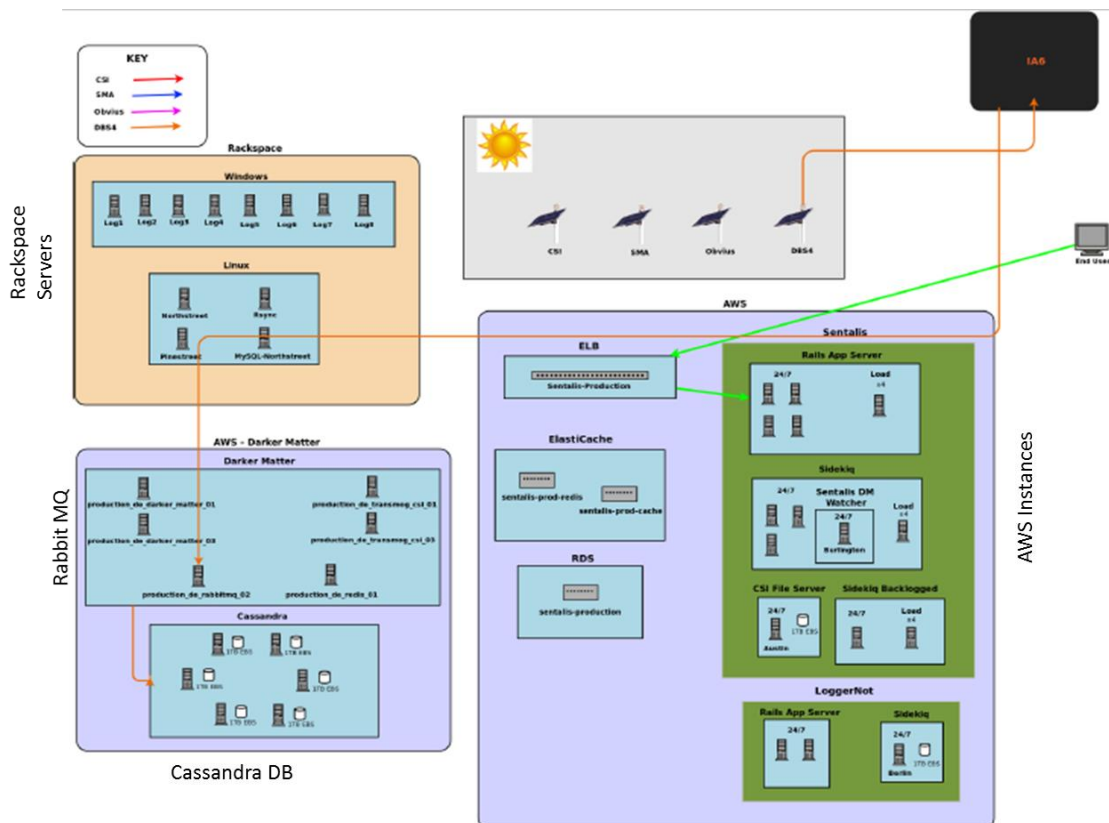
options commercially available at that time, AWS presented the best environment to meet most of the requirements.

## Solution Overview:

With a deep understanding of the AWS features to address the client requirements, Newt Global offered the following solution.

Launched EC2 instances for Dev, QA, Stage and Prod environments. Created appropriate security groups and mounted EBS volumes and configured to store log files and project archives. Calculated CPU, RAM and IOP's for the application and selected instance type based on the calculations.

Used **OpsWorks** to deploy and manage applications on Dev, QA, Stage and Prod environments. Created workflow to manage IAM access and ensured seamless communication between AWS and other tools like Git, NewRelic etc. The workflow automation extended to launching and managing MySQL and Postgres instances for staging and production environments.



## Deployment highlights:

- Wrote cookbooks on **Chef** for infra and deployment automation.
- SSH Key Management – Automated key addition and deletion for user access control. Created a shell script automatically add or delete user keys form all the instances in different environments.
- A Web-hook is used between GitHub and key management script to automatically execute the script whenever there is a modification in the source code.
- Created Shell, Ruby and Python scripts for monitoring different services across different platforms and restart the stopped services. It also triggers email alert using **AWS SES**.
- Database cluster formation with **Cassandra**, node addition, node deletion, upgrade and maintenance.
- Created **Docker file** to Containerize IA6 application on stage and production instances.
- Configured monitoring tools like **NewRelic**, **Graphite** and **Cloudwatch** for instance and application monitoring.

## Business benefits:

- ✓ Deployments were made easier and the turn-around time was reduced by 4 times.
- ✓ The Development cycles were made faster as the feedback loop was faster due to automated deployment.
- ✓ The Synchronization between the Development and Deployment was reduced by over 50%.
- ✓ Ease of monitoring of all the applications was done using open source and native AWS based monitoring tools.

