eDrilling Docs

Kubernetes Deployment Requirements

This document describes the hardware and software requirements for both on-premise and in-cloud deployments in Kubernetes.

A Kubernetes deployment handles scalability, security and high-availability. For a Windows server deployment, refer to separate requirement document

1. Software Requirements

Backend software

- Kubernetes cluster
- Optional: Azure AD for user management and authentication [1].

The following web browsers are supported:

- Google Chrome / Chromium
- Firefox
- Edge v79 or newer

2. Hardware Requirements

2.1. Deployment option 1: on-premise

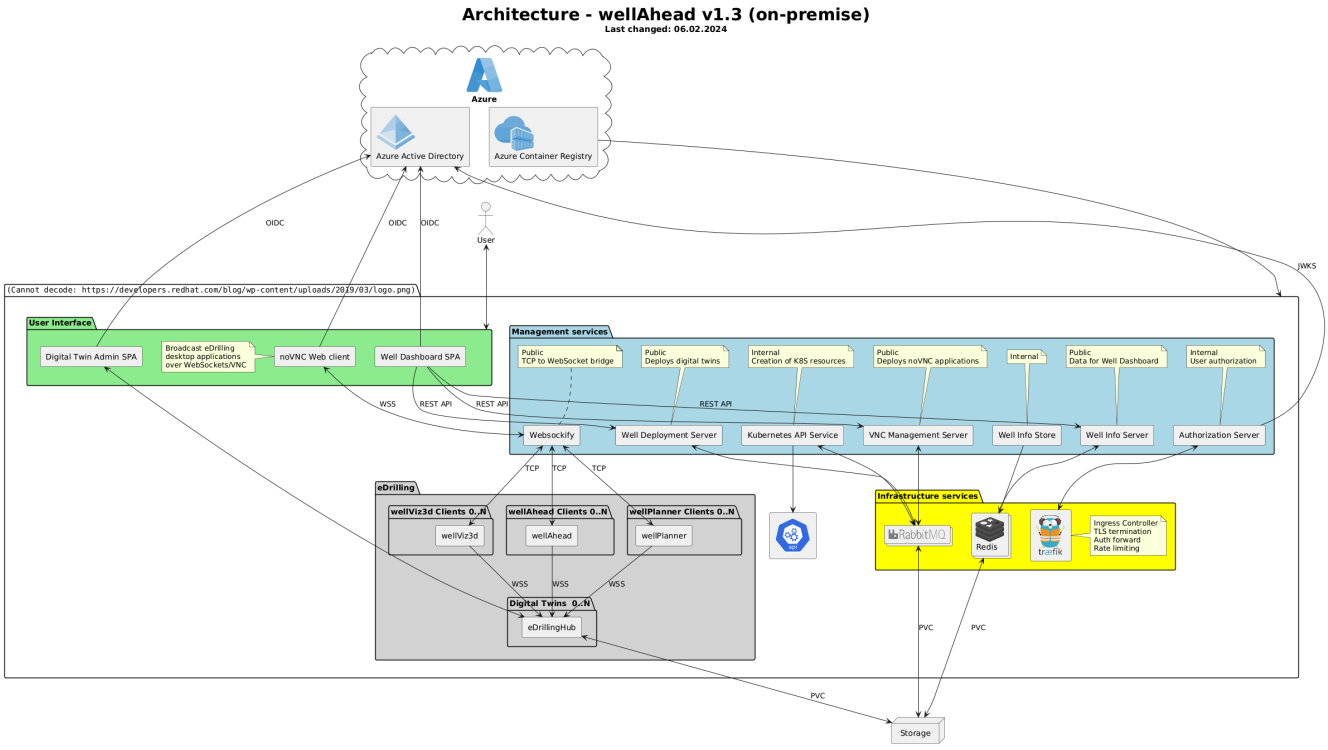

Figure 1. On-Premise architecture

NOTE

We assume a pre-existent Kubernetes cluster (with a minimum of 3 master nodes [2]) running Linux-based nodes.

Quantity of worker nodes (in addition to master nodes) required is a function of the number of wells to be running in parallel, and number of concurrent users of monitoring and planning applications. Storage class will be used for persistent storage of data using SSD.

Requirement formulas

- $\texttt{CPU (cores) = 10 (infrastructure) + } x_{\texttt{wellAhead}} \texttt{ * 1.5 + } x_{\texttt{wellGuide}} \texttt{ * 9.5 + } x_{\texttt{wellAhead\_client}} \texttt{ + } x_{\texttt{wellViz3D}} \texttt{ * 2 + } x_{\texttt{wellPlanner}}$
- $\texttt{Memory (GiB) = 12 (infrastructure) + } x_{\texttt{wellAhead}} \texttt{ * 3.5 + } x_{\texttt{wellGuide}} \texttt{ * 5.5 + } x_{\texttt{wellAhead\_client}} \texttt{ + } x_{\texttt{wellViz3D}} \texttt{ + } x_{\texttt{wellPlanner}}$

Where:

- $x_{\text{wellAhead}}$ is the amount of running wells.
- $x_{\text{wellAhead\_client}}$ is the monitoring and plotting tool, also called OpenView2D. This is also a required tool in wellGuide.
- $x_{\text{wellGuide}}$ requirements are in reality non-linear. If you request multiple wellGuides (>3), reevaluate these numbers.

NOTE

For high-availability we recommend at least 3 servers used as worker nodes.

2.1.1. Example

If a company requests to monitor 12 wellAheads, 2 wellGuides, with 12 wellAhead clients, 6 wellViz3Ds, and 15 wellPlanner applications, the requirements will be:

- $\texttt{86 = 10 + 12 * 1.5 + 2 * 9.5 + 12 + 6 * 2 + 15}$
- $\texttt{98 = 12 + 12 * 3.5 + 2 * 5.5 + 12 + 6 + 15}$

Hence we recommend 86 cores and 98 GiB.

2.2. Deployment option 2: cloud (example: Azure)

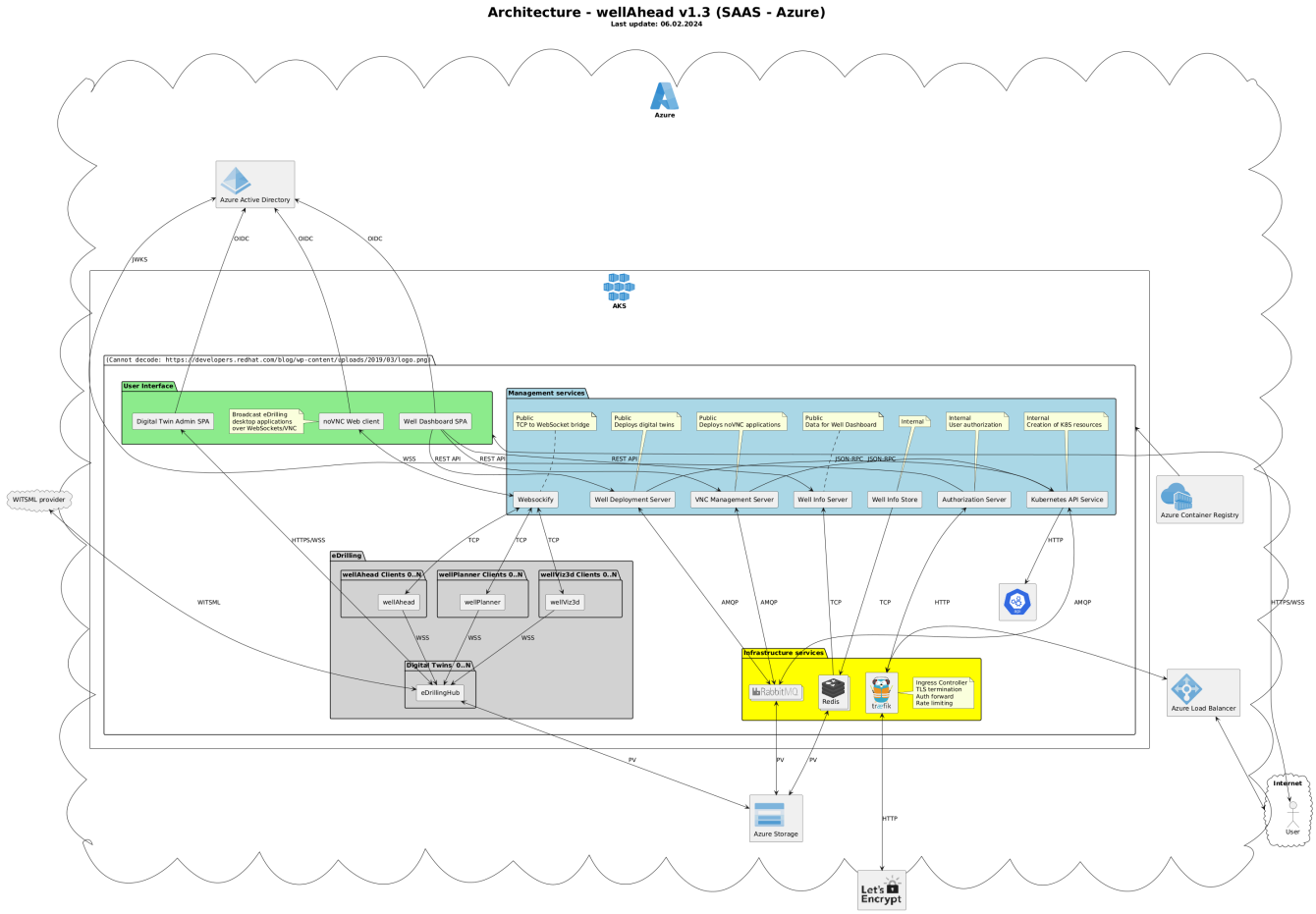

Figure 2. Azure architecture

Load balancer Managed disks (SSD) Managed Kubernetes services (3 node pools for wellAhead + 1 node pool for wellGuide):

- System node pool (3 nodes) running Standard_F2s_v2 (2 vCPU, 4 GiB memory)
- User node pool #1 (autoscalability) running Standard_D2s_v3 (2 vCPU, 8 GiB memory)
- User node pool #2 (autoscalability) running Standard_F4s_v2 (4 vCPU, 8 GiB memory)
- User node pool #3 (autoscalability) running Standard_D8plds_v5 (8 vCPU, 16 GiB memory)

2.3. Basis of calculation of infrastructure requirements

Static deployments, including 3rd party software, wellGuide, wellAhead:

CPU (cores): request/limit: 5 / 14.5 (assuming 1.5 / 3 for Longhorn) Memory (GiB): request/limit: 7 / 16 (assuming 1.5 / 3 for Longhorn)

We assumed average values between request and limit for actual requirements.

2.4. Test results

Example Hardware

- Instance type: D2s_v3 (on Azure) - 2.4 GHz Intel Xeon® E5-2673 v3 (Haswell) or the 2.3 GHz Intel Xeon® E5-2673 v4 (Broadwell)

- Disk: 2300 IOPS, 150 MB/second

Performance

- 4 EDHs per 2 VCPUs (max 1 VCPU per pod): 0.5 VCPUs per pod.
- We measured the performance to be ~ 250 ms per data set, hence 0.25 VCPU per pod is well within limits.

4 wells per Kubernetes Node (2 VCPUs) would be the limit on Azure for the tested instance type, since they limit the disk volumes to 4 for each Node.

If we assume:

- One data set per second (from Witsml)
- Use a modern 8 thread CPU
- Leave some resources available for administration, analytics and reserve (<25%)
- Have a local Kubernetes environment without restriction on amount of disk volumes per Node
- Similar or higher IOPS as performed in our test
- Not count the Kubernetes "mother" Node, since we are now considering scaling
- In a real environment (automated deployment), it is difficult to predict on which Node (server) each Pod (well) will be placed, so some Nodes might not be utilized fully.

```
3 (wells per thread) x 8 (threads) = 24 wells per CPU
```

1. If not provided, Keycloak will be deployed
2. one node refers to one server