Building an Enterprise Analytics Platform with Jupyter Enterprise Gateway

github.com/jupyter-incubator/enterprise_gateway

**Jupyter Notebook Platform Architecture**
- Notebook UI runs in the browser
- The Notebook Server serves the ‘Notebooks’
- Kernels interpret/execute cell contents
- Are responsible for code execution
- Abstracts different languages

**Remote Kernelspec**
```json
{
    "kernel_name": "python",
    "display_name": "Python on Kubernetes",
    "language": "python",
    "class_name": "enterprise_gateway.services.processproxies.k8s.KubernetesProcessProxy",
    "context": "kern",
    "addr": "http://kubernetes_hapi.kubemaster01:50007",
    "subprocess_argv": ["python", "/usr/local/share/jupyter/kernels/python_kubernetes/scripts/launch_kubernetes.py"],
    "env": {
        "argv": ["argv"],
        "env": {
            "connection_file": "/usr/local/share/jupyter/kernels/python_kubernetes/scripts/launch_kubernetes.py",
            "display_name": "Python on Kubernetes",
            "language": "python",
            "config": {"connection_file": "/usr/local/share/jupyter/kernels/python_kubernetes/scripts/launch_kubernetes.py"}
        }
    }
}```

**Enterprise Analytics Platform Characteristics**

**Shared Compute Resources**
- Enterprise Cloud, Public Cloud or Hybrid

**Distributed Consumers**
- Notebooks running local
- Notebooks as services

**Different Utilization Patterns**
- High number of idle resources

**Jupyter Notebook Platform Limitations**
- Single-user sharing the same distributed filesystem and privileges
- Resources are limited to single node
- No security, users can see and control each other’s process

**Jupyter Enterprise Gateway** addresses enterprise requirements and use cases

**Current Feature Set**
- Optimized Resource Allocation
  - Run Spark in YARN Cluster Mode to better utilize cluster resources
  - Supports YARN, IBM Spectrum Conductor, Kubernetes resource managers
  - Pluggable architecture for additional resource managers

- Enhanced Security
  - Enable TLS for all socket communications
  - Any HTTP communication should be encrypted (SSL)

- Multiuser support with user impersonation
  - Enhance security and sandboxing by enabling user impersonation when running kernels.
  - Individual HDFS home folder for each notebook user.
  - Use the same user ID for notebook and batch jobs.

**Integrated with the Jupyter stack**

**Supported Platforms**
- Spark
- Kubernetes
- IBM Spectrum Conductor

**Remote Kernel Creation & Shutdown**

**Distributed Kernels == Optimized Resources**

**Platforms Supported**
- Jupyter Notebook
- Jupyter Client
- Jupyter Kernel Gateway
- Jupyter Notebook Server