**Project Analysis**

**-Nikhil T Y**

**Project Name: - Kinetic**

**Organization: - Kinetic Community LF?**

***Introduction: -***

Kinetic Data was built from the ground up for large, complex enterprise organizations that require a platform that is both **flexible** and **scalable**.

The Kinetic Platform enables **real** change, so a time investment is required to learn the underlying philosophies and abstraction layers in order to be successful. Once understood, the power of the Kinetic Platform can be unleashed to digitize and transform all types of processes within a business.

We've been doing this for a long time and have poured everything we've learned into our software. Kinetic Data has solved for the pains that you don't know you have until you have them, such as:

* In-flight workflow versioning
* Transforming and manipulating data on-the-fly in-between workflow steps
* Complex error handling and routing of errors to tools for automated or manual resolution
* The ability to integrate with ANY system without waiting for the vendor to release an update or build it for you
* Full control of the UI/UX within the end user experience to solve an accessibility issue
* Dynamic security policies based on functions and variables, not just roles and groups
* The ability to work within your existing CI/CD process

***Project Summary: -***

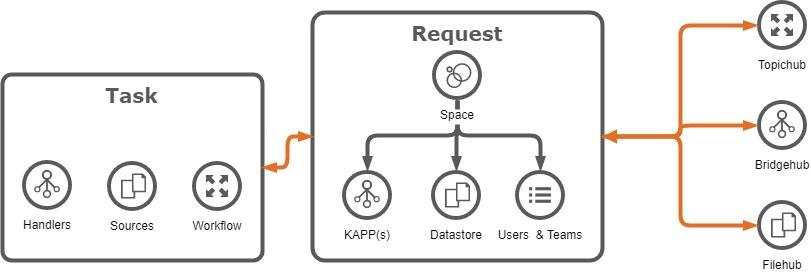
| Website: | <https://community.kineticdata.com/docs> |
| --- | --- |
| Organization or foundation name: | Kinetic community LF? |
| License: | no name user license model |
| Open / Proprietary | Open |
| Source Code | [KineticCommunity/request-ce-bundle-kinetic-oscar2 (github.com)](https://github.com/KineticCommunity/request-ce-bundle-kinetic-oscar2) |
| Brief Description | Kinetic Data encourages the use of the latest stable releases for all software supporting our applications. All versions referenced in this document are the latest stable release at the time of this writing. |

**PROJECT DETAILS:**

Key features-

* **Agent-**A portion of the Kinetic Platform that sits inside the firewall to execute calls that must take place internally for kinops systems that connect to on-prem systems (such as handlers and bridges). Agent replaced Kinetic Bridgehub and Kinetic Filehub. Agent is the harness that runs bridge adapters and file adapters. It can also be extended to run task handlers in a customer's security context.
* **API-**A software intermediary that allows two applications to talk to each other.
* **Attribute-**User-defined metadata for Spaces, Kapps, Users, Forms, Categories, Datastore Forms, or Teams. Also sometimes used as the name for the defined values in a Bridge Model.
* **Bridge Mapping**-The location where the bridge data source is mapped to the bridge model properties. This tells the Kinetic Platform how to get the raw data from the data source, and how to map the source data names to data fields for Kinetic Bridge Models.
* **Connector-**The connection between two nodes within a workflow. May or may not contain logic. If logic is applied, the connector path is only followed if the attached logic statement evaluates to not false. Why "not false"? Your connector could return a "Name" of a person - and if a name exists we want that branch of the workflow to execute.
* **Kinetic App (Kapp)-**A collection of Forms within the Kinetic Request CE System. Kapps have their own metadata, categories, webAPIs, forms, and permissions. Typically a Kapp will represent a single end user experience. Example Kapps include a Service Portal or Work Order System.
* **Kinetic Security Language (KSL)-**A functional language created for writing and managing security policies. KSL is implemented as JavaScript code.
* **Workflow-**A tree or routine. A process that does the work for a given event. Sample events: Submission Created, User Defined, Form Modified, etc..

***Architecture:-***



The Kinetic Platform consists of three primary components:

Kinetic Core,

Kinetic Task, and

Kinetic Agent.

# Kinetic Core

Request is the Platform component that allows you to build powerful and beautiful front ends for your customers. It contains a drag-and-drop form editor where you can easily construct the forms you wish to present your users. It also supports the complexities of multi-system integrations, multi-tenancy, and workflow integration on all its components.

You may see Kinetic Core referred to as Request CE, Request Core Edition, CE, or Core Edition. The platform is built with a version of Request (CE) that is entirely our own creation, unlike our previous versions (RE, or Remedy Edition) that relied on Remedy to be the platform.

Request is the point of connection in the Platform where Task, Filehub, Bidgehub, and Request itself all come together to provide the customer with the optimal user experience.

# Kinetic Task

Task is the Platform component that allows you to build and execute workflows. It contains a drag-and-drop workflow editor where you connect code snippets called handlers together to build flexible and powerful workflows. It also contains the engine that runs to execute these workflows and handle errors, providing fast workflow processing and robust error handling for all your workflow needs.

There are hundreds of handlers already built and available for use. These can connect you with everything from your local Request system to your AWS provisioning service and everything in between. Also, if you desired integration has an API but no handlers currently available, there are instructions available for you to be able to build your own.

While Task is a powerful workflow engine without the rest of the platform and can be used standalone, using Task with the Platform allows you to integrate that powerful workflow into your beautiful front-end experience for an optimal user experience.

# Kinetic Agent

Agent is the Platform component that provides the connectivity between the front end of the platform and other systems. This is what provides multi-system integration within the Kinetic Platform's front ends.

Current usage-

Users are the primary way to identify and authorize systems and individuals within the Kinetic Platform. Users within Kinetic can be created manually by other users that have permissions to create new users, or automatically by configuring the platform to authenticate against an IDP (Identity Management Provider) using common protocols like SAML.

Fundamentally, users must exist within the system in order to interact with areas of the system that require authentication. All users within Kinetic are scoped to a space, and no users are shared between spaces.

# **Technical Details**

# Hardware

## Web Application Server (Minimum 2)

* CPU Cores - 4+
* RAM - 8+GB
* Storage - HDD or SSD - 500GB minimum
* OS - Ubuntu 16.04 LTS

## Cassandra Database Server (Minimum 3)

* CPU Cores - 4+
* RAM - 8+GB
* Storage - Solid State Drive (SSD) - 1TB minimum
* OS - Ubuntu 16.04 LTS

# Software

## Database Server

* Java Server JRE 8 or JDK 8
* Python 3 (used for cqlsh - a command line CQL tool)
* Apache Cassandra version 3.11

## Web Server

* Production Grade Relational Database (for Kinetic Task)
  + PostgreSQL 10 (Microsoft SQL Server and Oracle are also supported)
  + This database does not need to be installed on the web server, it just needs to be accessible from the web server.
* Java Server JRE 8 or JDK 8
* Apache Tomcat Apache Tomcat 8.5 w/adjusted Java Heap size to 2 GB+
  + Kinetic Request CE
  + Kinetic Bridgehub
  + Kinetic Filehub
  + Kinetic Task

# Port Summary

## User Ports

* 80, 8080, 443, 8443 - Ports used to access the system. Typically only ONE will be chosen. 80/8080 are non-secure, 443/8443 are SSL ports.
* 3389 - Used by Admins to access the web servers via Remote Desktop. Does not  
  have to be open to all users, but to a restricted Source IP/Range. (Does RDB require additional ports?)
* 22 - Used by Admins to access the Cassandra servers. Does not have to be open to all users, but to a restricted Source IP/Range.
* 1433 - Used by Admins to access the Task database. Does not have to be open to all users, but to a restricted Source IP/Range

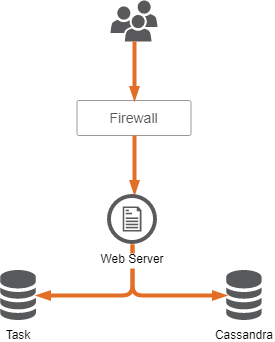
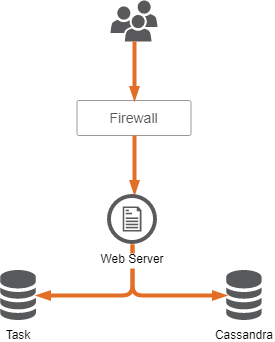
## Internal to application

* 1433 - Used by web servers to access Task Database servers
* 9042, 9160 - Used by web servers to access Cassandra Databases
* 7000, 7001, 7199 - used for inter-server communication between Cassandra servers.

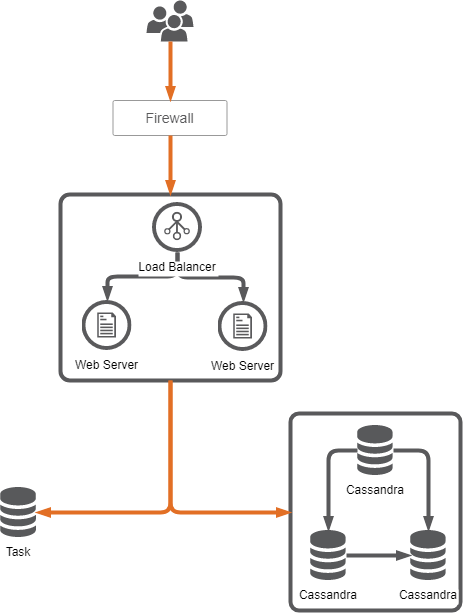
## Integration

Example: 389 - AD Integration used for authentication / and other lookup/updates. (This is a configurable port, and needs to match what the AD system is configured for.)Preferred **Technical** Architecture Diagrams

## Development

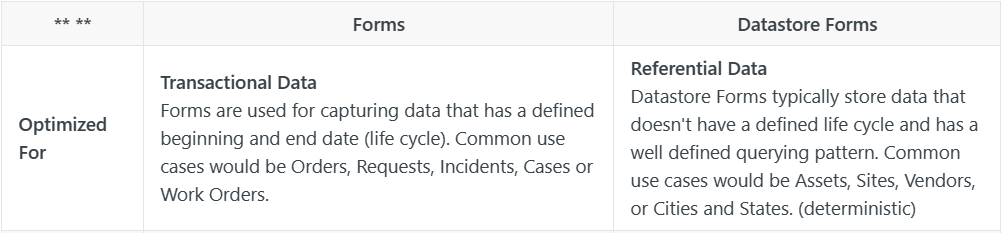
Production

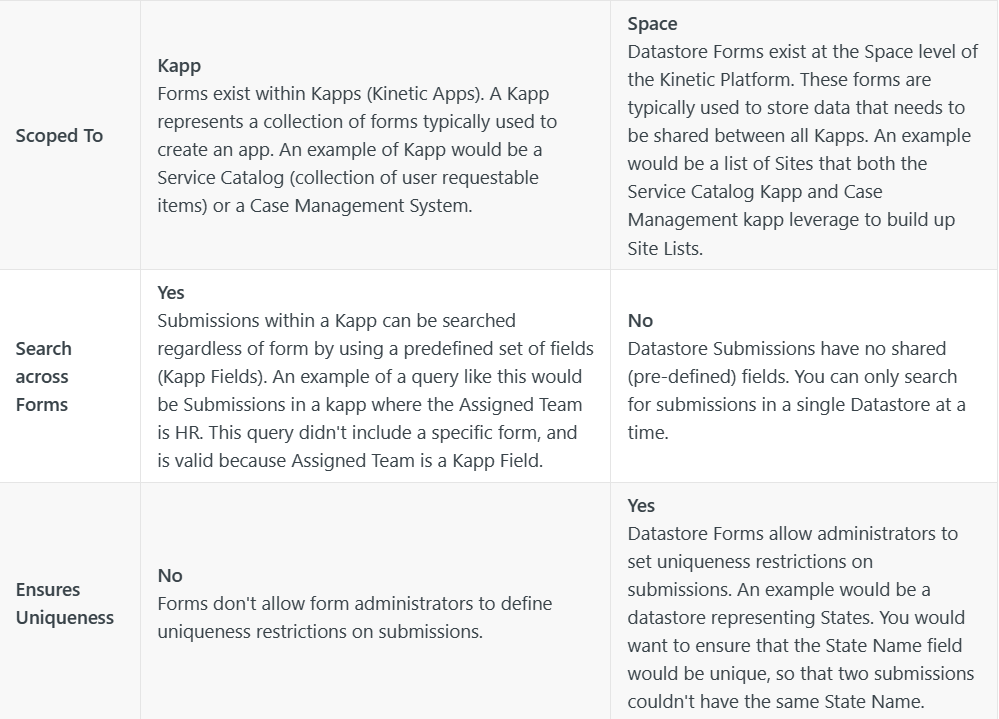


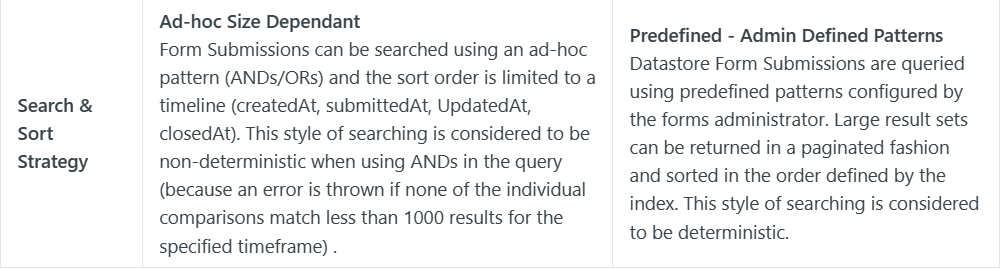
# ***Other Information***

*Forms vs Datastore Form Comparison*

There are a few key differences between *regular* (Kapp) Forms and Datastore Forms. The primary difference is in the way one queries for Submissions (or records created) of one of these forms.

**

**

**

# *Security: -*

Security for platform components is handled by Security Policies. Security policies are something defined by a system admin and can be as simple or complex as necessary. These policies can incorporate a variety of information, including the user's indentity, teams, roles, and more.Security policies can be used, in some cases, to determine access to the platform consoles themselves.

# Creating Security Policies

Security Policies are created in the same manner, regardless of the level they are created for. You may have different information depending on the level, but creating the policy is the same.

Rules are built (written) using KSL, a scripting language created for such things. See the related KSL article for more information on KSL and for sample security policies.

Each Policy Rule console has the same fields:

* Name. Descriptive name for the rule
* Type. Drop-down list of options for the rule depending on where you are in the Request console (Space or Kapp).
* Message. Message presented if the rule resolves to False
* Rule. KSL that resolves to True or False

# Security Policy Precedence

Security policies are defined in different places, depending on where they are applied. If no policy is defined, then the policy is used from the next highest level.

In level order:

1. Form
2. Kapp
3. Space

# Consoles

In general, you need to be a Space Admin to have access to the consoles.

If you allow a user access to Form Modification, they have access to the kapp the form is part of. They can see other consoles in the kapp, but cannot make changes.

Each kapp also has a setting for kapp visibility and kapp modification. While kapp visibility is just to see the forms for the kapp, kapp modification allows a user to have access to the kapp consoles.

kapp security settings are in the Settings console for the kapp, under the Security tab.

You can also see the default values for form security permissions that are set here.

# Form Security Policies

There are four options for Security Policies for Forms. They control form modification and visibility, and submision access and modification.  
Another option is to set the form to anonymous. This option is featured in another article because it's not specifically a security option.

Each of the four options has the same list of choices.

* Form Display.
* Form Modification.
* Subbmission Access.
* Submission Modification.

## *Alternatives & Competitors to Kinetic: -*

## Kinetic stacks up to the competition, the reviews from current & previous users in industries like Machinery, Automotive, and Mechanical or Industrial Engineering, and find the best product for your business.

* SAP ERP software is a proven foundation for the world's largest organizations. Streamline procurement, manufacturing, service, sales, finance, and HR processes.
* NetSuite is a cloud ERP solution, providing a suite of applications, from accounting and financial planning, to warehouse management, ecommerce, inventory management and beyond.
* SYSPRO is a global leader in the production of world-class ERP software, the company now caters to the specialized needs of over 15,000 licensed companies in more than 60 countries worldwide
* Microsoft Dynamics 365 for Operations is the complete ERP solution for enterprises and offers solutions for manufacturing, distribution, retail, services, and public sector.
* Acumatica is a leading provider of Cloud ERP and Cloud Accounting software for mid-size businesses and non-for-profits
* SAP Business One single, integrated solution to manage your entire business

These are some of alternatives or competitors to Kinetic

Reference Links: -

=>[New information in the Library | Kinetic Data Documentation](https://docs.kineticdata.com/docs)

=> [(27) Kinetic Data: Overview | LinkedIn](https://www.linkedin.com/company/kinetic-data/)

=> [The Kinetic Platform | Kinetic Data](https://kineticdata.com/platform/)

……

**\*\*\*\*\*\*\*\*\***