ThreadConnect Integration Platform

Contents

ThreadConnect Integration Platform Overview

About ThreadConnect Integration Platform as a Service

Learn how to create flows to integrate complex enterprise systems, automate actions, and reuse flows with this powerful drag-and-drop tool.

Note: During this Beta release, ThreadConnect Integration Platform as a Service is available only to GE internal users.

ThreadConnect Integration Platform as a Service enables users to integrate cross-cloud or on-premise applications. Users can leverage multiple native connectors to integrate complex enterprise systems, automate, and reuse flows, publish or subscribe to APIs, microservices, business apps, and data. The ThreadConnect drag-and-drop interface is largely self-documenting with integrated online Help and tool tips to simplify the building of powerful, reliable, and scalable integrations and orchestrations through configurable connectors. Users drag and drop connectors and processors to create data flow logic for data aggregation, transformation, and orchestration by abstracting the connectivity complexities. Communication between Enterprise applications, native connectors, and APIs to devices and cloud applications and services is secure and reliable.

Benefits of ThreadConnect

Build powerful and scalable data flows using integration patterns such as aggregation, transformation, and orchestration of data using data flows. Creating custom flows is easy with visual command and control in this drag-and-drop interface.

- A library of more than 200 connectors that can abstract the complexity of connectivity by providing configurable connectors and well documented with context sensitive help
- Inherently asynchronous design allows for very high throughput and natural buffering even as processing and flow rates fluctuate
- With the highly concurrent model, developers don't have to worry about the typical complexities of concurrency
- Guaranteed Delivery for all messages – all messages are reliably sent to desired destinations even if the destination end-point is down, data provenance provides visibility of when message is received and when its delivered
- Real-time and Batch processing – data flows can be real-time such as API driven or event-driven framework and batch jobs for example - that fetch data at set intervals from database (data sources) and orchestrate data to target end points
- Visual data lineage and provenance for all design and run-time activities shows the points at which data enters and exits the system, as well as how it flows through so the data path is well understood and easily tracked

ThreadConnect Integration Platform Key Features

Learn how these key features can help you create robust flows to integrate enterprise systems using ThreadConnect Integration Platform as a Service.

Flow Management

Guaranteed Delivery

A core philosophy of ThreadConnect has been that even at very high scale, guaranteed delivery is a must. This is achieved through effective use of a purpose-built, persistent write-ahead log and

content repository. Together they are designed in such a way as to allow for very high transaction rates, effective load-spreading, copy-on-write, and play to the strengths of traditional disk read/write operations.

Data Buffering with Back Pressure and Pressure Release

ThreadConnect supports buffering of all queued data, as well as the ability to provide back pressure as those queues reach specified limits or to age off data as it reaches a specified age (its value has perished).

Prioritized Queuing

ThreadConnect allows the setting of one or more prioritization schemes for how data is retrieved from a queue. The default is oldest first, but there are times when data should be pulled newest first, largest first, or some other custom scheme.

Flow-Specific QoS

There are points of a dataflow where the data is absolutely critical and it is loss intolerant. There are also times when it must be processed and delivered within seconds to be of any value. ThreadConnect enables the fine-grained flow specific configuration of latency, throughput, and loss tolerance.

Ease of Use

Visual Command and Control

Dataflows can become quite complex. Being able to visualize those flows and express them visually can help greatly to reduce that complexity and to identify areas that need to be simplified. ThreadConnect enables not only the visual establishment of dataflows but it does so in real-time. Rather than being design and deploy it is much more like molding clay. If you make a change to the dataflow that change immediately takes effect. Changes are fine-grained and isolated to the affected components. You don't need to stop an entire flow or set of flows just to make some specific modification.

Flow Templates

Dataflows tend to be highly pattern oriented and, while there are often many different ways to solve a problem, it helps greatly to be able to share those best practices. Templates allow subject matter experts to build and publish their flow designs and for others to benefit and collaborate on them.

Data Provenance

ThreadConnect automatically records, indexes, and makes available provenance data as objects flow through the system even across fan-in, fan-out, transformations, and more. This information becomes extremely critical in supporting compliance, troubleshooting, optimization, and other scenarios.

Recovery and Recording a Rolling Buffer of Fine-grained History

The ThreadConnect content repository is designed to act as a rolling buffer of history. Data is removed only as it ages off the content repository or as space is needed. This combined with the data provenance capability makes for an incredibly useful basis to enable click-to-content, download of content, and replay, all at a specific point in an object's life cycle which can even span generations.

Security

System to System

A dataflow is only as good as it is secure. ThreadConnect at every point in a dataflow offers secure exchange through the use of protocols with encryption such as 2-way SSL. In addition ThreadConnect enables the flow to encrypt and decrypt content and use shared-keys or other mechanisms on either side of the sender/recipient equation.

User to System

ThreadConnect enables 2-Way SSL authentication and provides pluggable authorization so that it can properly control a user's access and at particular levels (read-only, dataflow manager, admin). If a user

enters a sensitive property like a password into the flow, it is immediately encrypted server side and never again exposed on the client side even in its encrypted form.

Flexible Scaling Model

Scale-out (Clustering)

ThreadConnect is designed to scale-out through the use of clustering many nodes together as described above. If a single node is provisioned and configured to handle hundreds of MB per second, then a modest cluster could be configured to handle GB per second. This then brings about interesting challenges of load balancing and fail-over between ThreadConnect and the systems from which it gets data. Use of asynchronous queuing based protocols like messaging services, Kafka, etc., can help.

Scale-up and Sale-down

ThreadConnect is also designed to scale-up and down in a very flexible manner. In terms of increasing throughput from the standpoint of the ThreadConnect framework, it is possible to increase the number of concurrent tasks on the processor under the Scheduling tab when configuring. This allows more processes to execute simultaneously, providing greater throughput. On the other side of the spectrum, you can perfectly scale ThreadConnect down to be suitable to run on edge devices where a small footprint is desired due to limited hardware resources. To specifically solve the first mile data collection challenge and edge use cases, you can find more details

Getting Started with ThreadConnect Integration Platform as a Service

ThreadConnect Integration Platform Setup

Create your secure environment and prepare to create a ThreadConnect service instance.

To use the ThreadConnect Integration Platform service, you must first subscribe to the Predix User Account and Authentication (UAA) service as the trusted issuer. You can use the UAA dashboard or the Cloud Foundry command line interface to create and configure your service.

Setup Task Roadmap

Note:

If you have used other services from the Predix Catalog, you probably have UAA services set up and can use the existing UAA credentials. You can create up to 10 service instances from the same UAA client. If this is the case, you can start with Step 4 in the Setup Task Roadmap to create your TheradConnect service instance.

#	Task	Description
1	(Optional) Configure your proxy settings.	Depending on your location and network configuration, you might need to configure your proxy settings to access remote resources. See Defining Proxy Connections to Remote Resources.
2	Subscribe to the User Account and Authentication service and Create the UAA service instance as the trusted issuer.	1. Log into your Predix account at `https://www.predix.io`. 2. Navigate to the **Catalog** > **Services** tab and click the User Account and Authentication tile. 3. Click **Subscribe** on the required plan to open the UAA dashboard where you will set up your UAA service instance. See Creating a UAA Service Instance You can create a maximum of 10 UAA instances in your space. As a best practice, use the same UAA instance for your services. Optionally, you may use the Cloud Foundry command-line interface (CLI) to create your UAA instance.
3	Configure your UAA service instance .	In the UAA dashboard, click **Configure** button to continue your setup. In this dashboard you can configure the OAuth2 client and, optionally, create user groups, and add users to the groups.
	Create and configure Oauth2 clients to set up access to your service authenticated using UAA.	When you create a UAA instance, an admin client is automatically created for you to access UAA for additional configuration. You must also create a new client for your service instance with specific scopes. See If an Oauth2 client already exists, you can update the client to add your service instance. See Creating an OAuth2 Client.

#	Task	Description
4	Bind your application to the service instance.	See Binding an Application to the UAA Instance.
5	Configure your ThreadConnect service instance.	Continue in Creating a ThreadConnect Service Instance on page 6

Configuring OAuth2 Clients to Provide Users Access to the ThreadConnect Integration Platform as a Service

Create OAtuth2 clients for foundation access to the Thread Connect Integration Platform.

About This Task

You must create and configure OAuth2 Clients to provide users access to the ThreadConnect Integration Platform.

Before You Begin

You must have configured UAA and created the OAuth2 Client.

Procedure

1. Create a new client by selecting the **Create Client** button.
2. In the **Authorized Grant Types** tab, deselect **client_credentials**. Then, select **authorization_code** and **refresh_token**.
3. Enter a value for the Client id.
4. Enter the Client Secret.
5. Enter **openid** in the Scope field.
6. Enter **openid** in the **Auto Approved Scopes** field.
7. In the **Redirect URI** field, add this value: **http*://*.digital.ge.com/****

Related Tasks

Creating a ThreadConnect Service Instance on page 6

Create a service instance of ThreadConnect Integration Platform as a Service and start creating robust flows.

Creating User Access to ThreadConnect Integration Platform as a Service

Create user definitions for the clients you have set up for access to the ThreadConnect Integration Platform.

About This Task

You must create and create users associated with the OAuth2 clients for access to the ThreadConnect Integration Platform.

Before You Begin

You must have configured UAA and created the OAuth2 Client.

Procedure

1. Create a new user by selecting the **Create User** button.
2. Enter the desired user name.
3. Enter the user's email address.
4. Enter the user's password that will be used to log in to the ThreadConnect service instance.

   Be sure to remember the client and User information; you will need it to create a ThreadConnect service isntance.

Creating a ThreadConnect Service Instance

Create a service instance of ThreadConnect Integration Platform as a Service and start creating robust flows.

Before you Begin ...

Before creating a service instance for ThreadConnect you must have the base URLs for UAA instances that the ThreadConnect service instance will trust. See ThreadConnect Integration Platform Setup on page 4.

Procedure

1. Sign into your Predix account at `https://www.predix.io`.
2. Navigate to **Catalog** > **Services** tab, and click the **ThreadConnect** service tile.
3. Click **Subscribe** on the required plan.
4. On the new Service Instance page, enter the following information:

Field	Description
Org	Select your org.
Space	Select the space for your application.
User Account & Authentication (UAA)	Choose an existing UAA instance or create a new instance of UAA. For more information, see Creating a UAA Service Instance
Service instance name	Specify a unique name for your instance.
Service Plan	Select a plan.
Client id	Enter the Client ID.
clientSecret	Enter the clientSecret you defined earlier.
identityZoneId	Enter your UAA identity Zone ID used to create the UAA service instance.

Using Cloud Foundry Commands to Create your ThreadConnect Service Instance

Procedure

1. List the services in the Cloud Foundry marketplace.

   ```
   cf marketplace
   ```

   The ThreadConnect service, `thread-connect-service`, is listed as an available service.

2. Create an ThreadConnect service instance.

```
cf create-service threadconnect <plan>
<my_threadconnect_service_instance>  -c
                    '{"trustedIssuerIds":["https://<predix-uaa-
instance-uri>/oauth/token"],
                    "clientId":"<clientId>",
"clientSecret":"<clientSecret>",
                    "identityZoneId":"<predix-uaa-instance-guid>"}'
```

   where:

   <plan> is the plan associated with a service.

3. Bind your ThreadConnect gateway service instance to your application to provision connection details for your service instance in the VCAP environment variables. Cloud Foundry runtime uses VCAP_SERVICES environment variables to communicate with a deployed application about its environment.

   Use the Cloud Foundry CLI to log into Cloud Foundry:

```
cf login
```

4. Bind your application to the service instance you created.

```
cf bind-service <application_name> <my_threadconnect_instance>
```

5. Restage your application to ensure that environment variable changes take effect:

```
cf restage <application_name>
```

6. View the environment variables for your application:

```
cf env <application_name>
```

   The environment variables which contain your basic authorization credentials are shown:client ID and the endpoint URI:

```
{
 "VCAP_SERVICES": {
  "threadconnect": [
   {
    "credentials": {
     "uri": "https://tc-service-proxy.run.aws-usw02-
pr.ice.predix.io",
     "zone": {
      "http-header-name": "Predix-Zone-Id",
      "http-header-value": "48767c33-974e-434a-a687-5bc08ba6cef1",
      "oauth-scope": "threadconnect.zones.48767c33-974e-434a-
a687-5bc08ba6cef1.user"
     }
    },
    "label": "threadconnect",
    "name": "tc-service-test2",
    "plan": "Beta",
    "provider": null,
    "syslog_drain_url": null,
    "tags": [],
    "volume_mounts": []
   }
  ]
```

```
  }
}
```

To access your ThreadConnect instance, you must navigate to your unique Dashboard URL, which can be found with the following command: `cf service <my_threadconnect_instance>`

After you navigate to this URL, you will be asked for your login credentials, which will be the same as the User that you set up in your Trusted Issuer.

7. Add the `oauth-scope` to your client or user in the UAA Authority.
8. Add the ThreadConnect scope to the UAA instance.

After you submit the command to create your service instance, the system creates your new platform. This process typically requires 10 to 15 minutes. If you try to access the UI before the platform is fully created, you will receive a connection error. If after 20 minutes you still cannot access the UI for the platform, contact the Support team at DCThreadConnectPredixSupport@ge.com