# Implementing ADFS Single Sign-on (SSO) Integration

* [Goal](#ImplementingADFSSingleSign-on(SSO)Integ)
* [Prerequisites](#ImplementingADFSSingleSign-on(SSO)Integ)
* [Implementation Steps](#ImplementingADFSSingleSign-on(SSO)Integ)
  + [A. The ADFS Configuration](#ImplementingADFSSingleSign-on(SSO)Integ)
  + [B. The Incorta Configuration](#ImplementingADFSSingleSign-on(SSO)Integ)
    - [1. Enable SSO for a Tenant](#ImplementingADFSSingleSign-on(SSO)Integ)
    - [2. Create a Configuration file](#ImplementingADFSSingleSign-on(SSO)Integ)
    - [3. Modify Incorta server.xml](#ImplementingADFSSingleSign-on(SSO)Integ)
    - [4. Restart Incorta](#ImplementingADFSSingleSign-on(SSO)Integ)
* [Verify the connection](#ImplementingADFSSingleSign-on(SSO)Integ)
  + [known Issues](#ImplementingADFSSingleSign-on(SSO)Integ)
    - [ADFS Supported Browsers](#ImplementingADFSSingleSign-on(SSO)Integ)
* [Related articles](#ImplementingADFSSingleSign-on(SSO)Integ)

Product documentation now contains this information. Please refer to it and [log a DOC Jira ticket](https://incorta.atlassian.net/wiki/wiki/pages/createpage.action?spaceKey=INC&title=Create%20a%20Documentation%20Request%2FTicket&linkCreation=true&fromPageId=591626244) if it needs updated. Thanks.

Not deleting the following content because it may contain customer-specific info. Needs review - HRK

# Goal

To enable single sign-on using Active Directory Federation Services (ADFS), configure ADFS and Incorta.

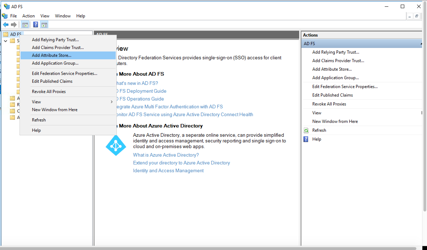
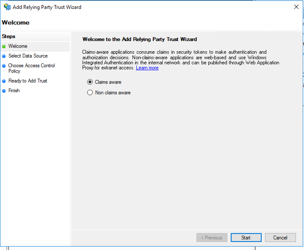
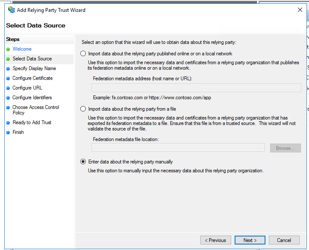
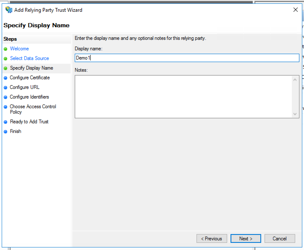
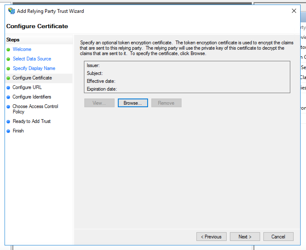
# Prerequisites

 ADFS only accepts https URLs. You must set your Incorta instance and any other solutions you use with Incorta to https. Please refer to [DELETE - Enable SSL Certificate in 4.x](https://incorta.atlassian.net/wiki/wiki/spaces/INC/pages/786169858/DELETE+-+Enable+SSL+Certificate+in+4.x) or [DELETE - Secure Tomcat with TLS/SSL](https://incorta.atlassian.net/wiki/wiki/spaces/INC/pages/282099770/DELETE+-+Secure+Tomcat+with+TLS+SSL) for 3.x

# Implementation Steps

The steps are configuration parts that need to be done in the ADFS and in Incorta.

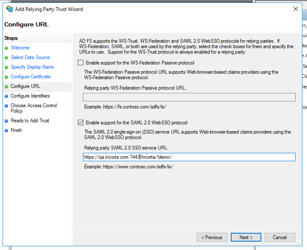
## A. The ADFS Configuration

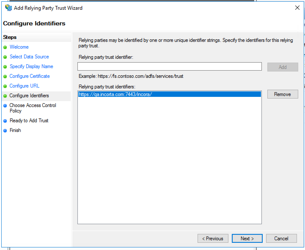
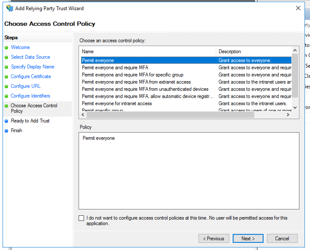
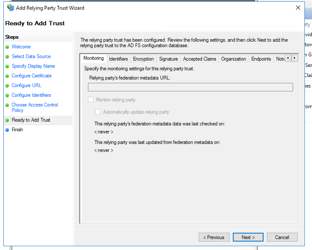
1. ADFS Manager
   1. From ADFS manager right click on **ADFS** and then select **Add Relaying Party Trust**.    
      
   2. Select **Claims aware,** then **Start.**  
      
   3. Select **Enter data about the relaying party Manually,** then **Next**.  
      
   4. Enter a display name and select **Next.**   
      
   5. Browse to an encryption certificate or select **Next** to continue without an encryption certificate.  
      
   6. Select **Enable for the SAML 2.0 WebSSO protocol**.
   7. Enter the Incorta SSO link in the format and select Next

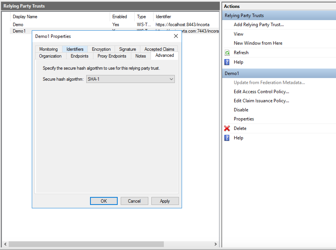
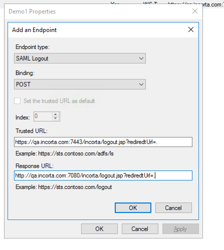
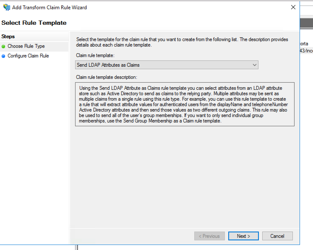
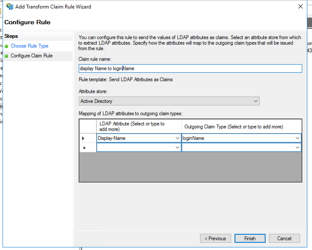
SSO Tenant Format

https://<incorta-server>/incorta/!<tenant-name>/

For example, <https://qa.incorta.com/incorta/!demo/>



* 1. Add a relying identifier. For example, enter the Incorta URL [https://incorta.com/incorta](https://localhost:8443/incorta/!demo/). Select **Next.**   
     
  2. Select **Permit everyone,** then **Next.**  
     
  3. Select **Next**.   
     
  4. Select **Finish**.

1. Select the relying parts in the left panel and the relying party you created and select properties in the right panel.
2. Select the **Advance** tab, then SHA-1 in **Secure hash algorithm.**    
   
3. Select the **Endpoint** tab.
4. Select **Add.**
5. Select **SAML logout** as the endpoint type and enter a URL in the format https://<server-name>/incorta/logout.jsp?rediredtUrl=.  In the **Trusted URL** field and the **Response URL** field.  
   
6. Select **Add Claim**.
7. Select **Send LDAP Attributes as Claim** from **Claim rule template** and select **Next**.  
   
8. Enter a **Claim rule name**. From **LDAP Attributes,** select Display-Name and login name from the **Outgoing claim type**. Select **Finish**.  
   

## B. The Incorta Configuration

In Incorta, these steps are needed:

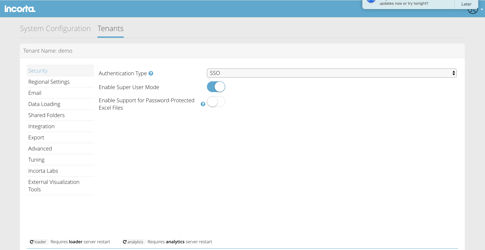
1. Enable SSO for a Tenant
2. Create a Configuration file and refer to it in the Incorta Server.xml file
3. Modify server.xml to refer to the configuration file.
4. Restart Incorta.

### 1. Enable SSO for a Tenant

Enable Tenant for single sign-on can be done using any of the two alternatives:

* The administration screen (Incorta Admin UI)

1. Navigate to Incorta Admin UI
2. Select the Tenants tab.
3. Select a Tenant.



* Using the tmt command line

For incorta 3.x

./tmt.sh  --update-property <tenantname> sso-login-enable true

Example: ./tmt.sh  --update-property demo sso-login-enable true

For incorta 4.x

./tmt.sh  --clnm <CLUSTER\_NAME> --update-property <tenantname> sso-login-enable true

Example: ./tmt.sh -clnm  localCluster  --update-property demo sso-login-enable true

### 2. Create a Configuration file

Below is a sample configuration file. 

Change the following properties:

* onelogin.saml2.sp.entityid: the value of the "relying identifier" in ADFS.
* onelogin.saml2.sp.assertion\_consumer\_service.url: The value of **Relying party SAML 2.0 SSO Service URL** in ADFS.
  + Follow this format: [https://<incorta-server>/incorta/!<tenant-name>/](https://localhost:8443/incorta/!demo/)
* onelogin.saml2.sp.single\_logout\_service.url: A URL formatted like this: http:///<incortaHostName>/incorta/logout.jsp?rediredtUrl=.
* onelogin.saml2.idp.entityid: The value of the entityID attribute in ADFS metadata file (can be found in the following URL [https://<adfs-server>/FederationMetadata/2007-06/FederationMetadata.xml](https://adcs-setup.adfs.incortaops.com/FederationMetadata/2007-06/FederationMetadata.xml)).
* onelogin.saml2.idp.single\_sign\_on\_service.url: The value [https://](https://adcs-setup.adfs.incortaops.com/adfs/ls/idpinitiatedsignon.aspx)[<adfs-server>](https://adcs-setup.adfs.incortaops.com/FederationMetadata/2007-06/FederationMetadata.xml)/adfs/ls/idpinitiatedsignon.aspx.
* onelogin.saml2.idp.single\_logout\_service.url: The value [https://](https://adcs-setup.adfs.incortaops.com/adfs/ls/?wa=wsignout1.0)[<adfs-server>/](https://adcs-setup.adfs.incortaops.com/FederationMetadata/2007-06/FederationMetadata.xml)adfs/ls/?wa=wsignout1.0 .
* onelogin.saml2.idp.x509cert: The value of "X509Certificate" in ADFS metadata file (can be found in the following URL [https://<adfs-server>/FederationMetadata/2007-06/FederationMetadata.xml](https://adcs-setup.adfs.incortaops.com/FederationMetadata/2007-06/FederationMetadata.xml)).

Fields highlighted in B are customer-specific.

------ Beginning of the File   /home/incorta/IncortaAnalytics/sso/ssoDemoConf.properties     --------

# If 'strict' is True, then the Java Toolkit will reject unsigned  
# or unencrypted messages if it expects them signed or encrypted  
# Also will reject the messages if not strictly follow the SAML  
onelogin.saml2.strict = false

# Enable debug mode (to print errors)  
onelogin.saml2.debug = true

# Service Provider Data that we are deploying  
#

# Identifier of the SP entity (must be a URI)  
onelogin.saml2.sp.entityid = <https://localhost:8443/incorta>

# Specifies info about where and how the <AuthnResponse> message MUST be  
# returned to the requester, in this case our SP.  
# URL Location where the <Response> from the IdP will be returned  
onelogin.saml2.sp.assertion\_consumer\_service.url = <https://localhost:8443/incorta/!demo/>

# SAML protocol binding to be used when returning the <Response>  
# message. Onelogin Toolkit supports for this endpoint the  
# HTTP-POST binding only  
onelogin.saml2.sp.assertion\_consumer\_service.binding = urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Redirect

# Specifies info about where and how the <Logout Response> message MUST be  
# returned to the requester, in this case our SP.  
onelogin.saml2.sp.single\_logout\_service.url = <https://localhost:8443/incorta/logout.jsp?rediredtUrl=>.

# SAML protocol binding to be used when returning the <LogoutResponse> or sending the <LogoutRequest>  
# message. Onelogin Toolkit supports for this endpoint the  
# HTTP-Redirect binding only  
onelogin.saml2.sp.single\_logout\_service.binding = urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Redirect

# Specifies constraints on the name identifier to be used to  
# represent the requested subject.  
# Take a look on lib/Saml2/Constants.php to see the NameIdFormat supported  
#onelogin.saml2.sp.nameidformat = urn:oasis:names:tc:SAML:1.1:nameid-format:unspecified

# Usually x509cert and privateKey of the SP are provided by files placed at  
# the certs folder. But we can also provide them with the following parameters

onelogin.saml2.sp.x509cert =  
# Requires Format PKCS#8 BEGIN PRIVATE KEY   
# If you have PKCS#1 BEGIN RSA PRIVATE KEY convert it by openssl pkcs8 -topk8 -inform pem -nocrypt -in sp.rsa\_key -outform pem -out sp.pem  
onelogin.saml2.sp.privatekey =

# Identity Provider Data that we want connect with our SP  
#

# Identifier of the IdP entity (must be a URI)  
onelogin.saml2.idp.entityid =<http://adcs-setup.adfs.incortaops.com/adfs/services/trust>

# SSO endpoint info of the IdP. (Authentication Request protocol)  
# URL Target of the IdP where the SP will send the Authentication Request Message  
onelogin.saml2.idp.single\_sign\_on\_service.url = <https://adcs-setup.adfs.incortaops.com/adfs/ls/idpinitiatedsignon.aspx>  
onelogin.saml2.security.want\_nameid = false  
# SAML protocol binding to be used when returning the <Response>  
# message. Onelogin Toolkit supports for this endpoint the  
# HTTP-Redirect binding only  
onelogin.saml2.idp.single\_sign\_on\_service.binding = urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Redirect  
#if the above did not work try the below  
#onelogin.saml2.idp.single\_sign\_on\_service.binding = urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST

# SLO endpoint info of the IdP.  
# URL Location of the IdP where the SP will send the SLO Request  
#onelogin.saml2.idp.single\_logout\_service.url = <https://incorta-dev.onelogin.com/trust/saml2/http-redirect/slo/610260>  
onelogin.saml2.idp.single\_logout\_service.url = <https://adcs-setup.adfs.incortaops.com/adfs/ls/?wa=wsignout1.0>

# Optional SLO Response endpoint info of the IdP.  
# URL Location of the IdP where the SP will send the SLO Response. If left blank, same URL as onelogin.saml2.idp.single\_logout\_service.url will be used.  
# Some IdPs use a separate URL for sending a logout request and response, use this property to set the separate response url  
onelogin.saml2.idp.single\_logout\_service.response.url =

# SAML protocol binding to be used when returning the <Response>  
# message. Onelogin Toolkit supports for this endpoint the  
# HTTP-Redirect binding only  
onelogin.saml2.idp.single\_logout\_service.binding = urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Redirect

# Public x509 certificate of the IdP  
onelogin.saml2.idp.x509cert =  
# Instead of use the whole x509cert you can use a fingerprint  
# (openssl x509 -noout -fingerprint -in "idp.crt" to generate it,  
# or add for example the -sha256 , -sha384 or -sha512 parameter)  
#  
# If a fingerprint is provided, then the certFingerprintAlgorithm is required in order to  
# let the toolkit know which Algorithm was used. Possible values: sha1, sha256, sha384 or sha512  
# 'sha1' is the default value.  
# onelogin.saml2.idp.certfingerprint =   
# onelogin.saml2.idp.certfingerprint\_algorithm = sha1

# Security settings  
#

# Indicates that the nameID of the <samlp:logoutRequest> sent by this SP  
# will be encrypted.  
onelogin.saml2.security.nameid\_encrypted = false

# Indicates whether the <samlp:AuthnRequest> messages sent by this SP  
# will be signed. [The Metadata of the SP will offer this info]  
onelogin.saml2.security.authnrequest\_signed = false

# Indicates whether the <samlp:logoutRequest> messages sent by this SP  
# will be signed.  
onelogin.saml2.security.logoutrequest\_signed = false

# Indicates whether the <samlp:logoutResponse> messages sent by this SP  
# will be signed.  
onelogin.saml2.security.logoutresponse\_signed = false

# Sign the Metadata  
# Empty means no signature, or comma separate the keyFileName and the certFileName  
onelogin.saml2.security.want\_messages\_signed =

# Indicates a requirement for the <samlp:Response>, <samlp:LogoutRequest> and  
# <samlp:LogoutResponse> elements received by this SP to be signed.  
onelogin.saml2.security.want\_assertions\_signed = false

# Indicates a requirement for the Metadata of this SP to be signed.  
# Right now supported null (in order to not sign) or true (sign using SP private key)   
onelogin.saml2.security.sign\_metadata = false

# Indicates a requirement for the Assertions received by this SP to be encrypted  
onelogin.saml2.security.want\_assertions\_encrypted = false

# Indicates a requirement for the NameID received by this SP to be encrypted  
onelogin.saml2.security.want\_nameid\_encrypted = false

# Authentication context.  
# Set Empty and no AuthContext will be sent in the AuthNRequest,  
# Set comma separated values urn:oasis:names:tc:SAML:2.0:ac:classes:urn:oasis:names:tc:SAML:2.0:ac:classes:Password  
#onelogin.saml2.security.requested\_authncontext = urn:oasis:names:tc:SAML:2.0:ac:classes:urn:oasis:names:tc:SAML:2.0:ac:classes:Password  
#onelogin.saml2.security.requested\_authncontext = urn:oasis:names:tc:SAML:2.0:ac:classes:urn:oasis:names:tc:SAML:2.0:ac:classes:unspecified,urn:oasis:names:tc:SAML:2.0:ac:classes:urn:oasis:names:tc:SAML:2.0:ac:classes:Password,urn:oasis:names:tc:SAML:2.0:ac:classes:urn:oasis:names:tc:SAML:2.0:ac:classes:PasswordProtectedTransport,urn:oasis:names:tc:SAML:2.0:ac:classes:urn:oasis:names:tc:SAML:2.0:ac:classes:TLSClient,urn:oasis:names:tc:SAML:2.0:ac:classes:urn:oasis:names:tc:SAML:2.0:ac:classes:X509,urn:oasis:names:tc:SAML:2.0:ac:classes:urn:federation:authentication:windows,urn:oasis:names:tc:SAML:2.0:ac:classes:urn:oasis:names:tc:SAML:2.0:ac:classes:Kerberos  
#onelogin.saml2.security.requested\_authncontext = urn:oasis:names:tc:SAML:2.0:ac:classes:urn:oasis:names:tc:SAML:2.0:ac:classes:Password,urn:oasis:names:tc:SAML:2.0:ac:classes:urn:oasis:names:tc:SAML:2.0:ac:classes:PasswordProtectedTransport,urn:oasis:names:tc:SAML:2.0:ac:classes:urn:oasis:names:tc:SAML:2.0:ac:classes:TLSClient,urn:oasis:names:tc:SAML:2.0:ac:classes:urn:oasis:names:tc:SAML:2.0:ac:classes:X509,urn:oasis:names:tc:SAML:2.0:ac:classes:urn:federation:authentication:windows,urn:oasis:names:tc:SAML:2.0:ac:classes:urn:oasis:names:tc:SAML:2.0:ac:classes:Kerberos  
onelogin.saml2.security.requested\_authncontext = urn:oasis:names:tc:SAML:2.0:ac:classes:Password,urn:oasis:names:tc:SAML:2.0:ac:classes:PasswordProtectedTransport,urn:oasis:names:tc:SAML:2.0:ac:classes:TLSClient,urn:oasis:names:tc:SAML:2.0:ac:classes:X509,urn:federation:authentication:windows,urn:oasis:names:tc:SAML:2.0:ac:classes:Kerberos

# Allows the authn comparison parameter to be set, defaults to 'exact'  
#onelogin.saml2.security.requested\_authncontextcomparison = exact  
onelogin.saml2.security.requested\_authncontextcomparison = minimum

# Indicates if the SP will validate all received xmls.  
# (In order to validate the xml, 'strict' and 'wantXMLValidation' must be true).  
onelogin.saml2.security.want\_xml\_validation = true

# Algorithm that the toolkit will use on signing process. Options:  
# '<http://www.w3.org/2000/09/xmldsig#rsa-sha1>'  
# '<http://www.w3.org/2000/09/xmldsig#dsa-sha1>'  
# '<http://www.w3.org/2001/04/xmldsig-more#rsa-sha256>'  
# '<http://www.w3.org/2001/04/xmldsig-more#rsa-sha384>'  
# '<http://www.w3.org/2001/04/xmldsig-more#rsa-sha512>'  
onelogin.saml2.security.signature\_algorithm = <http://www.w3.org/2000/09/xmldsig#rsa-sha1>

# Organization  
[onelogin.saml2.organization.name](http://onelogin.saml2.organization.name) = SP Java   
onelogin.saml2.organization.displayname = SP Java Example  
onelogin.saml2.organization.url = <http://sp.example.com>

# Contacts  
onelogin.saml2.contacts.technical.given\_name = Technical Guy  
onelogin.saml2.contacts.technical.email\_address = [technical@example.com](mailto:technical@example.com)  
onelogin.saml2.contacts.support.given\_name = Support Guy  
onelogin.saml2.contacts.support.email\_address = support@@[example.com](http://example.com)

                                          --------- End of File ------------

### 3. Modify Incorta server.xml

Modify server.xml to add the following tag at the beginning of <Host> tag:

* For incorta 3.x , the Server.xml  is located in <incorta home>/server/Conf) to add the following tag at the beginning of <Host> tag
* For incorta 4.x , the Server.xml is located on the Analytic Service  at <incorta\_home>/IncortaNode/services/<service\_id>/conf

 ```

 <Valve className="com.incorta.sso.valves.OneLoginValve"

 confFilesMap="Tenant\_Name=Absolute\_Path,Tenant\_Name2=Absolute\_Path2"

 LoggingEnabled = "true"

 />

 ```

   Where:

1. Tenant\_Name is the name of Incorta Tenant
2. Absolute\_Path is the path of the SSO configuration file
3. LoggingEnabled  This flag turns on the valve logging messages. By default, it's false which mean the logging is turned off

Example  :

<Valve className="com.incorta.sso.valves.OneLoginValve"

 confFilesMap="demo=/home/incorta/IncortaAnalytics/sso/ssoDemoConf.properties"

 LoggingEnabled = "true"

 />

### 4. Restart Incorta

* For Incorta 3.x, Restart incorta service
* For Incorta 4.x, the analytic service needs to be restarted

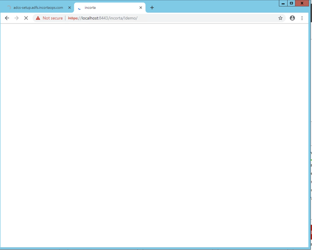
# Verify the connection

Launch incorta URL for the SSO enabled tenant

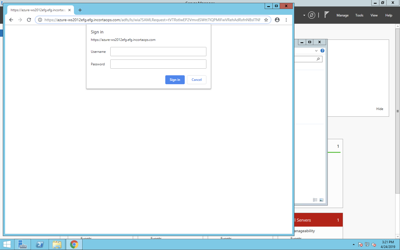
https://server\_ip:https\_port/incorta/!tenant\_name/

ex:

<https://localhost:8443/incorta/!demo/>



Since incorta is a non microsoft application a login popup form is shown at the first time login only



## known Issues

### ADFS Supported Browsers

ADFS property called “WIASupportedUserAgents”. Which means the supported browsers that are allowed for WIA(Windows Integrated Authentication). If your users will be using an IE Edge or the newer version of Chrome, you will need to this additional step in The ADFS Configuration after step 9

Run the following in PowerShell:

Set-ADFSProperties -WIASupportedUserAgents @("MSIE 6.0", "MSIE 7.0", "MSIE 8.0", "MSIE 9.0", "MSIE 10.0", "MSIE 11.0", "Trident/7.0", "MSIPC", "Windows Rights Management Client", "Mozilla/5.0")

Then restarted the ADFS service.

For more information please refer to this article

[ADFS: Enable Single Sign-on (SSO) for Edge and Chrome browser](https://www.vspbreda.nl/nl/ms-office/office-365/solved-adfs-enable-single-sign-on-sso-for-edge-and-chrome-browser/)

# Related articles

<https://www.vspbreda.nl/nl/ms-office/office-365/solved-adfs-enable-single-sign-on-sso-for-edge-and-chrome-browser/>

* Page:

[Inspector Executor](https://incorta.atlassian.net/wiki/wiki/spaces/INC/pages/4726751247/Inspector+Executor)

* Page:

[Automatic Table Selection](https://incorta.atlassian.net/wiki/wiki/spaces/INC/pages/4473487372/Automatic+Table+Selection)

* Page:

[How To Change The Thread Pool Used for ETL Steps](https://incorta.atlassian.net/wiki/wiki/spaces/INC/pages/4656824321/How+To+Change+The+Thread+Pool+Used+for+ETL+Steps)

* Page:

[Geminga](https://incorta.atlassian.net/wiki/wiki/spaces/INC/pages/4575166466/Geminga)

* Page:

[Supporting Multi Source data](https://incorta.atlassian.net/wiki/wiki/spaces/INC/pages/3164241946/Supporting+Multi+Source+data)