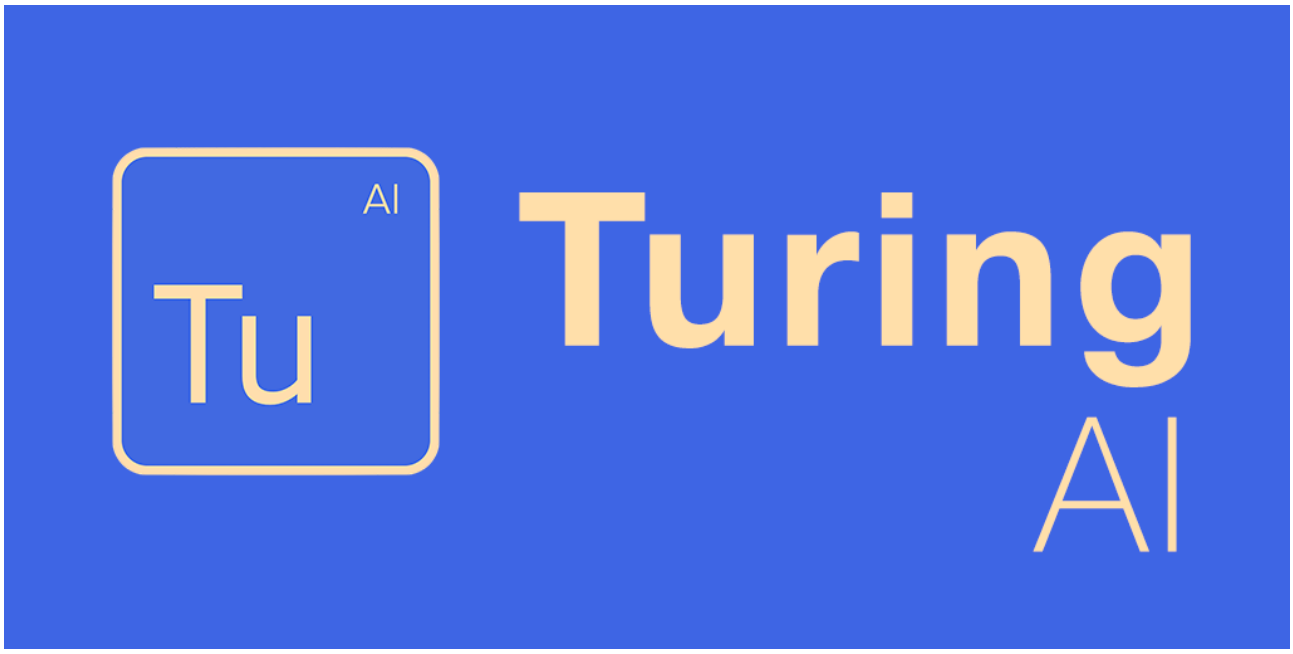

Viglet Turing AI
Connectors

Viglet Team

Version 0.3.4, 05-08-2021

Table of Content

Preface

There are several connectors to allow you to index content in Viglet Turing AI.

Chapter 1. Apache Nutch

Plugin for Apache Nutch to index content using crawler.

1.1. Installation

Turing support Apache Nutch 1.12 and 1.8 only, so go to https://viglet.com/turing/download/ and click on "Integration > Apache Nutch" link to download the turing-nutch-<NUTCH_RELEASE>-bin.zip.

1. Extract turing-nutch-<NUTCH_RELEASE>-bin.zip file into /appl/viglet/turing/nutch.

```
mkdir -p /appl/viglet/turing/nutch
unzip turing-nutch.zip -d /appl/viglet/turing/nutch
```

2. Download and install Apache Nutch 1.12 or 1.18 binary into http://nutch.apache.org > Downloads > apache-nutch-<NUTCH_RELEASE>-bin.tar.gz.

```
mkdir -p /appl/apache/
cp apache-nutch-<NUTCH_RELEASE>-bin.tar.gz /appl/apache
cd /appl/apache
tar -xvzf apache-nutch-<NUTCH_RELEASE>-bin.tar.gz
ln -s apache-nutch-<NUTCH_RELEASE>-bin nutch
```

3. Copy the Turing Plugin to Apache Nutch.

```
cp -R /appl/viglet/turing/nutch/indexer-viglet-turing /appl/apache/nutch/plugins
cp -f /appl/viglet/turing/nutch/conf/* /appl/apache/nutch/conf/
```

1.2. Configuration

1.2.1. Nutch 1.12

This step is only for Apache Nutch 1.12. Edit the /appl/apache/nutch/conf/nutch-site.xml, add or modify the following properties:

```
<property>
  <name>solr.server.url</name>
  <value>http://127.0.0.1:2700/Sample</value>
  <description>
      Turing URL + "/" + Turing Semantic Navigation Site.
  </description>
</property>
<property>
  <name>turing.url</name>
  <value>http://127.0.0.1:2700</value>
  <description>
      Defines the Turing URL into which data should be indexed using the
      indexer-turing plugin.
  </description>
</property>
<property>
  <name>turing.site</name>
  <value>Sample</value>
  <description>
      Defines the Turing Semantic Navigation Site.
  </description>
</property>
<property>
  <name>turing.auth</name>
  <value>true</value>
  <description>
      Whether to enable HTTP basic authentication for communicating with Turing.
Use the username and password properties to configure your credentials.
  </description>
</property>
<property>
  <name>turing.username</name>
  <value>admin</value>
  <description>
      The username of Turing server.
  </description>
</property>
```

```
<property>
  <name>turing.password</name>
  <value>admin</value>
  <description>
      The password of Turing server.
  </description>
</property>
<property>
  <name>turing.timestamp.field</name>
  <value>modification_date</value>
  <description>
      Field used to store the timestamp of indexing. The default value is "tstamp".
  </description>
</property>
<property>
  <name>turing.field.type</name>
  <value>Page</value>
  <description>
      Type of Content. The default value is "Page".
  </description>
</property>
<property>
  <name>turing.field.source_app</name>
  <value>Nutch</value>
  <description>
      Name of Source Application. The default value is "Nutch".
  </description>
</property>
<!--
<property>
  <name>turing.field.hello</name>
  <value>foo</value>
  <description>
      This a test.
  </description>
</property>
<property>
  <name>turing.field.world</name>
  <value>bar</value>
  <description>
      This is another test.
  </description>
</property>
-->
```

Field with Timestamp

Can specify what is the field will be used to store the timestamp of indexing. The default value is `tstamp`. So modify the value of `turing.timestamp.field` property into nutch-site.xml:

```
<property>
  <name>turing.timestamp.field</name>
  <value>modification_date</value>
  <description>
      Field used to store the timestamp of indexing. The default value is "tstamp".
  </description>
</property>
```

Source App Name

Turing AI Semantic Navigation Site allows to index content from many sources, so can identify where the content was indexed, can specify the name of the source changing the `turing.field.source_app` into nutch-site.xml file. The default value is `Nutch`:

```
<property>
  <name>turing.field.source_app</name>
  <value>Nutch</value>
  <description>
      Name of Source Application. The default value is "Nutch".
  </description>
</property>
```

Fixed Fields

To create new fixed field during indexing, add new properties with prefix `turing.field` + `name of new custom field` into nutch-site.xml file, for example:

```
<property>
  <name>turing.field.hello</name>
  <value>foo</value>
  <description>
      This a test.
  </description>
</property>
<property>
  <name>turing.field.world</name>
  <value>bar</value>
  <description>
      This is another test.
  </description>
</property>
```

IMPORTANT	Need add these fields to Solr schema.xml file and create them in Semantic Navigation Site > Fields

Parameters

Modify the following parameters:

Table 1. nutch-site.xml parameters

Parameter	Description	Default value
solr.server.url	Turing URL + "/" + Turing Semantic Navigation Site.	-
turing.url	Defines the fully qualified URL of Turing AI into which data should be indexed.	http://localhost:2700
turing.site	Turing Semantic Navigation Site Name.	Sample
turing.weight.field	Field's name where the weight of the documents will be written. If it is empty no field will be used.	-
turing.auth	Whether to enable HTTP basic authentication for communicating with Turing AI. Use the `username` and `password` properties to configure your credentials.	true
turing.username	The username of Turing AI server.	admin
turing.password	The password of Turing AI server.	admin
turing.timestamp.field	Field used to store the timestamp of indexing.	tstamp
turing.field.`FIELD_NAME`	Modify or create a custom field during indexing.	-

1.2.2. Nutch 1.18

This step is only for Apache Nutch 1.18. Edit the /appl/apache/nutch/conf/index-writers.xml

```
<writers xmlns="http://lucene.apache.org/nutch"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://lucene.apache.org/nutch index-writers.xsd">
 <writer id="indexer_viglet_turing_1"
        class="com.viglet.turing.nutch.indexwriter.TurNutchIndexWriter">
        <parameters>
            <param name="url" value="http://localhost:2700" />
            <param name="site" value="Sample" />
            <param name="commitSize" value="1000" />
            <param name="weight.field" value=""/>
            <param name="auth" value="true" />
            <param name="username" value="admin" />
            <param name="password" value="admin" />
        </parameters>
        <mapping>
            <copy>
                <field source="content" dest="text"/>
                <!-- <field source="title" dest="title,search"/> -->
            </copy>
            <rename>
                <field source="metatag.description" dest="description" />
                <field source="metatag.keywords" dest="keywords" />
                <field source="metatag.charset" dest="charset" />
            </rename>
            <remove>
                <field source="segment" />
                <field source="boost" />
            </remove>
        </mapping>
    </writer>
</writers>
```

Parameters

Modify the following parameters:

Table 2. index-writers.xml parameters

Parameter	Description	Default value
url	Defines the fully qualified URL of Turing AI into which data should be indexed.	http://localhost:2700
site	Turing Semantic Navigation Site Name.	Sample
weight.field	Field's name where the weight of the documents will be written. If it is empty no field will be used.	-
commitSize	Defines the number of documents to send to Turing AI in a single update batch. Decrease when handling very large documents to prevent Nutch from running out of memory. **Note**: It does not explicitly trigger a server side commit.	1000
auth	Whether to enable HTTP basic authentication for communicating with Turing AI. Use the `username` and `password` properties to configure your credentials.	true
username	The username of Turing AI server.	admin

Parameter	Description	Default value
password	The password of Turing AI server.	admin

1.3. Index a Website

1.3.1. Nutch Command Line

There are many ways to index a website using Apache Nutch. Learn more at https://cwiki.apache.org/confluence/display/nutch/NutchTutorial.

For example, a simple way to index https://viglet.com:

1. Nutch expects some seed URLs from where to start the crawling.

```
cd /appl/apache/nutch/
mkdir urls
echo "https://viglet.com" > urls/seed.txt
```

> **TIP** You can also limit crawling to a certain hostname etc. by setting a regular expression in `/appl/apache/nutch/runtime/local/config/regex-filter.txt`

2. Index the content with Turing AI

```
# 1.12
cd /appl/apache/nutch/
bin/crawl -i urls/ crawl-output/ 5

# 1.18
cd /appl/apache/nutch/
bin/crawl -i -s urls/ crawl-output/ 5
```

or with parameter, for instance:

```
# 1.12 (Alternative 1)
cd /appl/apache/nutch/
bin/crawl -D turing.force.config=true -D turing.site="Sample_EN" -i urls/ crawl-
output/ 5

# 1.12 (Alternative 2)
cd /appl/apache/nutch/
bin/crawl -D solr.server.url="http://localhost:2700/Sample_EN" -i urls/ crawl-
output/ 5

# 1.18
cd /appl/apache/nutch/
bin/crawl -D turing.site="Sample_EN" -i -s urls/ crawl-output/ 5
```

Table 3. crawl Parameters

Parameter	Example	Description
-D solr.server.url	-D solr.server.url="http://localhost:2700/Sample"	Turing URL + "/" + Turing Semantic Navigation Site.
-D turing.force.config	-D turing.force.config=true	Use turing.url and turing.site instead of solr.sever.url
-D turing.url	-D turing.url="localhost:2700"	Defines the fully qualified URL of Turing AI into which data should be indexed.
-D turing.site	-D turing.url="Sample"	Turing Semantic Navigation Site Name.
-D turing.auth	-D turing.auth=false	Whether to enable HTTP basic authentication for communicating with Turing AI. Use the `username` and `password` properties to configure your credentials.

Parameter	Example	Description
-D turing.username	-D turing.username="admin"	The username of Turing AI server.
-D turing.password	-D turing.password="admin"	The password of Turing AI server.

1.3.2. Nutch Provider for WEM

Web Experience Management, version 16.2 includes an example of a Page Searchable Provider using Apache Nutch, the installation and configuration is described at http://webapp.opentext.com/piroot/wcmgt/v160200/wcmgt-aci/en/html/jsframe.htm?nutch-provider-config

You can use the same Nutch Provider for InfoFusion (`com.vignette.as.server.pluggable.service.pagesearch.nutch.NutchProvider`), but using the Nutch with Turing Plugin. In Nutch Provider Configuration at WEM Configuration Console, change the variables below:

- SOLR_URL: Fill with Turing URL, for example, http://localhost:2700, instead of Solr URL;
- NUTCH_CONFIGURATION: In the XML file, put the name Turing Semantic Navigation Site in the `core` attribute, for example:

```
<?xml version="1.0" encoding="UTF-8"?>
<nutch-config
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xmlns="http://www.vignette.com/xmlschemas/nutch-config"
        xsi:schemaLocation="http://www.vignette.com/xmlschemas/nutch-config nutch-
config.xsd">
    <default crawlId="WEM_default" core="Sample"/>
    <configuration crawlId="WEM_en" core="Sample_EN">
        <locale name="en"/>
        <locale name="en_US"/>
    </configuration>
    <configuration crawlId="WEM_es" core="Sample_ES">
        <locale name="es"/>
    </configuration>
    <configuration crawlId="WEM_de" core="Sample_DE">
        <locale name="de"/>
    </configuration>
    <configuration crawlId="WEM_fr" core="Sample_FR">
        <locale name="fr"/>
    </configuration>
    <configuration crawlId="WEM_it" core="Sample_IT">
        <locale name="it"/>
    </configuration>
</nutch-config>
```

Chapter 2. Database

Command line that uses the same concept as sqoop (https://sqoop.apache.org/), to create complex queries and map attributes to index based on the result.

2.1. Installation

Go to https://viglet.com/turing/download/ and click on "Integration > Database Connector" link to download it.

Copy the turing-jdbc.jar file to /appl/viglet/turing/jdbc

```
mkdir -p /appl/viglet/turing/jdbc
cp turing-jdbc.jar.jar /appl/viglet/turing/jdbc
```

2.2. Run

To run Turing JDBC Connector executable JAR file, just execute the following line:

```
$ java -jar /appl/viglet/turing/jdbc/turing-jdbc.jar <PARAMETERS>
```

2.2.1. Parameters

Table 4. Turing JDBC parameters

Parameter	Required	Default Value	Description
--connect, -c	yes		Specify JDBC connect string
--driver, -d	yes		Manually specify JDBC driver class to use
--query, -q	yes		Import the results of statement
--site	yes		Specify the Semantic Navigation Site
--chunk, -z	no	100	Number of items to be sent to the queue
--class-name	no		Customized Class to modified rows
--deindex-before -importing	no	false	Deindex before importing
--encoding	no	UTF-8	Encoding Source
--file-content-field	no		Field that shows Content of File
--file-path-field	no		Field with File Path

Parameter	Required	Default Value	Description
--file-size-field	no		Field that shows Size of File in bytes
--help	no		Print usage instructions
--include-type-in-id, -i	no	false	Include Content Type name in Id
--max-content-size	no	5	Maximum size that content can be indexed (megabytes)
--multi-valued-field	no		Multi Valued Fields
--password, -p	no		Set authentication password
--remove-html-tags -field	no		Remove HTML Tags into content of field
--server, -s	no	http://localhost:2700	Viglet Turing Server
--show-output, -o	no	false	Show Output
--type, -t	no	CONTENT_TYPE	Set Content Type name
--username, -u	no		Set authentication username

2.2.2. Example

```
java -jar ./turing-jdbc.jar --deindex-before-importing true \
--include-type-in-id true -z 1 \
--file-path-field filePath --file-content-field text \
--file-size-field fileSize -t Document \
--multi-valued-separator ";" --multi-valued-field field1,field2 \
--class-name com.viglet.turing.tool.ext.TurJDBCCustomSample \
-d com.mysql.jdbc.Driver -c jdbc:mysql://localhost/sampleDB  \
-q "select * from sampleTable" -u sampleUser -p samplePassword
```

Chapter 3. File System

Command line to index files, extracting text from files such as Word, Excel, PDF, including images, through OCR.

3.1. Installation

Go to https://viglet.com/turing/download/ and click on "Integration > FileSystem Connector" link to download it.

Copy the turing-filesystem.jar file to /appl/viglet/turing/fs

```
mkdir -p /appl/viglet/turing/fs
cp turing-filesystem.jar /appl/viglet/turing/fs
```

3.2. Run

To run Turing FileSystem Connector executable JAR file, just execute the following line:

```
$ java -jar /appl/viglet/turing/fs/turing-filesystem.jar <PARAMETERS>
```

3.2.1. Example

```
$ java -jar build/libs/turing-filesystem.jar --server http://localhost:2700 --nlp
b2b4a1ff-3ea3-4cec-aa95-f54d0f5f3ff8 --source-dir /appl/myfiles --output-dir
/appl/results
```

Chapter 4. OpenText WEM Listener

OpenText WEM Listener to publish content to Viglet Turing

4.1. Installation

4.1.1. Download

Go to https://viglet.com/turing/download/ and click on "Integration > WEM Listener" link to download it.

Extract the turing-wem.zip file to /appl/viglet/turing/wem

```
mkdir -p /appl/viglet/turing/wem
unzip turing-wem.zip -d /appl/viglet/turing/wem
```

4.1.2. Classpath

1. Copy the turing-wem-all.jar to WEM and CDS Library directory, for example:

   ```
   cp /appl/viglet/turing/wem/turing-wem-all.jar
   /appl/ot/WEM/Content/<VERSION>/lib/
   ```

2. Edit the `cda.classpath` file of Management and Delivery Stages, for examples:

   ```
   /appl/otwork/WEM/inst-vgninst/cfgagent/vcm-vgninst/cdsvcs/stage-mgmt/cds-
   mgmt/cda-mgmt/conf/cda.classpath
   /appl/otwork/WEM/inst-vgninst/cfgagent/vcm-vgninst/cdsvcs/stage-Live/cds-
   Live/cda-Live/conf/cda.classpath
   ```

3. These cda.classpath files contain the following lines:

```
CLASSPATH.6=\#INSTALL_DIR\#/lib/jaxws
CLASSPATH.5=\#INSTALL_DIR\#/lib
CLASSPATH.4=\#INSTALL_DIR\#/lib/appsvcsda/jsp-api.jar
CLASSPATH.3=\#INSTALL_DIR\#/lib/appsvcsda/vgn-appsvcs-dadataobject.jar
CLASSPATH.2=\#INSTALL_DIR\#/lib/jax-qname.jar
CLASSPATH.1=\#INSTALL_DIR\#/jdbc
```

4. Add the following line in each cda.classpath

```
CLASSPATH.7=\#INSTALL_DIR\#/lib/turing-wem-all.jar
```

4.1.3. WEM Deploy

Add the turing-wem-all.jar into WEM using configp:

```
$ ./configp
=============================================================

Configuration Program Main Menu

----------------------------------------
   1.  Connect to WEM Server
   2.  Create a Disconnected Configuration Agent
   3.  Remove a Disconnected Configuration Agent
   4.  Repair Management Server

   q.  Quit

   > 1
=============================================================
Connect to WEM Server: WEM Server Connection Information


WEM Server host: wemserver
WEM Server port: 27110
WEM Server administrator: vgnadmin
WEM Server administrative password:

****************************************
 You have entered the following:

  WEM Server host = wemserver
  WEM Server port = 27110
  WEM Server administrator = vgnadmin
  WEM Server administrative password = ********


Is this correct ( (y)es, (n)o, (b)ack, (c)ancel )?[y]:
Connecting...
Connected to t3://wemserver:27110
=============================================================

Managing Configuration Services
----------------------------------------
   1.  Manage a Product Instance
   2.  Create a Configuration Agent
```

```
   3.  Remove a Configuration Agent
   4.  Register a Configuration Agent
   5.  Manage Applications
   6.  List Configuration Settings

   b.  Back
   q.  Quit

   > 5
===========================================================
Manage Applications: Manage Application


  To register or unregister Extension Modules, select
  Register Product Extensions. To modify an existing
  deployed application, select Update Runtime Services.

Select type of application update
---------------------------------
   1.  Register Product Extensions
   2.  Update Runtime Services

   b.  Back
   c.  Cancel

   > 1

****************************************
 You have entered the following:

  Select type of application update = Register Product Extensions


Is this correct ( (y)es, (n)o, (b)ack, (c)ancel, (u)ndo )?[y]:
===========================================================
Manage Applications: Deployment Types


  You can choose to deploy an extension which exists
  within the VCM ear container or a standalone application
  outside of the VCM ear container.

Do you want to deploy an extension or standalone application?
-------------------------------------------------
   1.  Extension
   2.  Standalone Application
```

```
   b.  Back
   c.  Cancel

   > 1

****************************************
 You have entered the following:

  Do you want to deploy an extension or standalone application? = Extension


Is this correct ( (y)es, (n)o, (b)ack, (c)ancel, (u)ndo )?[y]:
============================================================
Manage Applications: Deployment Actions


Register Extension Type
-----------------------
   1.  JAR Extension Module
   2.  WAR Extension Module
   3.  Multiple Extension Modules - can include both JAR and WAR files

   b.  Back
   c.  Cancel

   > 1
Deployment Action
-----------------
   1.  Deploy Extension
   2.  Undeploy Extension

   b.  Back
   c.  Cancel

   > 1

****************************************
 You have entered the following:

  Register Extension Type = jarext (JAR Extension Module)
  Deployment Action = Deploy Extension


Is this correct ( (y)es, (n)o, (b)ack, (c)ancel, (u)ndo )?[y]:
============================================================
```

```
Manage Applications: Extension JAR Path


  Enter the path to the archive file containing the
  extension. This file is registered with the repository
  and deployed to the application server.

  Important!! Deployment of an extension could take
  up to 15 mins.

JAR Path (example: C:\vign_extn.jar): /appl/viglet/turing/wem/turing-wem-all.jar

*****************************************
 You have entered the following:

  JAR Path (example: C:\vign_extn.jar) = /appl/viglet/turing/wem/turing-wem-all.jar


Is this correct ( (y)es, (n)o, (b)ack, (c)ancel, (u)ndo )?[y]: y
===============================================================
Manage Applications: Confirm Configuration


  Are you ready to perform this action?



Continue? ( (y)es, (n)o, (b)ack, (c)ancel )? [y]: y

Confirm Configuration:

  All the information has been collected. Would you
  like to commit the configuration? (y/n) [y]: y

Step 1 of 3: Validating Input ...
Step 2 of 3: Check Configuration Status ...
Step 3 of 3: Updating Application ...

Success:

The configuration wizard completed successfully.
```

4.1.4. Resource

Access the Configuration Console (http://wem_host:wem_port/configconsole) and add the VigletTuring Generic Resource in each Delivery Stage that will index to Turing Semantic Navigation.

For example:

1. Click on right-button on `Configuration Console > Content > Delivery Services > Content Delivery Stage - Live > Resources`, select Add Resource
2. In Resource Type, select "Generic Resource" and click Next
3. In Resource Name, type: `VigletTuring` and click Next
4. In Generic Resource Type, select "Other(Any stage-specific resource subtype information)" and click Next
5. In Resource Subtype, type: `Properties` and click Next
6. In Resource Information > Non-Encrypted Data type: `fill later` and Encrypted Data leaves blank and click Next
7. In Confirm Configuration click Finish.
8. Edit "Configuration Console > Content > Delivery Services > Content Delivery Stage - SebraeNA > Resources > Resource Type - Generic > Resource - VigletTuring > Generic Resource > DATA" and replace "fill later" for:

```
turing.url=http://localhost:2700
turing.mappingsxml=/appl/viglet/turing/wem/conf/CTD-Turing-Mappings.xml
turing.login=admin
turing.password=admin

cda.default.urlprefix=http://localhost
cda.default.contextname=sites
sn.default.site=Sample
sn.it_IT.site=SampleIT

sites.association.priority=SampleSite
```

Where

Table 5. VigletTuring Generic Resource Properties

Parameter	Required	Description
turing.url	yes	Turing URL.
turing.mappingsxml	yes	XML File.
turing.login	yes	Turing Login.
turing.password	yes	Turing Password.
cda.default.urlprefix	no	Prefix will be used to create URL of content in Search.
cda.default.contextname	no	Context Name of DPS.
sn.default.site	yes	Name of site on Turing Semantic Navigation, that will be used to index the WEM Content.
sn.<locale>.site	no	If the content has locale attribute, you can specify a different Semantic Navigation Site that will be indexed.
sites.association.priority	no	If the content is associated with more than one site, you can define which site will be chosen to avoid conflict.

NOTE Repeat this procedure in other Management and Delivery Stages that will use Turing Semantic Navigation

IMPORTANT The Listener uses URL Scheme from Site to generate Content URL.

4.1.5. Events

Access the Configuration Console (http://wem_host:wem_port/configconsole) and add the EventListener in each Delivery Stage that will index to Turing Semantic Navigation.

Configure the Deployment.ManagedObjectCreate, Update and Delete listeners.

1. Register the required listeners to the events as specified below:
   - `Configuration Console > Content > Delivery Services > Content Delivery Stage - Live > Content Delivery Services - Live > Application Services > Events > Deployment.ManagedObjectCreate`

     ```
     com.viglet.turing.wem.listener.DeploymentEventListener
     ```

   - `Configuration Console > Content > Delivery Services > Content Delivery Stage - Live > Content Delivery Services - Live > Application Services > Events > Deployment.ManagedObjectDelete`

     ```
     com.viglet.turing.wem.listener.DeploymentEventListener
     ```

   - `Configuration Console > Content > Delivery Services > Content Delivery Stage - Live > Content Delivery Services - Live > Application Services > Events > Deployment.ManagedObjectUpdate`

     ```
     com.viglet.turing.wem.listener.DeploymentEventListener
     ```

   NOTE Be sure to copy any existing listeners from the current run value and append the new listener to the end of the list during registration. If needed, see section 6 of the Management Console Extensibility SDK guide for more information on registering event listeners.

2. Commit the configuration changes and restart the DA

4.1.6. Command Line

Copy `/appl/viglet/turing/wem/command-line/<WEM_VERSION>/turing-wem` to

<WEM_DIR>/bin, it works a lot like vgncontentindex command line.

4.2. Configuration

4.2.1. Mapping

Create a /appl/viglet/turing/wem/conf/CTD-Turing-Mappings.xml file with the following lines:

```
<?xml version="1.0" encoding="UTF-8"?>
<mappingDefinitions>
    <common-index-attrs>
        <srcAttr className="com.viglet.turing.wem.ext.TurCTDName" mandatory="true">
            <tag>type</tag>
        </srcAttr>
        <srcAttr className="com.viglet.turing.wem.ext.TurWEMPublicationDate"
mandatory="true">
            <tag>publication_date</tag>
        </srcAttr>
        <srcAttr className="com.viglet.turing.wem.ext.TurWEMModificationDate"
mandatory="true">
            <tag>modification_date</tag>
        </srcAttr>
        <srcAttr className="com.viglet.turing.wem.ext.TurSiteName" mandatory=
"true">
            <tag>site</tag>
        </srcAttr>
        <srcAttr className="com.viglet.turing.wem.ext.HTML2Text">
            <tag>text</tag>
        </srcAttr>
        <srcAttr className="com.viglet.turing.wem.ext.HTML2Text">
            <tag>abstract</tag>
        </srcAttr>
        <srcAttr className="com.viglet.turing.wem.ext.DPSUrl" mandatory="true">
            <tag>url</tag>
        </srcAttr>
    </common-index-attrs>
    <mappingDefinition contentType="INNOVATE_PRESS_RELEASE">
        <index-attrs>
            <srcAttr xmlName="title">
                <tag>title</tag>
            </srcAttr>
            <srcAttr xmlName="teaser">
                <tag>abstract</tag>
            </srcAttr>
            <srcAttr xmlName="body">
                <tag>text</tag>
            </srcAttr>
            <srcAttr textValue="foo bar">
                <tag>text</tag>
            </srcAttr>
        </index-attrs>
    </mappingDefinition>
</mappingDefinitions>
```

NOTE

There should be a srcAttr element for each content type field to be indexed by Turing AI. The xmlName attribute should contain the XML Name of the relevant field.

4.3. CTD-Turing-Mappings.xml Elements

The following sections describe the delements defined in the CTD-Turing-Mappings.xml file under the root element `<mappingDefinitions>`:

4.3.1. common-index-attrs

Table 6. srcAttr (common-index-attrs) Element Definition

Element	Description
srcAttr	List of tags (turing fields) that can be used by CTDs in mappingDefinition.

Table 7. srcAttr (common-index-attrs) Attributes

Attribute	Required/ Optional	Default Value	Description
mandatory	Optional	"false"	If "true", it means the tag will always be inserted in all CTDS.
classname	Required	-	Custom class to process the field value. Implicitly define this custom class to process the field value className in mappingDefinition srcAttr when the same tag is used.

4.3.2. mappingDefinition

Table 8. mappingDefinition Element Definition

Element	Description
mappingDefinition	CTD Mapping.

Table 9. mappingDefinition Attribute

Attribute	Required/ Optional	Default Value	Description
contentType	Required	-	Content Type XML Name.

Table 10. index-attrs Element Definition

Element	Description
index-attrs	List of Content Type Field

Table 11. srcAttr (mappingDefinition) Element Definitiion

Element	Description
srcAttr	Content Type Field to be indexed by Turing AI.

Table 12. srcAttr (mappingDefinition) Attributes

Attribute	Required/ Optional	Default Value	Description
xmlName	Required (if className or textValue is missing)	-	Content Type Field XML Name.
relation	Optional	-	Content Type Relation XML Name.
uniqueValues	Optional	"false"	A List return unique values.
valueType	Optional	-	If "html" then convert HTML to Text.

Attribute	Required/ Optional	Default Value	Description
classname	Required (if xmlName or textValue is missing)	-	Custom class to process the field value.
textValue	Required (if xmlName or classname is missing)	-	returns a text for the tag (Turing field)

Table 13. tag Element Definition

Element	Element Description
tag	Turing AI Semantic Navigation Field

Chapter 5. Wordpress

Wordpress plugin that allows you to index posts.

5.1. Installation

1. Upload the `turing4wp` folder to the `/wp-content/plugins/` directory
2. Activate the plugin through the 'Plugins' menu in WordPress
3. Configure the plugin with the hostname, port, and URI path to your Solr installation.
4. Load all your posts and/or pages via the "Load All Posts" button in the settings page.