# BUS EMPRESARIAL

# Configuración Pipelines para compilacion servicios con Ant.

## **Resumen**

Este documento tiene la información correspondiente a la configuración de Tareas, variables y pipelines para compilar servicios java con Ant.

**Tabla de Contenido**

[Configuracion de ANT en servidor de compilación para aplicaciones JAVA 1](#_Toc26272838)

[**Resumen** 1](#_Toc26272839)

[**1.** **Pre-requisitos.** 1](#_Toc26272840)

[**2.** **Instalación de Apache Ant.** 2](#_Toc26272841)

[**3.** **Registro variables del sistema.** 3](#_Toc26272842)

[**4.** **Validar Disponibilidad.** 5](#_Toc26272843)

## **Pre-requisitos.**

* Cola de agentes configurada con agentes para compilación que tengan Ant y Suite Oracle instalados. Para ver detalles de esa configuración, revisar la guía: **Configurar ANT y Suite Oracle en Servidores de compilacion.docx**.

## **Grupos de tareas.**

Con el fin de configurar los pipelines para compilaciones y despliegues, se hace uso de la utilidad del grupo de tareas aprovechando las características de los servicios de BUS EMPRESARIAL ya que los procesos para compilar y desplegar son comunes. Se han configurado los siguientes grupos:

**Compilación**

* Compilar Servicio con ConfigurationFile
* Compilar Servicio sin ConfigurationFile

**Despliegue y rollback del despliegue**

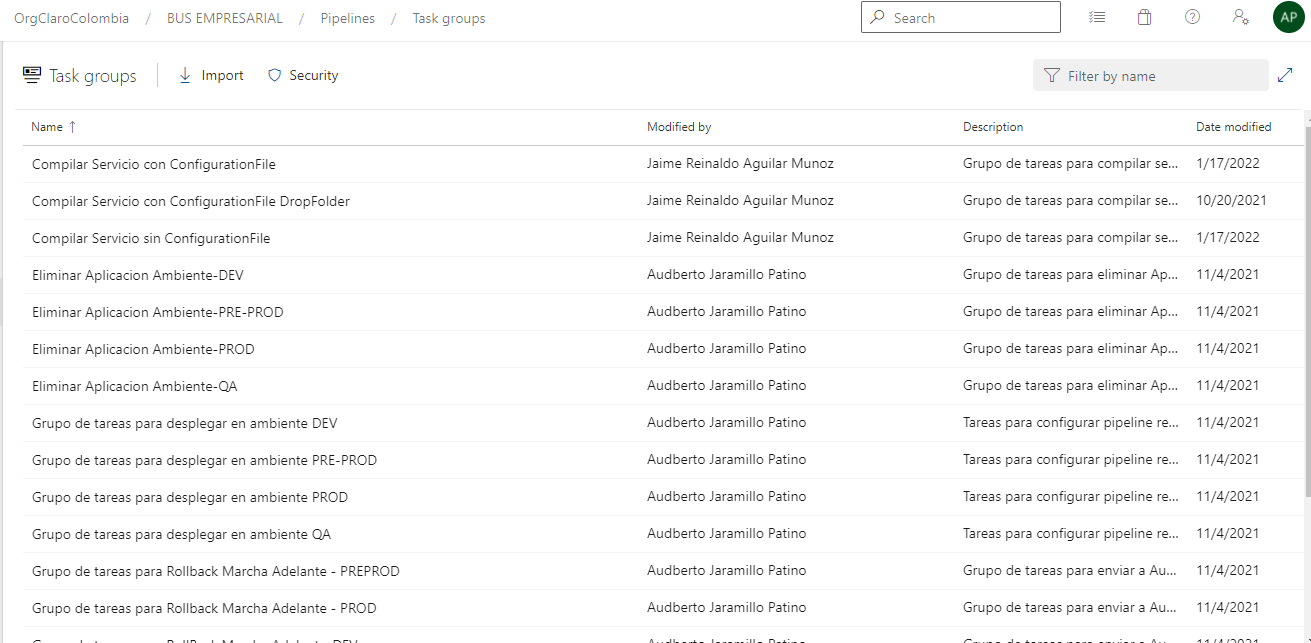
* Grupos de tareas para desplegar en ambientes DEV, QA, PREPROD y PROD
* Grupos de tareas para rollback de despliegue en ambientes DEV, QA, PREPROD y PROD

**Eliminación de aplicación y rollback de eliminación**  
-Eliminar Aplicacion Ambientes DEV, QA, PRE-PROD y PROD

* RollBack Eliminar Proyecto Ambientes DEV, QA, PREPROD y PROD

Actualizar imagen de grupos de tareas

**Imagen grupos de tareas**



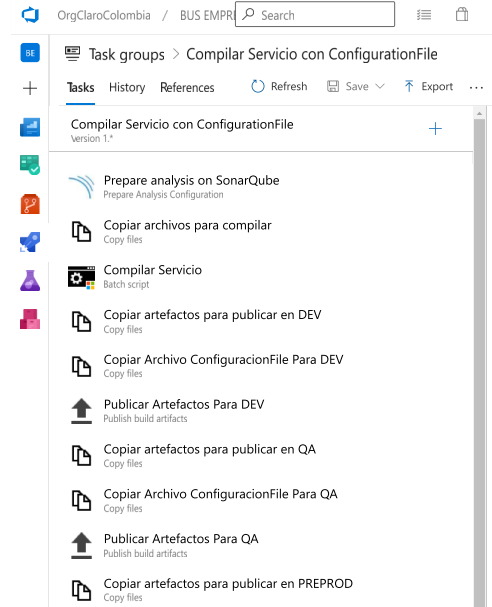
## **Grupo de tareas para compilar.**

En el grupo de tareas del Team Project se encuentran configurados los siguientes grupos:

1. Compilar Servicio con ConfigurationFile
2. Compilar Servicio sin ConfigurationFile

La diferencia entre estos dos grupos de tareas es que para el primer caso tiene configurada una tarea (por cada ambiente) que se encarga de enviar el archivo ConfigurationFile.  
Se configura un pipeline de compilación, por cada repositorio de código fuente y tiene asociado el grupo de tareas según requiera o no del archivo ConfigurationFile durante el proceso de despliegue. Las tareas que se documentan a continuación corresponden a la configuración que incluye los archivos "ConfigurationFile".

## **Imagen configuración del grupo de tareas:**



## **Tareas configuradas en el grupo Compilar Servicio con ConfigurationFile**

## **Preparar Análisis en SonarQube**: Se utiliza para configurar el acceso al servidor de SonarQube y definición de variables a utilizar:

**SonarQube Server**: Asignar la conexión a utilizar. Actualmente está configurado como Sonar-Claro.La conexión se configura mediante la opción *Service Connection* del proyecto. Agregar url de sonar claro y token para clarocolombia y alminspiraclaro

**Choose the way to run the analysis**: Seleccionar la opción *Use standalone scanner*

**Mode**: Seleccionar *Manually provide configuration*

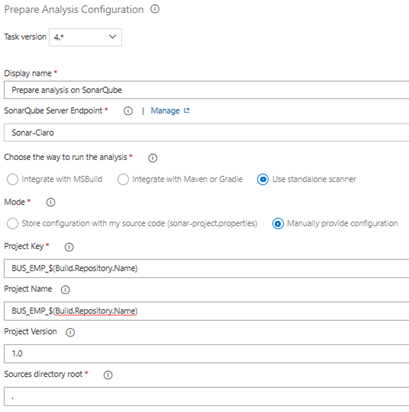
**Project key**: BUS\_EMP\_$(Build.Repository.Name). Compuesto por un prefijo utilizado para identificar el nombre del proyecto mas la variable que contiene el nombre del repositorio del código fuente.

**Project name**: BUS\_EMP\_$(Build.Repository.Name). Se utiliza la misma configuración del Project key.

**Custom Condition**, se configura lo siguiente para que el análisis se realice únicamente cuando se envía la compilación desde la rama Develop.  
and(succeeded(), eq(variables['Build.SourceBranch'], 'refs/heads/Develop'))

**Nota:** La rama Develop se debe configurar como rama por defecto.

**Imagen tarea configurada**



## **Copiar archivos para compilar**

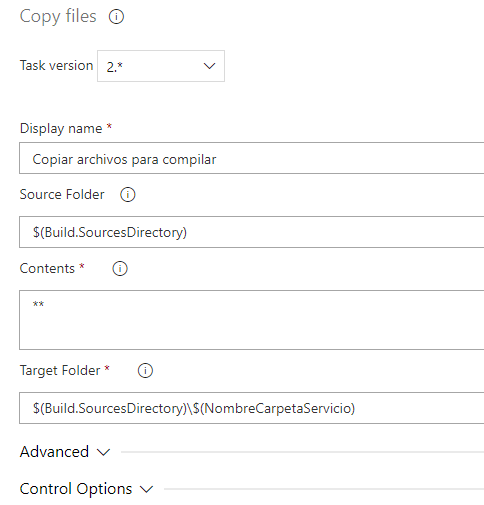
Tarea utilizada para copiar los artefactos del repositorio fuente clonado automáticamente por el pipeline, a una ruta temporal, con el fin de correr el proceso de compilación.

**Source Folder**: $(Build.SourcesDirectory): Variable predefinida; es la ruta donde el agente de compilación clona el repositorio de código fuente

**Contents**: Filtro para indicar cuales artefactos se copian. En esta opción se configura \*\* para indicar que se copia todo el contenido.

**Target Folder**: $(Build.SourcesDirectory)\$(NombreCarpetaServicio), Es la ruta donde se copian los artefactos. NombreCarpetaServicio es una variable global en la que se define el nombre del archivo .jpr que está en el repositorio del código fuente

**Imagen tarea configurada**

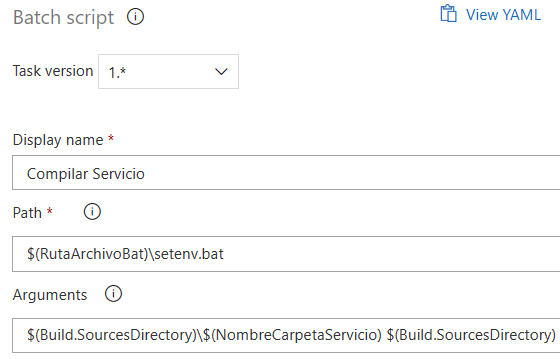


## **Compilar Servicio**

Tarea de línea de comando de Windows que utiliza el archivo setenv.bat para hacer la compilación del servicio

|  |  |
| --- | --- |
| **Nombre** | **Descripcion** |
| Path: $(RutaArchivoBat)\setenv.bat | Variable que contiene la ruta (D:\Oracle\Middleware\Oracle\_Home\osb\tools\configjar) donde se encuentra el archivo setenv.bat. Este archivo tiene la configuración necesaria para compilar con Ant |
| Arguments: Build.SourcesDirectory | Variable pre-definida que apunta al directorio de trabajo donde se encuentra el codigo fuente. La carpeta se encuentra alojada en el directorio del trabajo del agente que hace la compilación: Ej: D:\Serv28\_1\_work\xx\s |
| Arguments: NombreCarpetaServicio | Variable del pipeline que contiene el nombre del servicio. En cada pipeline se le configura a la variable el nombre del servicio según lo configurado en el archivo fuente .jpr |

**Imagen tarea configurada**



## **Copiar artefactos para publicar**

Tarea utilizada para copiar el empaquetado .jar, creado en el proceso de compilación, a la carpeta de los artefactos compilados $(build.artifactstagingdirectory) para posterior publicación en el DropFolder. Este proceso de copiado se realiza en las tareas:  
Copiar artefactos para publicar en DEV, QA, PREPROD Y PROD

|  |  |
| --- | --- |
| **Nombre** | **Descripcion** |
| **Source Folder** | Se deja vacío. Por defecto es la ruta donde el agente de compilación clona el repositorio de código fuente; equivale a la variable $(Build.SourcesDirectory) |
| **Contents** | Filtro para indicar cuales artefactos se copian. En esta opción se configura lo siguiente: \*\*\\\*\*\\\*$(NombreCarpetaServicio).jar\* \*\*\\\*\\\*.war \*\*\\\*\*\\.class |
| **Target Folder** | $(build.artifactstagingdirectory) esta es una variable predefinida en Azure DevOps y contiene la ruta donde se copian los artefactos compilados |

**Imagen tarea configurada**

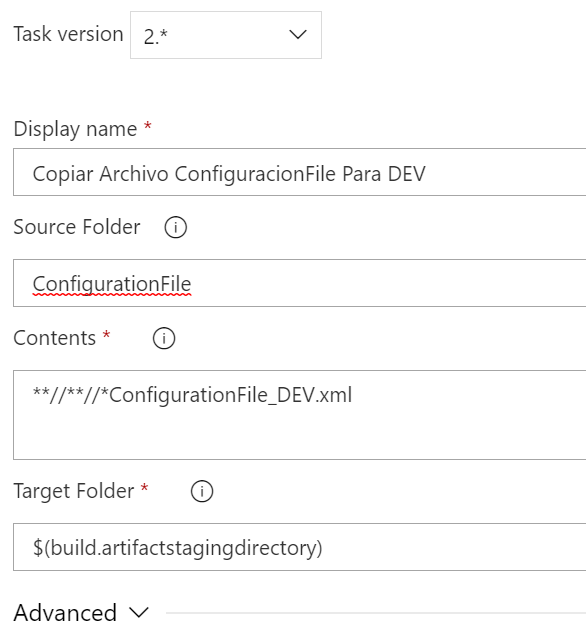


## **Copiar archivo ConfigurationFile**

Tarea utilizada para copiar el archivo ConfigurationFile\_XXX.xml, de la carpeta ConfigurationFile, a la carpeta de los artefactos compilados $(build.artifactstagingdirectory). En una tarea posterior se deja disponible en el DropFolder. Este proceso de copiado se realiza en las tareas: Copiar Archivo ConfiguracionFile Para DEV, QA, PREPROD Y PROD, PREPROD Y PROD

|  |  |
| --- | --- |
| **Nombre** | **Descripcion** |
| **Source Folder** | Se especifíca la carpeta donde se encuentra el archivo ConfigurationFile |
| **Contents** | Filtro para indicar el archivo .xml a copiar, de acuerdo al ambiente donde se vaya a realizar el despliegue. Por ejemplo para el ambiente de desarrollo se configura:  \*\*//\*\*//\*ConfigurationFile\_DEV.xml |
| **Target Folder** | $(build.artifactstagingdirectory) esta es una variable predefinida en Azure DevOps y contiene la ruta donde se copian los artefactos compilados. |

**Imagen tarea configurada**



## **Publicar artefactos**

Tarea utilizada para copiar al DropFolder los artefactos necesarios para hacer el despliegue. Se toman de la ruta $(build.artifactstagingdirectory). Así, al final de cada proceso de compilación, los artefactos quedarán disponibles para descargarlos desde el pipeline que lo(s) creó o bien a tráves de la url que se crea para hacer la descarga.

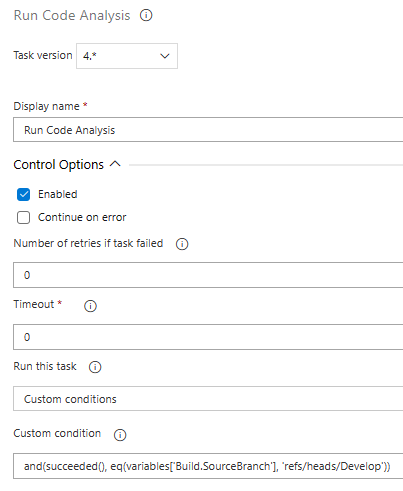
|  |  |
| --- | --- |
| **Nombre** | **Descripcion** |
| **Path to publish** | Ruta (build.artifactstagingdirectory) desde la cual se toman los artefactos. La carpeta se encuentra alojada en el directorio del trabajo del agente que hizo la compilación: Ej: D:\Serv28\_1\_work\xx\a |
| **Artifact Name** | Se especifica el nombre con el cual va a quedar disponible el o los artefactos que se utilizarán en el despliegue. En cada pipeline a la variable $(NombreCarpetaServicio) se le asigna el nombre del servicio según lo configurado en el archivo fuente .jpr, más el ambiente en que va a ser publicado. Ejemplo: $(NombreCarpetaServicio)\_DEV |
| **Artifact publish location** | Corresponde a la opcion para indicar que los artefactos se almacenan en Azure Pipelines (tambien se le conoce como DropFolder). Se deja configurado Azure Pipelines |



## **Run code análisis SonarQube**

Esta tarea se encarga de realizar el análisis del código fuente. Previamente se debe haber configurado la tarea que se encarga de establecer la conexión entre Azure DevOps y el Servidor de SonarQube. Detalles en el numeral 3.1.  
En la sección **Control Option** se habilita la característica **Custom condition** para condicionar la ejecución de esta tarea cuando se envíe la compilación desde la rama Develop   
and(succeeded(), eq(variables['Build.SourceBranch'], 'refs/heads/Develop'))

**Imagen tarea configurada**



## **Publicar resultados análisis SonarQube**

Tarea que se configura para publicar los resultados del proceso de análisis del código fuente. Para la configuración se dejan los valores presentados por defecto. En la sección **Control Option** se habilita la característica **Custom condition** para condicionar la ejecución de esta tarea cuando se envíe la compilación desde la rama Develop   
and(succeeded(), eq(variables['Build.SourceBranch'], 'refs/heads/Develop'))

**Imagen tarea configurada**

