HHS HR BizFlow CapHR Interface Deployment Guide

Document Control Information

Document Information

|  |  |
| --- | --- |
| Document Identification | HHS HR BizFlow CapHR Interface Deployment Guide.docx |
| Document Name | Deployment |
| Project Name | HHS BizFlow HR System |
| Client | US Health & Human Services |
| Document Author | Prabhjyot Virdi |
| Document Version | 1.0.0 |
| Document Status |  |
| Date Released | TBD |
| Business Specifications Requirement Document ID |  |
| Functional Specification ID |  |

Document Edit History

| Version | Date | Additions/Modifications | Prepared/Revised by |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Document Review/Approval History

| Date | Name | Organization/Title | Comments |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Distribution of Final Document

The following people are designated recipients of the final version of this document:

| Name | Organization/Title |
| --- | --- |
| <Name> | <Organization/Title> |
|  |  |

Table of contents

[1 Introduction 4](#_Toc519855799)

[2 Required system version information 5](#_Toc519855800)

[3 Development directory structure 6](#_Toc519855801)

[3.1 Root Directory Contents 6](#_Toc519855802)

[3.2 Source Code Directory Contents 6](#_Toc519855803)

[3.3 Resources Directory Contents 6](#_Toc519855804)

[3.4 Library Directory Contents 6](#_Toc519855805)

[3.5 Logs Directory Contents 6](#_Toc519855806)

[4 Methods of deployment file delivery 7](#_Toc519855807)

[4.1 Development file location in GitHub 7](#_Toc519855808)

[5 Deployment Steps 8](#_Toc519855809)

[5.1 Database Deployment 8](#_Toc519855810)

[5.1.1 Database Object Pre-requisites 8](#_Toc519855811)

[5.1.2 Create database package 8](#_Toc519855812)

[5.1.3 Grant Package Permissions 8](#_Toc519855813)

[5.2 Configuration 9](#_Toc519855814)

[5.2.1 application.properties configuration 9](#_Toc519855815)

[5.2.1.1 Target Database 9](#_Toc519855816)

[5.2.1.2 Email Notifications 9](#_Toc519855817)

[5.2.1.3 Log Properties 10](#_Toc519855818)

[5.2.1.4 Control CapHR Interface 10](#_Toc519855819)

[5.2.1.5 PL/SQL Stored Function Properties 10](#_Toc519855820)

[5.3 Build the Source Code 10](#_Toc519855821)

[5.4 Deployment on Sever 11](#_Toc519855822)

[5.5 Setting up the Cron Job 11](#_Toc519855823)

# Introduction

This document describes how to deploy the CapHR interface to the HHS BizFlow HR system. It is assumed that there are 3 environments, DEV, QA, and PROD for HHS BizFlow. There are 3 environments, DEV, QA and PROD, for the CapHR system. The CapHR DEV, QA and PROD environments will be connected with the HHS BizFlow DEV, QA and PROD environments respectively. The instructions for deployment for each environment will be the same except for configuration. Configuration should be done specifically for each target environment.

# Required system version information

1. BizFlow: BizFlow Server version 12.4
2. Database: Oracle 12c
3. Java: JDK 1.7
   1. Confirm that the JAVA\_HOME system variable points to the correct JDK 1.7 path
4. Apache ANT version 1.9.1
   1. Confirm that the ANT\_HOME system variable points to the ANT path

# Development directory structure

The following directory capture will be delivered as part of the release of the system.

/caphr/

/caphr/src/main/java/

/caphr/src/main/resources/

/caphr/lib/

## Root Directory Contents

The /caphr/ directory contains the build script, build.xml, required for building the source code with Apache Ant. The dependency jars will be copied into the generated executable jar file when the build script is run. A new executable jar for the interface application is created and will be stored in the /caphr/dist/ directory with the dependency jars. The root directory also stores a caphr.properties file which contains the JDK path.

## Source Code Directory Contents

The /caphr/src/main/java/ directory contains the java source code for running the interface batch job.

## Resources Directory Contents

The /caphr/src/main/resources/ directory contains the application.properties file which has configurable properties that need to be updated based on the target execution environment. The resources directory also has the shell scripts and manifest file used to execute the job.

## Library Directory Contents

The /caphr/lib/ directory contains the dependency jar files needed to run the application. These jar files will be copied onto the generated executable jar file when the code is compiled with Apache Ant.

## Logs Directory Contents

The <caphr\_installation\_dir>/logs/ directory will contain the WHRSC-CapHRInterface.log file generated at runtime, which contains logging information for each time the code is executed.

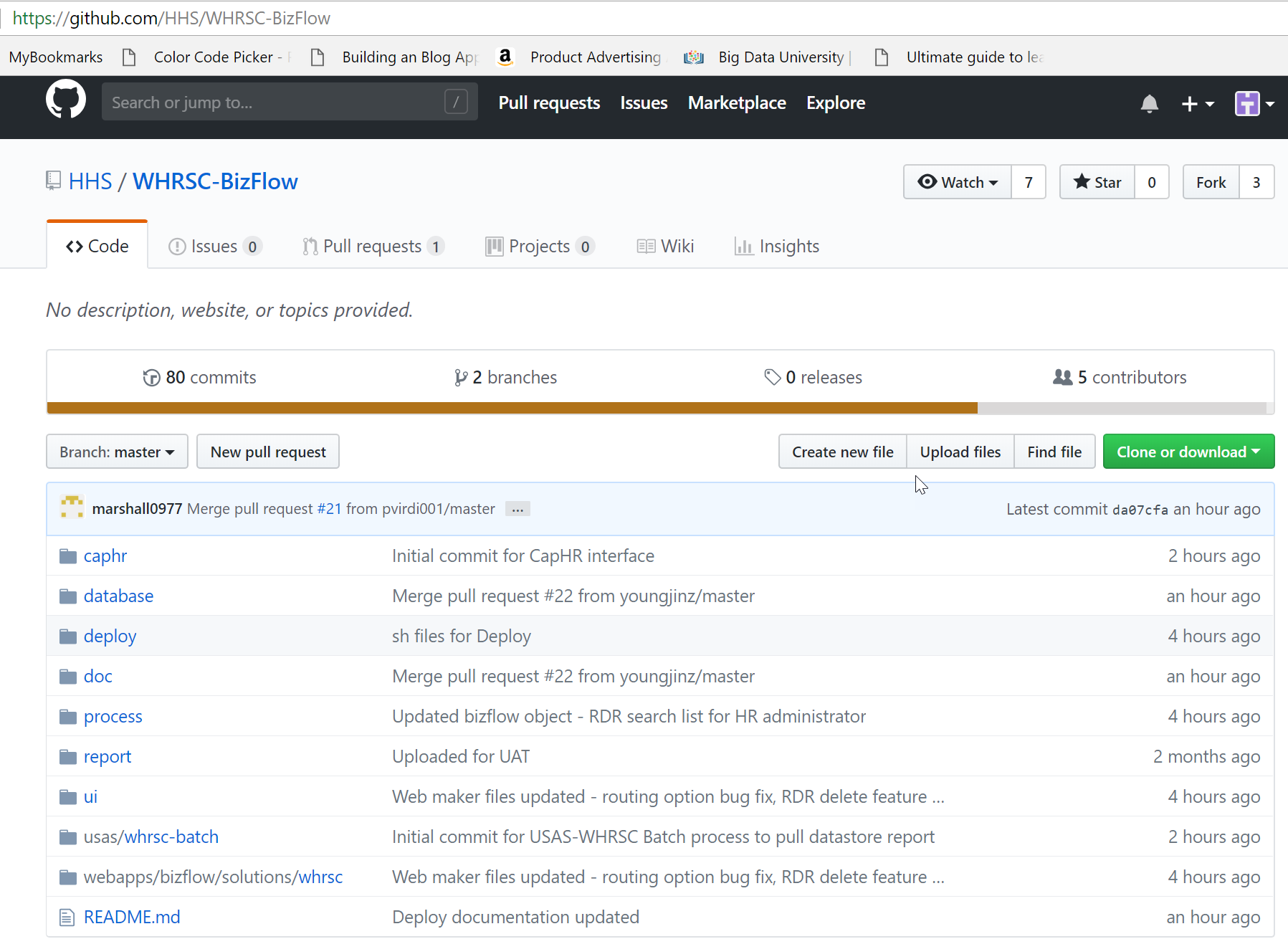
# Methods of deployment file delivery

The following methods of deployment file delivery may be considered:

* USB drive to HHS team
* Upload to a designated ftp/sftp site in HHS network by authorized Deloitte team
* Upload from HHS issued laptop by authorized Deloitte team
* Download from the WHRSC-BizFlow GitHub site (https://github.com/HHS/WHRSC-BizFlow) by HHS personnel

## Development file location in GitHub

1. Log in to GitHub and navigate to the WHRSC-BizFlow repository
2. Review the README.md for the CapHR Interface module
3. The database scripts are in the database folder under WHRSC-BizFlow Repository.



# Deployment Steps

## Database Deployment

Navigate to the database directory on GitHub. A DBA should perform the following steps using Oracle database client.

### Database Object Pre-requisites

* Confirm that the table, ERROR\_LOG, and stored procedure, SP\_ERROR\_LOG, exist in the WHRSC database schema.

If they do not exist then execute the following script using the HHS\_WHRSC\_HR user login:

WHRSC\_HR\_DB\_05\_create\_core\_table.sql

Then run the following script using the Oracle system user login:

WHRSC\_HR\_DB\_07\_grant\_permission\_core.sql

* Confirm that the tables required for the CapHR interface, exist in the WHRSC database schema.

If they do not exist then execute the following script using the HHS\_WHRSC\_HR user login:

WHRSC\_HR\_DB\_03\_create\_model\_objects.sql

Then run the following script using the Oracle system user login:

WHRSC\_HR\_DB\_04\_grant\_permission\_model.sql

* Using the Oracle system user login, execute the following script.

HHS\_HR\_DB\_CAPHR\_01\_table\_grants.sql

It will perform the following actions.

* Execute grants on the source tables for the CapHR interface

### Create database package

Using the HHS\_WHRSC\_HR user login, execute the following scripts.

WHRSC\_HR\_DB\_CapHR\_01\_create\_pkg\_spec\_caphr\_data.sql

WHRSC\_HR\_DB\_CapHR\_02\_create\_pkg\_body\_caphr\_data.sql

They will perform the following actions.

* Create database package called CAPHR\_DATA\_PKS. Make sure the package spec is compiled before the package body.

### Grant Package Permissions

Using the Oracle system user login, execute the following scripts.

WHRSC\_HR\_DB\_CapHR\_03\_pkg\_permission.sql

It will perform the following actions.

* Execute grants on the database package

If there are any errors, run the following query:

SELECT \* FROM DBA\_ERRORS

WHERE OWNER = ‘HHS\_WHRSC\_HR’;

## Configuration

The application.properties file needs to be updated depending on the target environment the interface is deployed. This file is located in the /caphr/src/main/resources/ directory.

### application.properties configuration

#### Target Database

Update the highlighted fields with the connection details for the target database. This should connect to the HHS WHRSC HR BizFlow Oracle database WHRSC schema (HHS\_WHRSC\_HR).

#Target database Configuration

whrsc.datasource.url=jdbc:oracle:thin:@host:port/service name

whrsc.datasource.username=username

whrsc.datasource.password=password

whrsc.datasource.driver-class-name=oracle.jdbc.driver.OracleDriver

whrsc.datasource.initialize=false

#### Email Notifications

The following property turns on/off the email functionality. If the property is set to false then no email notifications will be sent out. If the property is set to true then email notifications will be sent out. Make sure to fill out the rest of the email account properties if the property is set to true.

#Send Email Notifications –true or false

send.email.notification=false

If the HHS email account requires authentication, then fill out the blank fields below with the username and password. Make sure that the spring.mail.properties.mail.smtp.auth is ‘true’. If the email account does not require authentication, then leave the username and password fields blank. The spring.mail.properties.mail.smtp.auth property should be ‘false’. If the HHS email server requires a TLS connection then set the spring.mail.properties.mail.smtp.starttls.enable to ‘true’, otherwise set it to ‘false’.

#Email Notification Account

spring.mail.default-encoding=UTF-8

spring.mail.host=smtp.host

spring.mail.username=

spring.mail.password=

spring.mail.port=25

spring.mail.protocol=smtp

spring.mail.test-connection=false

spring.mail.properties.mail.smtp.auth=false

spring.mail.properties.mail.smtp.starttls.enable=false

Update the emailid.from with the email address of the user account that is sending out the email notifications. Update the emailid.to with the recipient email addresses. Separate email addresses with a comma.

#Email Notification properties

emailid.from=user email

emailid.to=recipient emails

The email subject and message configuration properties should not be updated. The job name, status, and parameters will be dynamically passed into the fields each time the batch job is executed.

email.subject=DATE\_TODAY - INTERFACE\_NAME: Batch Job 'JOB\_NAME' STATUS

email.message=The batch job 'JOB\_NAME' completed with the following status: [STATUS] and the following parameters: [PARAMETERS].

#### Log Properties

Keep the logging root at ‘INFO’ unless the interface is not running correctly. If there are any issues, then change the logging root to ‘DEBUG’.

#Logging properties

# Must be one of ("TRACE", "DEBUG", "INFO", "WARN", or "ERROR")

logging.level.root=INFO

The location of the log file will be generated at runtime. If specified as an absolute path, i.e. /logs/WHRSC-CapHRInterface.log, then this is the exact location of the log file. If specificed as a relative path, i.e. logs/WHRSC-CapHRInterface.log, it will be located under the runtime installation/deployment directory.

#output logs to a file

logging.file=logs/WHRSC-CapHRInterface.log

#### Control CapHR Interface

Running the CapHR database function can be turned on or off by changing the following property to true or false. If the property is set to true then the database function will execute. If the property is false then the function will not execute and a message will logged to the log file. The property, import.caphr.data, corresponds to the part of the interface that pulls over CapHR data from BIIS tables in the global schema and inserts it into the WHRSC CapHR’s tables. The default property is set to true and should not be changed to false unless there is a reason to turn off the interface.

#Run PL/SQL Function – True or False

import.caphr.data=true

#### PL/SQL Stored Function Properties

These properties point to a stored PL/SQL function in the WHRSC schema, HHS\_WHRSC\_HR. These properties should not be updated unless the package (catalog) or function name is changed.

plsql.caphr.catalog=CAPHR\_DATA\_PKS

plsql.caphr.schema=HHS\_WHRSC\_HR

plsql.caphr.function=FN\_IMPORT\_CAPHR\_DATA

## Build the Source Code

As a system administrator, complete the following steps to build the source code using Apache Ant:

1. Open up a command terminal.
2. Change the directory to the caphr directory, ex: ‘cd caphr’
3. Using a text editor, open the caphr.properties file and modify the JDK location in the following property

jdk.home.1.7=<full\_path\_to\_jdk\_home\_dir>

1. In the command line, run the ant build script, ex: ‘ant package.exejar’
2. Capture the generated module JAR file, configuration files, and shell script to run the module.

caphr/dist/run-jar.sh

caphr/dist/application.properties

caphr/dist/caphr-<version>.jar

## Deployment on Sever

1. Log in to the target environment server as the BizFlow service owner (or sudo).
2. Create a directory where the CapHR Interface module will be installed

For example: mkdir –p *<server\_dir>*/caphr/

1. Copy the module JAR file, configuration files and shell script
   1. From the build machine:
      1. caphr/dist/\*
   2. To the target environment:
      1. *<server\_dir>*/caphr/
2. Make sure the script has the correct JAVA\_HOME environment variable set. Open and edit run-jar.sh script.

JAVA\_HOME=*<path\_to\_java\_runtime\_home\_dir>*

1. Configure the properties as outlined above by opening and editing the application.properties file.

## Setting up the Cron Job

As a system administrator (admin user of Linux), create a cron job to run the interface job. Complete the following steps:

1. Switch the current user to a BizFlow system user.
2. Make the run.sh script executable. Type the following into the command line:

Cd <server\_dir>/caphr

$chmod 744 run-jar.sh

1. The job will run before normal business hours, e.g. 6AM, every morning. Create a new crontab with the following command:

crontab -e

#CapHR Batch job

0 6 \* \* \* /bin/sh <server\_dir>/caphr/run-jar.sh

Update the crontab command with the correct time if it is to run at a time other than 6AM.

About Deloitte

Deloitte provides audit, tax, consulting, and financial advisory services to public and private clients spanning multiple industries. With a globally connected network of member firms in 140 countries, Deloitte brings world-class capabilities and deep local expertise to help clients succeed wherever they operate. Deloitte's 165,000 professionals are committed to becoming the standard of excellence.

Deloitte's professionals are unified by a collaborative culture that fosters integrity, outstanding value to markets and clients, commitment to each other, and strength from cultural diversity. They enjoy an environment of continuous learning, challenging experiences, and enriching career opportunities. Deloitte's professionals are dedicated to strengthening corporate responsibility, building public trust, and making a positive impact in their communities.

Deloitte refers to one or more of Deloitte Touche Tohmatsu, a Swiss Verein, and its network of member firms, each of which is a legally separate and independent entity. Please see www.deloitte.com/about for a detailed description of the legal structure of Deloitte Touche Tohmatsu and its member firms. Please see <http://www.deloitte.com/us/about> for a detailed description of the legal structure of Deloitte LLP and its subsidiaries.

Internal Usage Statement

This publication is for internal distribution and use only among personnel of Deloitte Touche Tohmatsu, its member firms, and its and their affiliates. Deloitte Touche Tohmatsu, its member firms, and its and their affiliates shall not be responsible for any loss whatsoever sustained by any person who relies on this publication.

Copyright © 2012 Deloitte Development LLC. All rights reserved.

Member of Deloitte Touche Tohmatsu