



# 動作検証結果報告書 Enterprise Server と A-AUTO の連携

---

Version.2

Last updated: 2026 年 6 月



## 目次

|  |    |
|--|----|
| 1. 検証の目的 .....                                   | 3  |
| 2. 検証環境.....                                     | 3  |
| 1) OS エディション.....                                | 3  |
| 2) 製品 バージョン.....                                 | 3  |
| 3) ジョブスケジューラ実行環境.....                            | 3  |
| 4) JCL 実行機能.....                                 | 4  |
| 5) ジョブスケジューラ機能 .....                             | 4  |
| 3. 検証概要.....                                     | 5  |
| 1) A-AUTO スケジュール(ネットワーク)設定内容 .....               | 5  |
| 2) Enterprise Server インスタンスを使用した JCL の実行方法 ..... | 8  |
| 4. 検証結果.....                                     | 9  |
| 1) 正常終了ケース.....                                  | 9  |
| 2) 計画された異常終了ケース.....                             | 12 |
| 3) 未計画の異常終了ケース.....                              | 13 |
| 5. 結論 .....                                      | 15 |

各機能の詳細に関しては、製品マニュアルページからご利用になるバージョンを選択後、内容をご確認ください。

<https://www.amc.rocketsoftware.co.jp/support/manuals.asp>

## 1. 検証の目的

Enterprise Developer/Enterprise Server は、COBOL、PL/I 言語のサポート、IBM メインフレームで稼働する JCL、CICS、IMS DB/DC の互換機能を持ち、既存のアプリケーションをオープン環境の Linux や Windows などの OS 上で引き続き稼働させることができる製品です。

また、IBM メインフレームの JCL は、リプラットフォーム後もジョブスケジューラによってコントロールすることができます。

株式会社ユニリタ(旧称 BSP)の A-AUTO は、ベンダーに依存せず、多種多様なプラットフォームで稼働する運用自動化ツールとして多くのメインフレームで長い実績がありますが、オープン環境においても同様に豊富な実績があります。

本書は IBM メインフレームからオープン環境へシステム移行後もスケジュール管理を同等に行いたいとのご要望に基づき、Enterprise Server インスタンスが持つ JES 機能と、スケジューラソフトウェアである A-AUTO を連携させたジョブの実行検証を目的としています。

## 2. 検証環境

本書は下記環境で検証されました。

### 1) OS エディション

- ・Red Hat Enterprise Linux Server release 6.5
- ・Windows Server 2012 R2 Standard

### 2) 製品 バージョン

Enterprise Developer 2.2 Update2

補足1) Enterprise Server 2.2 Update2 と同等の開発用実行環境が含まれています。

補足2) 一部のスクリーンショットは Enterprise Server 11を引用しています。

### 3) ジョブスケジューラ実行環境

OS:Windows Server 2012 R2 Standard

実行製品:A-AUTO Version 8.0

#### 4) JCL 実行機能

Web ブラウザから、共通管理画面である Enterprise Server Common Web Administration (以降 ESCWA と称す)を表示して、Enterprise Server インスタンスの JES 機能を使用します。



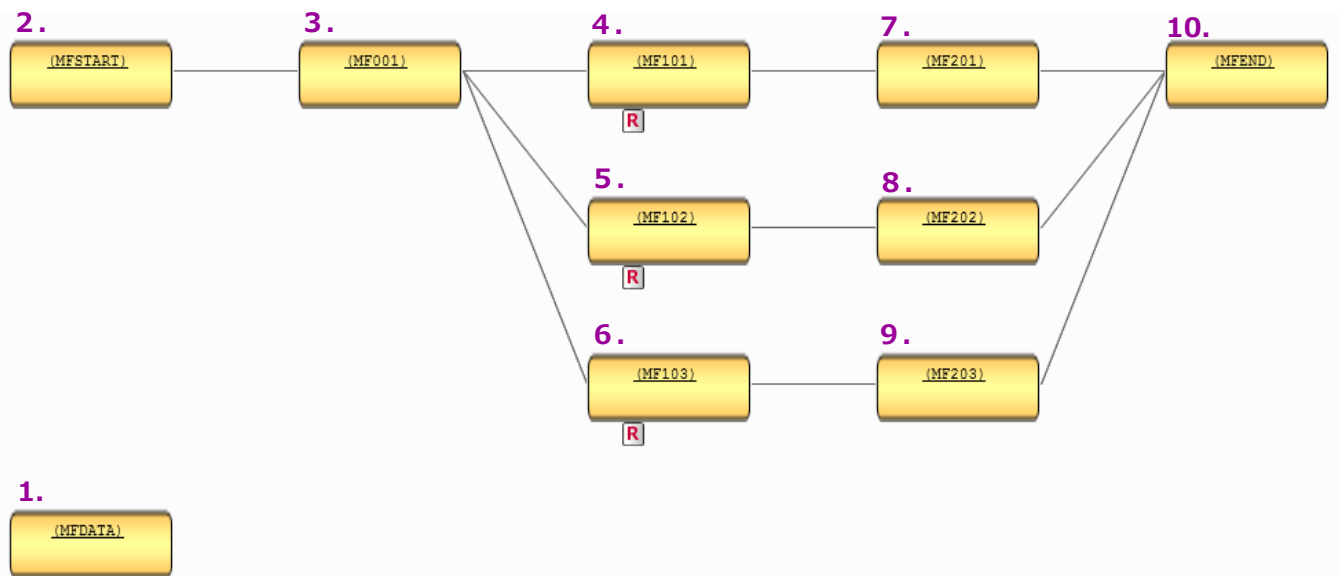
#### 5) ジョブスケジューラ機能

Web ブラウザから、A-AUTO が存在するマシンアドレスと設定ポートを指定して、A-AUTO 機能を使用します。



### 3. 検証概要

#### 1) A-AUTO スケジュール(ネットワーク)設定内容



#### ネットワークの設定)

1. **MFDATA**: データファイルを作成するネットワーク  
ジョブ: シェルから JCL を実行するジョブ
2. **MFSTART**: No.1 を先行ネットワークとして実行されるネットワーク  
ジョブ: シェルから MFSTART.gnt(RC=0)を実行する JCL を起動するジョブ
3. **MF001**: No.2 を先行ネットワークとして実行されるネットワーク  
ジョブ: シェルから MF001.gnt(RC=0)を実行する JCL を起動するジョブ
4. **MF101**: No.3 を先行ネットワークとして実行される並行処理ネットワーク  
ジョブ: シェルから MF101.gnt(RC=0)を実行する JCL を起動するジョブ
5. **MF102**: No.3 を先行ネットワークとして実行される並行処理ネットワーク  
ジョブ: シェルから MF102.gnt(RC=0)を実行する JCL を起動するジョブ
6. **MF103**: No.3 を先行ネットワークとして実行される並行処理ネットワーク  
ジョブ: シェルから MF103.gnt(RC=0)を実行する JCL を起動するジョブ
7. **MF201**: No.4 を先行ネットワークとして実行されるネットワーク  
ジョブ: シェルから MF201.gnt(RC=0)を実行する JCL を起動するジョブ
8. **MF202**: No.5 を先行ネットワークとして実行されるネットワーク  
ジョブ: シェルから MF202.gnt(RC=0)を実行する JCL を起動するジョブ
9. **MF203**: No.6 を先行ネットワークとして実行されるネットワーク  
ジョブ: シェルから MF203.gnt(RC=0)を実行する JCL を起動するジョブ

## 10. MFEND:No.7,8,9 を先行ネットワークとして実行されるネットワーク

**ジョブ:** ①シェルから MFEND.gnt(RC=0)を実行する JCL を起動するジョブ

②MFENDNR(RC=0)

①のリターンコードが 1 より小さい場合に実行されるジョブ

③MFENDAB(RC=99)

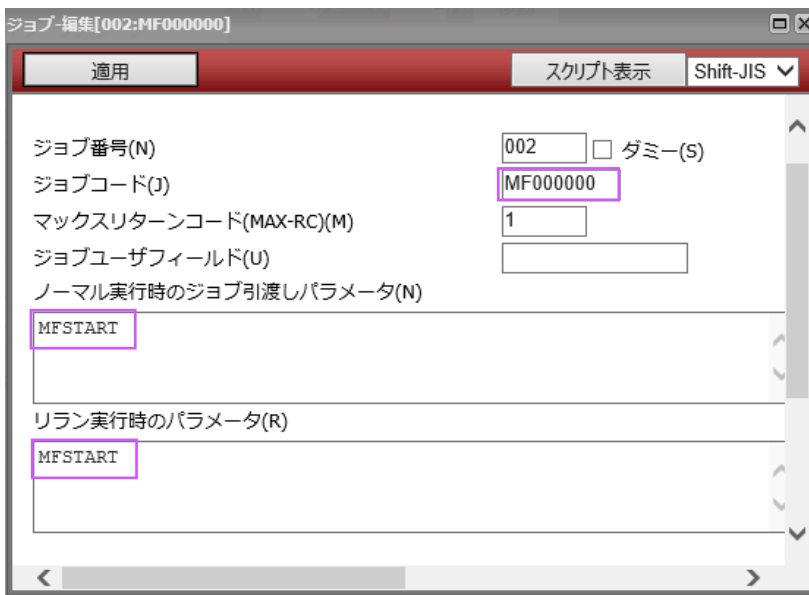
①のリターンコードが 0 より大きい場合に実行されるジョブ

**注意 1)** .gnt とは製品が提供する動的ロード実行ファイル形式を指します。

**注意 2)** RC とはリターンコードを指します。

### ジョブの設定内容)

ジョブコード(実行されるシェル名もしくはバッチファイル名)を作成しておき、呼ばれる JCL 名をパラメータへ設定します。



ジョブ-編集[002:MF000000]

適用      スクリプト表示      Shift-JIS ▼

ジョブ番号(N)      002       ダミー(S)

ジョブコード(J)      MF000000

マックスリターンコード(MAX-RC)(M)      1

ジョブユーザフィールド(U)     

ノーマル実行時のジョブ引渡しパラメータ(N)

MFSTART

リラン実行時のパラメータ(R)

MFSTART

### シェルの内容)

COBOL を実行するための環境変数と JCL の実行コマンドを、ジョブコードへ指定したシェルへ記載します。

#### 【Linux の例】

**環境変数:** COBOL 実行環境の指定あり

**実行ファイル拡張子:** なし

**引数:** \$1 で指定

```

スクリプト[2]
#!/bin/bash

export LANG=ja_JP.sjis
export COBDIR=/opt/mf/ES22U2
export COBMODE=64
export
COBCPY=$COBDIR/cpylib:/home/db2inst1/sqllib/include/cobol_mf
export
PATH=$COBDIR/bin:/home/db2inst1/sqllib/bin:/home/db2inst1/sqllib/adm:/home/db2inst1/sqllib/misc:/home/db2inst1/sqllib/db2tss/bin:/home/db2inst1/bin:$PATH
export
LIBPATH=$COBDIR/lib:/usr/lib:/lib:/home/db2inst1/sqllib/lib:/home/tarot/keit/xa
export LD_LIBRARY_PATH=/home/db2inst1/sqllib/lib64:$COBDIR/lib
export DB2INSTANCE=db2inst1
export MFBSI_DIR=/pub/mfbsi/MFVER
export ES_EMP_EXIT_1=mfbsiemx

mfbsijcl /rMFVER /j/home/aauto/mfverify/jcl/$1.jcl

```

### 【Windows の例】

環境変数: COBOL 実行環境の指定あり

実行ファイル拡張子: bat

引数: %1 で指定

```

スクリプト[1]
@echo off
pushd "C:\Program Files (x86)\Micro Focus\Enterprise Server 2012"
CALL k_createenv.bat 64

mfbsijcl /rMFVER /jC:\work\MFBSI\MFVER\jcl\%1.jcl

```

#### 【補足】

COBOL 実行環境の指定: k.createenv.bat ファイルには 2 つの環境変数を指定しています。

1. PATH へ追加  
製品インストールフォルダ配下の bin フォルダを指定します。  
例) C:¥Program Files (x86)¥Micro Focus¥Enterprise Developer¥bin64 または ¥bin
2. MFBSI\_DIR  
例) C:¥MFBSI¥MFVER  
mfbsijcl コマンドについては、後続で説明します。

## 2) Enterprise Server インスタンスを使用した JCL の実行方法

### ① cassub コマンドの利用

cassub コマンドには、スケジューラと Enterprise Server インスタンス間の通信、またはフィードバックを提供するメカニズムがありません。ジョブの終了や結果の確認は、ESCWAから Enterprise Server インスタンスのJES機能を使用して行います。

#### JESスプール機能の例)

一般 |>v| モニター |>v| ログ |>v| CICS |>v| JES |>v|

スプール | プリセット |>v| |>v| Q リスト |>v| |>v| フィルタ |>v|

<< < > >> ページ: 1 100 件

▼ フィルタ

| 名前       | ジョブID    | クラス | ユーザー  | 条件コード |
|----------|----------|-----|-------|-------|
| VSAMWRT2 | J0001032 | A   | SYSAD | 0000  |

#### DDエントリ

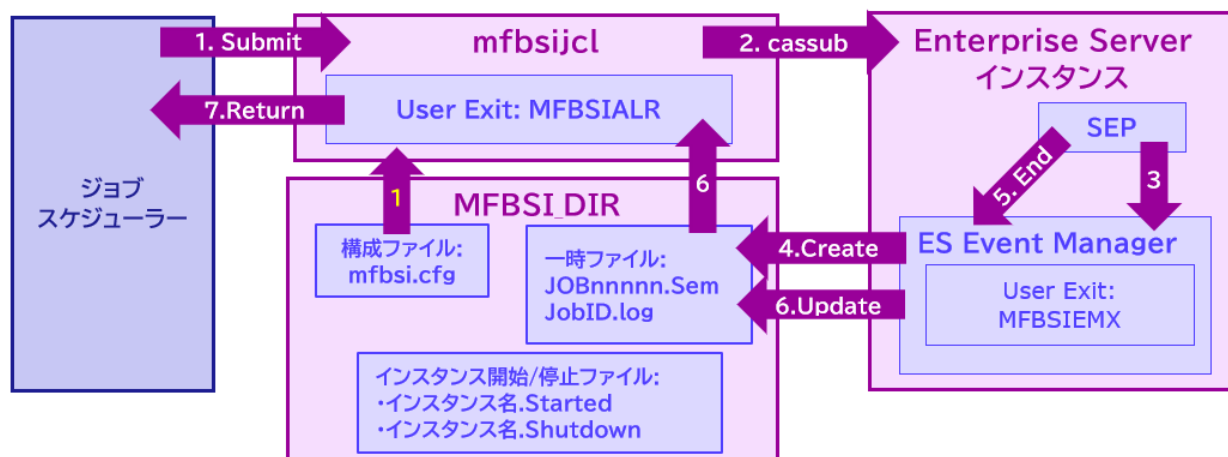
▼ フィルタ

| 名前 | 状態    | クラス | DD名      | ス...     | ステップ番... | PROC... | レコ... |
|----|-------|-----|----------|----------|----------|---------|-------|
|    | Hold  | A   | JESYSMSG |          | 0        |         | 83    |
|    | Ready | A   | SYSPRINT | DEFVSAM1 | 1        |         | 18    |
|    | Ready | A   | SYSOUT   | SORTSTEP | 2        | SORT1   | 12    |
|    | Ready | A   | SYSOUT   | APPL1    | 3        |         | 1     |
|    | Ready | A   | PRINTER  | APPL1    | 3        |         | 10    |
|    | Ready | A   | SYSPRINT | VERIFY1  | 4        |         | 41    |

### ② mfbsijcl(Batch Scheduler Integration)コマンドの利用

cassub コマンドでは実現できなかった、スケジューラと Enterprise Server インスタンス間の JCL 実行および通信を mfbsijcl コマンドがサポートします。具体的には前項の例にあるように、シエルまたはバッチファイルから mfbsijcl コマンドを使用して JCL を実行します。

#### mfbsijcl コマンドの仕組み)



機能詳細については、製品マニュアルの「プログラミング > メインフレームプログラミング > Batch Scheduler Integration」をご参照ください。

COBOL の実行環境では文字コードに ASCII-SJIS を使用しますが、ジョブログを A-AUTO に表示する際の文字コードが UTF8 であった場合、実行コマンドへコンバートコマンドを追加します。

Linux の例)

```
mfbsijcl /rMFVER /j/home/aaauto/mfverify/jcl/$1.jcl | iconv -f SJIS -t UTF8
```

これにより A-AUTO 側で受け取るジョブログは UTF8 となります。

Enterprise Server インスタンスの JES 機能の設定方法については、製品マニュアルの「ここからはじめよう > Getting Started > メインフレームチュートリアル > JCL チュートリアル」をご参照ください。

## 4. 検証結果

### 1) 正常終了ケース

A-AUTO から、全てのネットワークが正常終了するパターンを実行します。下記の画像は、データの作成が正常終了し、後続ネットワークの MFSTART の実行を待っている状態です。

|                          | ネットワーク                 | ネットワーク略称 | スケジュール日付   | 予定スタート時刻 | 実績スタート時刻 | 処理時間 | ステータス    | 属性     |
|--------------------------|------------------------|----------|------------|----------|----------|------|----------|--------|
| <input type="checkbox"/> | <a href="#">MF001</a>  |          | 2015/02/16 |          |          | 0000 | キューイング済み |        |
| <input type="checkbox"/> | <a href="#">MF101</a>  |          | 2015/02/16 |          |          | 0000 | キューイング済み | パラレルラン |
| <input type="checkbox"/> | <a href="#">MF102</a>  |          | 2015/02/16 |          |          | 0000 | キューイング済み | パラレルラン |
| <input type="checkbox"/> | <a href="#">MF103</a>  |          | 2015/02/16 |          |          | 0000 | キューイング済み | パラレルラン |
| <input type="checkbox"/> | <a href="#">MF201</a>  |          | 2015/02/16 |          |          | 0000 | キューイング済み |        |
| <input type="checkbox"/> | <a href="#">MF202</a>  |          | 2015/02/16 |          |          | 0000 | キューイング済み |        |
| <input type="checkbox"/> | <a href="#">MF203</a>  |          | 2015/02/16 |          |          | 0000 | キューイング済み |        |
| <input type="checkbox"/> | <a href="#">MFDATA</a> |          | 2015/02/16 |          | 09:20:43 | 0000 | 正常終了     |        |
| <input type="checkbox"/> | <a href="#">MFEND</a>  |          | 2015/02/16 |          |          | 0000 | キューイング済み |        |

この時点で ESCWA から Enterprise Server インスタンスのスパールを表示すると、MFDATA の実行結果が参照できます。

次に A-AUTO から MFSTART を実行して、後続ネットワークを起動させます。



MFSTART が正常終了後、後続ネットワークが実行されます。MF101、MF102、MF103 は並行処理のため、同時に実行されたことが A-AUTO から確認できます。

|                          | ネットワーク                  | ネットワーク略称 | スケジュール日付   | 予定スタート時刻 | 実績スタート時刻 | 処理時間 | ステータス    | 属性     |
|--------------------------|-------------------------|----------|------------|----------|----------|------|----------|--------|
| <input type="checkbox"/> | <a href="#">MF001</a>   |          | 2015/02/16 |          | 14:00:41 | 0000 | 正常終了     |        |
| <input type="checkbox"/> | <a href="#">MF101</a>   |          | 2015/02/16 |          | 14:00:46 | 0000 | 実行中      | パラレルラン |
| <input type="checkbox"/> | <a href="#">MF102</a>   |          | 2015/02/16 |          | 14:00:46 | 0000 | 実行中      | パラレルラン |
| <input type="checkbox"/> | <a href="#">MF103</a>   |          | 2015/02/16 |          | 14:00:46 | 0000 | 実行中      | パラレルラン |
| <input type="checkbox"/> | <a href="#">MF201</a>   |          | 2015/02/16 |          |          | 0000 | キューイング済み |        |
| <input type="checkbox"/> | <a href="#">MF202</a>   |          | 2015/02/16 |          |          | 0000 | キューイング済み |        |
| <input type="checkbox"/> | <a href="#">MF203</a>   |          | 2015/02/16 |          |          | 0000 | キューイング済み |        |
| <input type="checkbox"/> | <a href="#">MFDATA</a>  |          | 2015/02/16 |          | 09:20:43 | 0000 | 正常終了     |        |
| <input type="checkbox"/> | <a href="#">MFEND</a>   |          | 2015/02/16 |          |          | 0000 | キューイング済み |        |
| <input type="checkbox"/> | <a href="#">MFSTART</a> |          | 2015/02/16 |          | 11:29:32 | 0000 | 正常終了     |        |

一連の処理が正常終了しました。

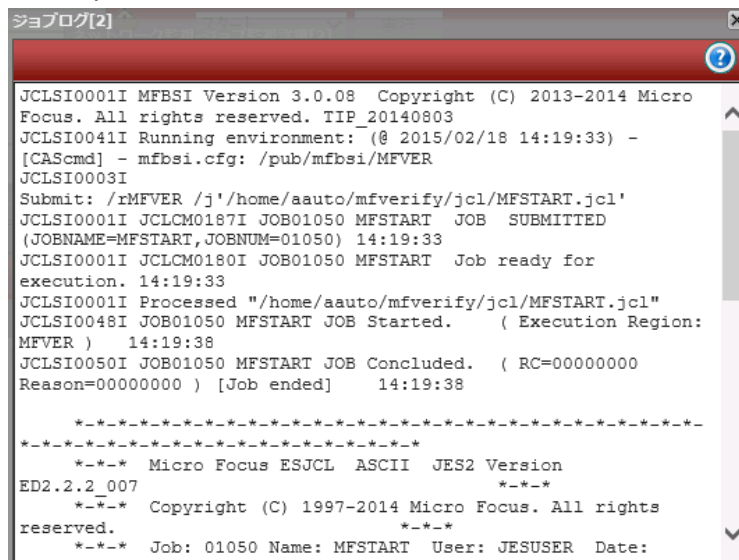
|                          | ネットワーク                  | ネットワーク略称 | スケジュール日付   | 予定スタート時刻 | 実績スタート時刻 | 処理時間 | ステータス | 属性     |
|--------------------------|-------------------------|----------|------------|----------|----------|------|-------|--------|
| <input type="checkbox"/> | <a href="#">MF001</a>   |          | 2015/02/17 |          | 09:32:16 | 0000 | 正常終了  |        |
| <input type="checkbox"/> | <a href="#">MF101</a>   |          | 2015/02/17 |          | 09:32:22 | 0000 | 正常終了  | パラレルラン |
| <input type="checkbox"/> | <a href="#">MF102</a>   |          | 2015/02/17 |          | 09:32:22 | 0000 | 正常終了  | パラレルラン |
| <input type="checkbox"/> | <a href="#">MF103</a>   |          | 2015/02/17 |          | 09:32:22 | 0000 | 正常終了  | パラレルラン |
| <input type="checkbox"/> | <a href="#">MF201</a>   |          | 2015/02/17 |          | 09:32:41 | 0000 | 正常終了  |        |
| <input type="checkbox"/> | <a href="#">MF202</a>   |          | 2015/02/17 |          | 09:29:00 | 0000 | 正常終了  |        |
| <input type="checkbox"/> | <a href="#">MF203</a>   |          | 2015/02/17 |          | 09:30:06 | 0000 | 正常終了  |        |
| <input type="checkbox"/> | <a href="#">MFEND</a>   |          | 2015/02/17 |          | 09:54:08 | 0000 | 正常終了  |        |
| <input type="checkbox"/> | <a href="#">MFSTART</a> |          | 2015/02/17 |          | 09:31:46 | 0000 | 正常終了  |        |

前項で説明した mfbisjcl を実行するための mfbisi 構成ファイルには、ジョブログを標準出力へ出力するように指定しています。これにより、A-AUTO から各ジョブのジョブログが確認可能です。

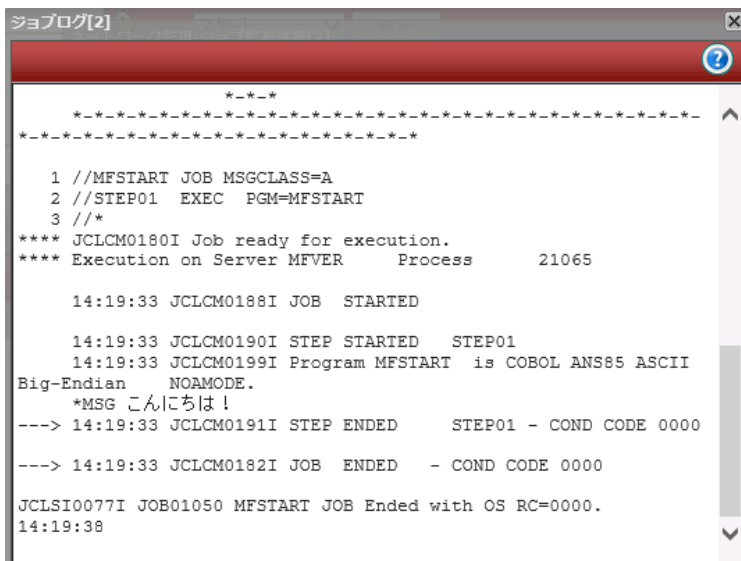
ジョブログ表示ボタンにて内容を表示します。



Enterprise Server インスタンスで実行されたジョブ番号が確認できます。



表示文字コードを Shift-JIS の場合は、プログラムで出力した日本語も確認できます。



```

*--*--*
*--*--*--*--*--*--*--*--*--*--*--*--*--*--*--*--*--*--*--*--*--*--*--*--*--*
1 //MFSTART JOB MSGCLASS=A
2 //STEP01 EXEC PGM=MFSTART
3 /**
**** JCLCM0180I Job ready for execution.
**** Execution on Server MFVER      Process      21065

14:19:33 JCLCM0188I JOB   STARTED

14:19:33 JCLCM0190I STEP  STARTED   STEP01
14:19:33 JCLCM0199I Program MFSTART is COBOL ANS85 ASCII
Big-Endian NOAMODE.
*MSG こんにちは！
--> 14:19:33 JCLCM0191I STEP ENDED   STEP01 - COND CODE 0000

--> 14:19:33 JCLCM0182I JOB   ENDED   - COND CODE 0000

JCLSI0077I JOB01050 MFSTART JOB Ended with OS RC=0000.
14:19:38
    
```

ESCWA から Enterprise Server インスタンスの JES スプールを表示すると、A-AUTO で実行指示された JCL の COND CODE はゼロで返却されていることが確認できます。

また、MFEND が正常終了時の後続ジョブである MFENDNR が実行され、正常終了していることが確認できます。

### A-AUTO の設定例)

#### ① MFENDNR:MFEND リターンコードが 1 より小さい場合に実行

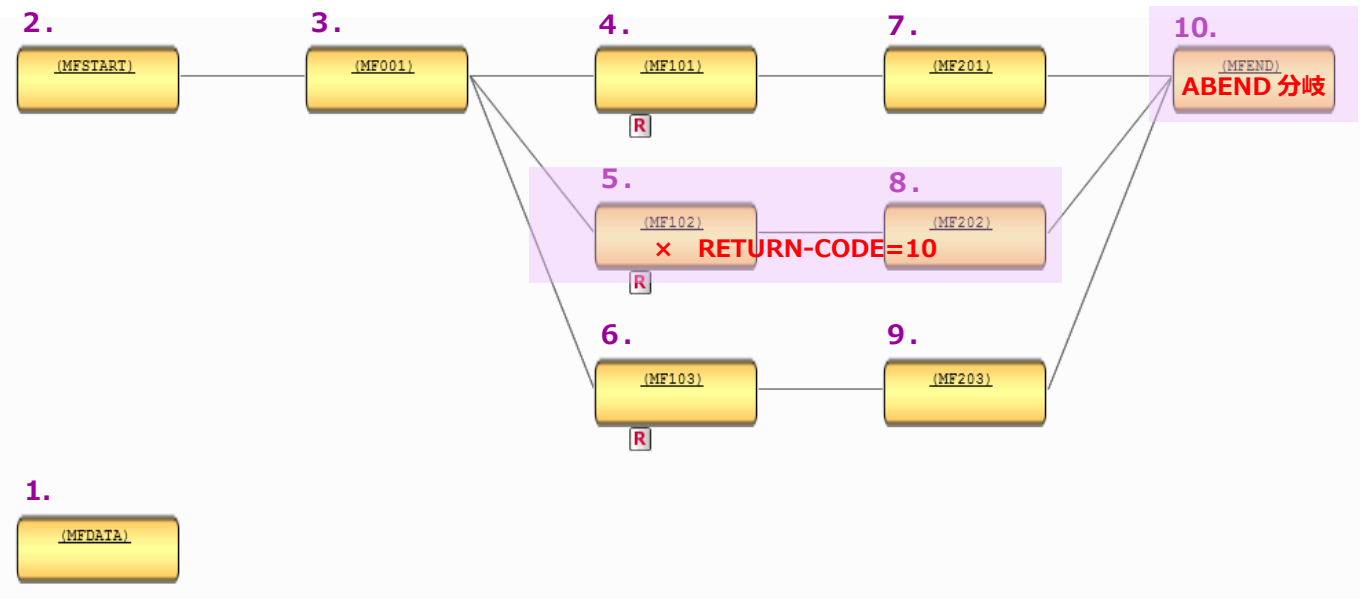
| ジョブ番号(N)                             | <input type="text" value="011"/> <input type="checkbox"/> ダミー(S) | <b>分岐設定</b><br><input checked="" type="radio"/> リターンコード分岐(C)<br><table border="1"> <thead> <tr> <th>先行ジョブ番号</th> <th>分岐設定</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>010(MF000000) RC&lt;1</td> </tr> <tr> <td>2</td> <td></td> </tr> <tr> <td>3</td> <td></td> </tr> <tr> <td>4</td> <td></td> </tr> </tbody> </table> | 先行ジョブ番号 | 分岐設定 | 1 | 010(MF000000) RC<1 | 2 |  | 3 |  | 4 |  |
|--------------------------------------|--|--|---------|------|---|--------------------|---|--|---|--|---|--|
| 先行ジョブ番号                              | 分岐設定   |  |         |      |   |                    |   |  |   |  |   |  |
| 1                                    | 010(MF000000) RC<1   |  |         |      |   |                    |   |  |   |  |   |  |
| 2                                    |  |  |         |      |   |                    |   |  |   |  |   |  |
| 3                                    |  |  |         |      |   |                    |   |  |   |  |   |  |
| 4                                    |  |  |         |      |   |                    |   |  |   |  |   |  |
| ジョブコード(J)                            | <input type="text" value="MF000000"/>                            |  |         |      |   |                    |   |  |   |  |   |  |
| マックスリターンコード(MAX-RC)(M)               | <input type="text" value="1"/>                                   |  |         |      |   |                    |   |  |   |  |   |  |
| ジョブユーザフィールド(U)                       | <input type="text"/>   |  |         |      |   |                    |   |  |   |  |   |  |
| ノーマル実行時のジョブ引渡しパラメータ(N)               | <input type="text"/>   |  |         |      |   |                    |   |  |   |  |   |  |
| <input type="text" value="MFENDNR"/> |  |  |         |      |   |                    |   |  |   |  |   |  |

#### ② MFENDAB:MFEND リターンコードが 0 より大きい場合に実行

| ジョブ番号(N)                             | <input type="text" value="012"/> <input type="checkbox"/> ダミー(S) | <input checked="" type="radio"/> リターンコード分岐(C)<br><table border="1"> <thead> <tr> <th>先行ジョブ番号</th> <th>分岐設定</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>010(MF000000) RC&gt;0</td> </tr> <tr> <td>2</td> <td></td> </tr> <tr> <td>3</td> <td></td> </tr> <tr> <td>4</td> <td></td> </tr> </tbody> </table> | 先行ジョブ番号 | 分岐設定 | 1 | 010(MF000000) RC>0 | 2 |  | 3 |  | 4 |  |
|--------------------------------------|--|---|---------|------|---|--------------------|---|--|---|--|---|--|
| 先行ジョブ番号                              | 分岐設定   |   |         |      |   |                    |   |  |   |  |   |  |
| 1                                    | 010(MF000000) RC>0   |   |         |      |   |                    |   |  |   |  |   |  |
| 2                                    |  |   |         |      |   |                    |   |  |   |  |   |  |
| 3                                    |  |   |         |      |   |                    |   |  |   |  |   |  |
| 4                                    |  |   |         |      |   |                    |   |  |   |  |   |  |
| ジョブコード(J)                            | <input type="text" value="MF000000"/>                            |   |         |      |   |                    |   |  |   |  |   |  |
| マックスリターンコード(MAX-RC)(M)               | <input type="text" value="1"/>                                   |   |         |      |   |                    |   |  |   |  |   |  |
| ジョブユーザフィールド(U)                       | <input type="text"/>   |   |         |      |   |                    |   |  |   |  |   |  |
| ノーマル実行時のジョブ引渡しパラメータ(N)               | <input type="text"/>   |   |         |      |   |                    |   |  |   |  |   |  |
| <input type="text" value="MFENDAB"/> |  |   |         |      |   |                    |   |  |   |  |   |  |

## 2) 計画された異常終了ケース

一部のプログラムから、計画された範囲内のリターンコードを返却するように変更します。



- 5. MF102: Max リターンコードを 20 として、プログラムがリターンコード 10 を返却
- 8. MF202: Max リターンコードを 20 として、プログラムがリターンコード 10 を返却
- 10. MFEND: Max リターンコードを 20 として、プログラムがリターンコード 10 を返却

上記の MF102、MF202、MFEND を A-AUTO から再度実行します。

スタート

エージェント名(A)

モニタID(M)

ネットワークID(C)

ネットワークの MF102、MF202、MFEND は Max リターンコードに 20 を設定しているため、異常終了とはならず、後続ネットワークが継続実行されますが、MFEND ではリターンコード分岐により MFENDAB が実行され、リターンコードに 99 が返却され、異常終了となります。

|                          | ネットワーク  | ネットワーク略称 | スケジュール日付   | 予定スタート時刻 | 実績スタート時刻 | 処理時間 | ステータス | 属性     |
|--------------------------|---------|----------|------------|----------|----------|------|-------|--------|
| <input type="checkbox"/> | MF001   |          | 2015/02/17 |          | 09:32:16 | 0000 | 正常終了  |        |
| <input type="checkbox"/> | MF101   |          | 2015/02/17 |          | 09:32:22 | 0000 | 正常終了  | パラレルラン |
| <input type="checkbox"/> | MF102   |          | 2015/02/17 |          | 10:38:34 | 0000 | 正常終了  | パラレルラン |
| <input type="checkbox"/> | MF103   |          | 2015/02/17 |          | 09:32:22 | 0000 | 正常終了  | パラレルラン |
| <input type="checkbox"/> | MF201   |          | 2015/02/17 |          | 09:32:41 | 0000 | 正常終了  |        |
| <input type="checkbox"/> | MF202   |          | 2015/02/17 |          | 10:38:43 | 0000 | 正常終了  |        |
| <input type="checkbox"/> | MF203   |          | 2015/02/17 |          | 09:30:06 | 0000 | 正常終了  |        |
| <input type="checkbox"/> | MFEND   |          | 2015/02/17 |          | 10:38:56 | 0000 | 異常終了  |        |
| <input type="checkbox"/> | MFSTART |          | 2015/02/17 |          | 09:31:46 | 0000 | 正常終了  |        |

異常終了により、A-AUTO のアラートが表示されます。

| ネットワークID | スケジュール日付   | エージェント名 | アラート種類 |
|----------|------------|---------|--------|
| MFEND    | 2015/02/17 | AUTO02  | 異常終了   |

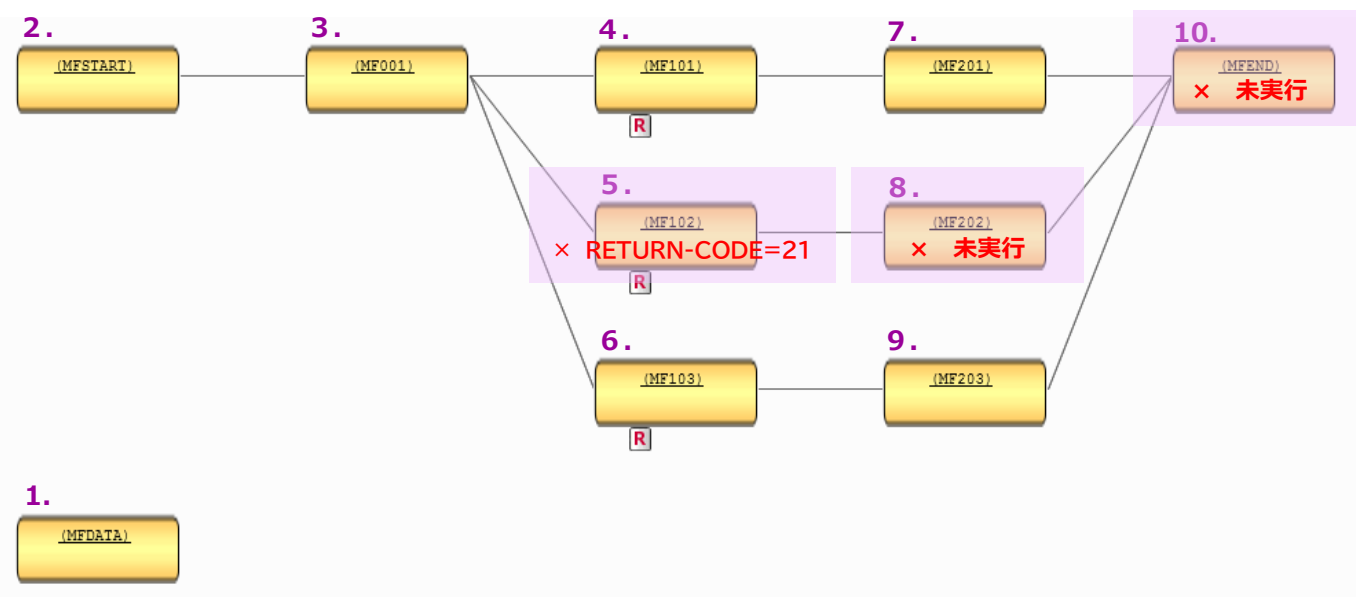
アラートから実績情報を表示させ、終了コードを確認します。

| 項目        | 実績    |
|-----------|-------|
| ジョブ・ステータス | 異常終了  |
| 終了ステータス   | MaxRC |
| 終了コード     | 0099  |
| 異常終了ステップ  |       |
| OS-JOB 番号 | UNIX  |

ESCWA から Enterprise Server インスタンスのJESスプールを表示すると、A-AUTO で実行指示された JCL からのリターンコードが表示されます。

### 3) 未計画の異常終了ケース

一部のプログラムから、計画外のリターンコードを返却するように変更します。



- 5. MF102:Max リターンコードを 20 として、プログラムがリターンコード 21 を返却
- 8. MF202:Max リターンコードを 20 として、プログラムがリターンコード 0 を返却
- 10. MFEND:Max リターンコードを 20 として、プログラムがリターンコード 0 を返却

上記の MF102、MF202、MFEND を A-AUTO から再度実行します。

スタート

---

エージェント名(A)     ▼

モニタID(M)         ▼

ネットワークID(C)

ネットワークの MF102 はリターンコード 21 を返却して、未計画の異常終了となり、後続ネットワークは実行されません。後続ネットワークである MF202、MFEND のスタート時刻を確認すると、前回終了時のままで実行されていないことがわかります。

|                          | ネットワーク  | ネットワーク略称 | スケジュール日付   | 予定スタート時刻 | 実績スタート時刻 | 処理時間 | ステータス | 属性     |
|--------------------------|---------|----------|------------|----------|----------|------|-------|--------|
| <input type="checkbox"/> | MF001   |          | 2015/02/17 |          | 09:32:16 | 0000 | 正常終了  |        |
| <input type="checkbox"/> | MF101   |          | 2015/02/17 |          | 09:32:22 | 0000 | 正常終了  | パラレルラン |
| <input type="checkbox"/> | MF102   |          | 2015/02/17 |          | 11:23:42 | 0000 | 異常終了  | パラレルラン |
| <input type="checkbox"/> | MF103   |          | 2015/02/17 |          | 09:32:22 | 0000 | 正常終了  | パラレルラン |
| <input type="checkbox"/> | MF201   |          | 2015/02/17 |          | 09:32:41 | 0000 | 正常終了  |        |
| <input type="checkbox"/> | MF202   |          | 2015/02/17 |          | 10:38:43 | 0000 | 正常終了  |        |
| <input type="checkbox"/> | MF203   |          | 2015/02/17 |          | 09:30:06 | 0000 | 正常終了  |        |
| <input type="checkbox"/> | MFEND   |          | 2015/02/17 |          | 10:38:56 | 0000 | 異常終了  |        |
| <input type="checkbox"/> | MFSTART |          | 2015/02/17 |          | 09:31:46 | 0000 | 正常終了  |        |

未計画の異常終了時により、A-AUTO のアラートが表示されます。

アラート一覧

スタート ▼    実行

---

削除

| <input type="checkbox"/> | ネットワークID | スケジュール日付   | エージェント名 | アラート種類 |
|--------------------------|----------|------------|---------|--------|
| <input type="checkbox"/> | MF102    | 2015/02/17 | AUTO02  | 異常終了   |

アラートから実績情報を表示させ、終了コードを確認します。

| 実績情報      |       |
|-----------|-------|
| 項目        | 実績    |
| ジョブ・ステータス | 異常終了  |
| 終了ステータス   | MaxRC |
| 終了コード     | 0021  |
| 異常終了ステップ  |       |
| OS-JOB 番号 | UNIX  |

## 5. 結論

以上の結果から、オープン環境における Enterprise Sever と A-AUTO の連携に関して、特筆する問題点はありませんでした。また、下記事項において連携可能なことを確認しました。

1. A-AUTO から、シェルまたはバッチファイルをキックし、JCL が Enterprise Server インスタンスで実行される
2. A-AUTO に設定したジョブパラメータが正常にシェルまたはバッチファイルへ引き渡され、Enterprise Server インスタンスで実行される
3. A-AUTO に先行ネットワークや先行ジョブを設定し、リターンコードによるジョブ分岐が実現できる
4. A-AUTO にネットワーク並行処理を設定し、正常に実行される
5. Web ブラウザから、A-AUTO、ESCWAを利用して実行状況が確認できる
6. mfbsijcl コマンドを利用して、ジョブを Enterprise Server インスタンスで実行することにより、JCL の COND CODE を A-AUTO へ返却できる
7. mfbsi 構成ファイルの設定値により、ジョブログが A-AUTO へ連携され、表示できる
8. プログラムで出力している日本語が、A-AUTO で正常に表示可能である
9. mfbsicjl コマンドに文字コードのコンバートコマンドを組み合わせることにより、コンバートされたジョブログが A-AUTO へ連携され、表示できる
10. ESCWA、A-AUTOともに、異なる OS 上で稼働しているジョブであることを意識することなく、Windows 上のWebブラウザから運用可能で、結果確認も同一管理画面から確認できる
11. Enterprise Server インスタンスのJES機能と A-AUTO で管理しているジョブ実行結果に相違がない

以上、動作検証結果の報告となります。