
Micro Focus 実行環境と A-AUTO 連携 動作検証報告書

2015 年 4 月 1 日

マイクロフォーカス株式会社

1. はじめに

Micro Focus の Enterprise Developer/Enterprise Server は、メインフレームで稼働する JCL や CICS/IMS アプリケーションをそのまま Linux/UNIX や Windows 上で稼働させ、リホストによるコスト削減を実現する製品です。

バッチ処理のリホストにおいては JCL や COBOL、PL/I プログラムはそのまま再利用することができますが、その運用はオペレーティングシステムの変更によって大きく変わってきます。メインフレーム上では多くの場合ジョブスケジューラによって運用が自動化されていますので、リホスト後も同様な運用自動化が可能である必要があります。

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各社のメインフレームで長い実績がありますが、Linux/UNIX や Windows でも同様に稼働しており豊富な実績があります。

本書では Linux 版と Windows 版の A-AUTO を使用して、Micro Focus Enterprise Server にリホストされた JCL の運転をスケジュールとして自動化することが可能であることを検証しました。

2. 検証目的

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、Micro Focus 実行環境とスケジューラソフトウェアである A-AUTO を使用したジョブ実行の検証を目的としています。

3. 検証環境

3.1 JCL 実行環境

UNIX OS ; Red Hat Enterprise Linux Server release 6.5

Windows OS ; Windows Server 2012 R2 Standard

実行製品 ; Micro Focus Enterprise Server 2.2 Update2

3.2 スケジューラ実行環境

Windows OS ; Microsoft Windows Server 2012 R2 Standard

実行製品 ; A-AUTO Version 8.0

3.3 JCL 運用機能

Internet Explorer を使用して JCL 実行環境が存在するマシンアドレスの 86 (デフォルト) ポートを指定します。



3.4 スケジューラ運用機能

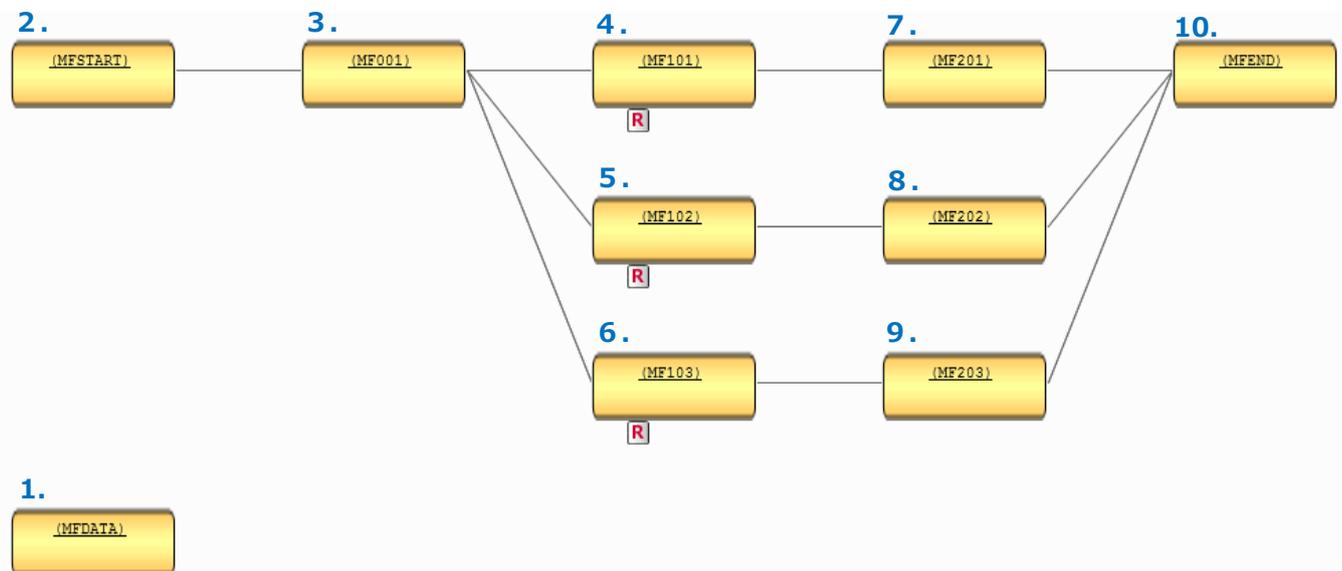
Internet Explorer を使用して A-AUTO が存在するマシンアドレスの設定ポートを指定します。



4. 検証概要

4.1 A-AUTO スケジュール（ネットワーク）設定内容

【検証ネットワーク】



【ネットワーク初期設定】

注意 1) .gnt とは Micro Focus が提供する動的ロード実行ファイル形式です。

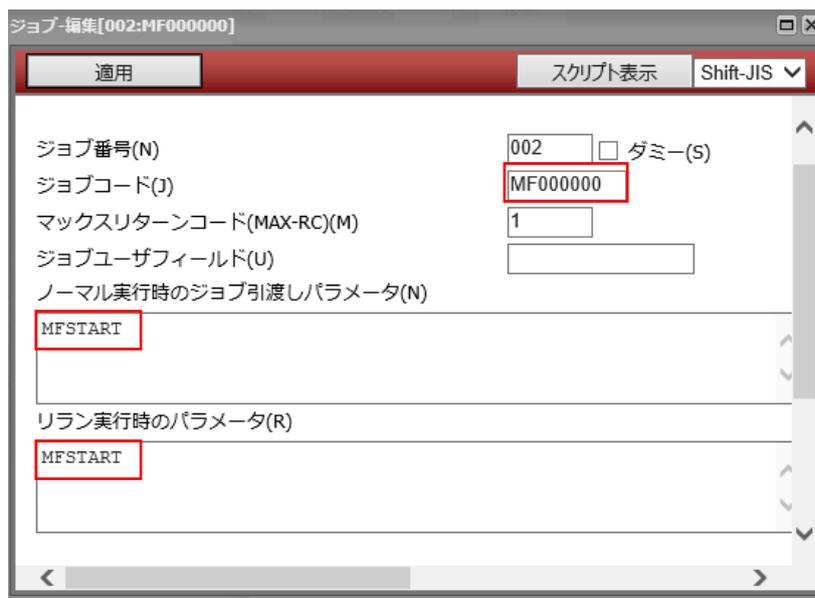
注意 2) RC とはリターンコードを指します。

- | | | |
|------------|-----|--|
| 1. MFDATA | ・・・ | データファイルを作成するネットワーク |
| ジョブ | → | JCL をシェルから起動するジョブ |
| 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 | → | MFSTART.gnt (RC=0) を実行する JCL をシェルから起動するジョブ |
| 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 | → | MF001.gnt (RC=0) を実行する JCL をシェルから起動するジョブ |
| 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 | → | MF101.gnt (RC=0) を実行する JCL をシェルから起動するジョブ |
| 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 | → | MF102.gnt (RC=0) を実行する JCL をシェルから起動するジョブ |
| 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 | → | MF103.gnt (RC=0) を実行する JCL をシェルから起動するジョブ |

- 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 → MF201.gnt (RC=0) を実行する JCL をシェルから起動するジョブ
- 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 → MF202.gnt (RC=0) を実行する JCL をシェルから起動するジョブ
- 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 → MF203.gnt (RC=0) を実行する JCL をシェルから起動するジョブ
- 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 MFEND.gnt (RC=0) を実行する JCL をシェルから起動するジョブ
 ② MFENDNR (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
 ③ MFENDAB (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

【ジョブ設定内容】

ジョブコード（実行されるシェル名）は汎用的なものを作成しておき、呼ばれる JCL 名をパラメータへ設定します。



COBOL を実行するための環境変数と JCL 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】

環境変数：COBOL 実行環境の指定あり

実行シェルファイル拡張子：なし

引数：\$1 で指定



```
#!/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:/hom
e/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

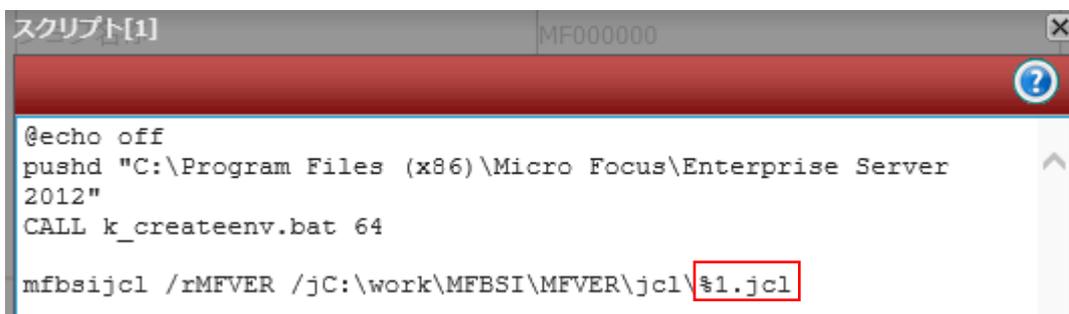
mfbsijcl /rMFVER /j/home/aauto/mfverify/jcl/$1.jcl
```

【Windows】

環境変数：COBOL 実行環境の指定あり

スクリプトファイル拡張子：bat

引数：%1 で指定



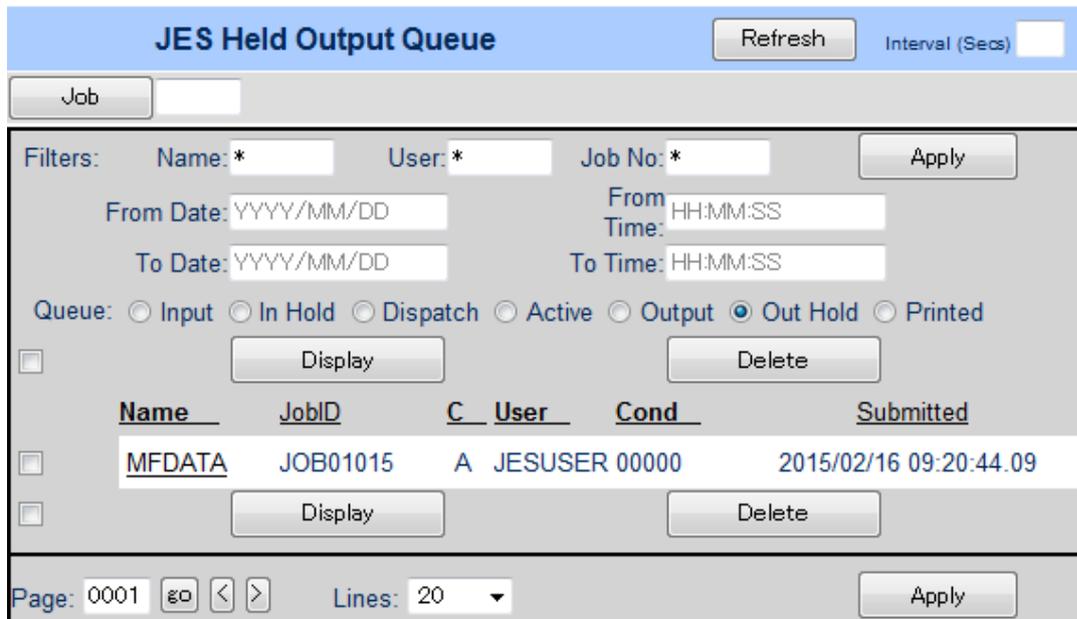
```
@echo off
pushd "C:\Program Files (x86)\Micro Focus\Enterprise Server
2012"
CALL k_createenv.bat 64

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

4.2 Micro Focus Enterprise Server 概要

【JCL 実行方法 : CASSUB】

ジョブを Enterprise Server で実行する方法として CASSUB コマンドを用意していますが、これにはスケジューラと Enterprise Server 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。Enterprise Server を使用してジョブを実行および監視することが可能ですが、これには下記のような画面にて対話操作が必要となります。

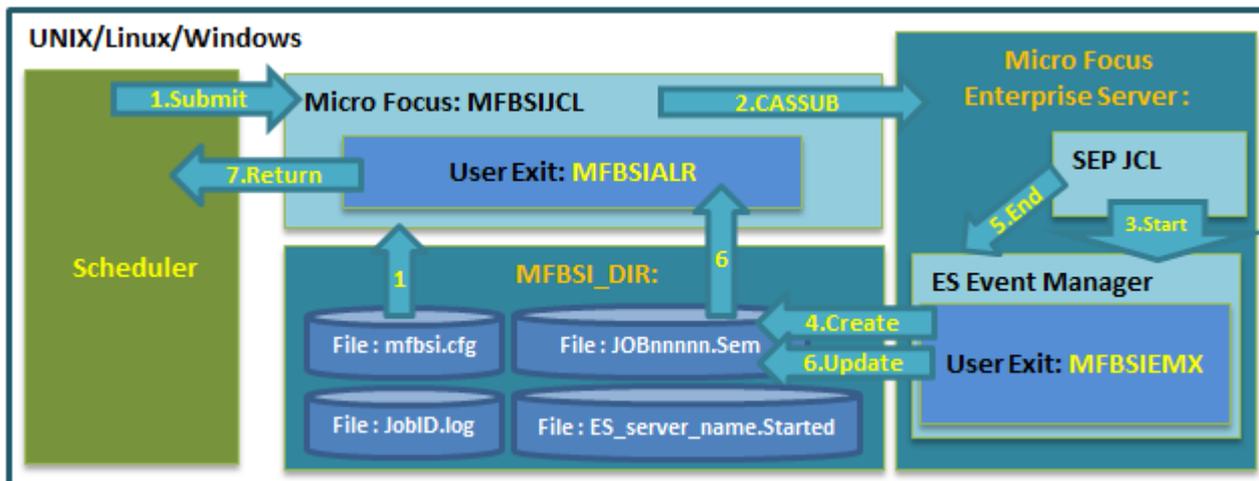


【JCL 実行方法 : Micro Focus Batch Scheduler Integration (MFBSI)】

CASSUB では実現できなかったスケジューラと Enterprise Server 間の JCL 実行および通信が、MFBSI を使用することにより可能になります。具体的にはシェルから MFBSIJCL コマンドを使用して JCL を実行します。

参照 URL) <http://www.microfocus.co.jp/manuals/ED22U2/html/GUID-7A21EA67-39FD-4402-96F3-FA27544168E2.html>

【MFBSIJCL 実行概要図】



【MFBSI 設定ファイル】

A-AUTO 画面へ実行したジョブログを表示させるには MFBSI 設定ファイルを変更します。

ファイル名 : mfbsi.cfg

変更値 : JESYSMSG と SJobID=StdOut 記述行を有効行へ変更します。

【ジョブログファイルの文字コード変換】

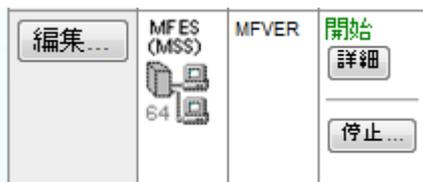
上記 MFBSI 設定ファイルの指定によりジョブログを A-AUTO へ表示する際の文字コードが例えば SJIS ではなく UTF8 であった場合、実行シェルのコマンドへコンバートコマンドを下記太字部分のように追加することが可能です。

Linux 例) `mfbsijcl /rMFVER /j/home/aauto/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 となりますので、表示時は A-AUTO 表示機能で UTF8 を選択することになります。

【Enterprise Server 作成概要】

1. JES リージョンを作成します。



2. 必要な環境変数を設定します。(下記は Linux の場合)

```
構成情報
[ES-Environment]
MFBSI_DIR=/pub/mfbsi/MFVER
ES_EMP_EXIT_1=mfbsiemx
ES_ALLOC_OVERRIDE=/home/aauto/mfverify/AllocRule.txt
```

3. リージョンの JES 機能を設定します。(下記は Linux パスの場合)



5. 検証結果

5.1 正常終了時

【ネットワークの設定】

初期設定のまま実行させます。

【ネットワークの初期状態】

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。データの作成は正常終了しており、後続ネットワークは 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"/>	MF001		2015/02/16			0000	キューイング済み	
<input type="checkbox"/>	MF101		2015/02/16			0000	キューイング済み	パラレルラン
<input type="checkbox"/>	MF102		2015/02/16			0000	キューイング済み	パラレルラン
<input type="checkbox"/>	MF103		2015/02/16			0000	キューイング済み	パラレルラン
<input type="checkbox"/>	MF201		2015/02/16			0000	キューイング済み	
<input type="checkbox"/>	MF202		2015/02/16			0000	キューイング済み	
<input type="checkbox"/>	MF203		2015/02/16			0000	キューイング済み	
<input type="checkbox"/>	MFDATA		2015/02/16		09:20:43	0000	正常終了	
<input type="checkbox"/>	MFEND		2015/02/16			0000	キューイング済み	

【Enterprise Server の初期状態】

この時点で Enterprise Server のスプールを参照すると、MFDATA の実行結果が参照できます。

JOB01015
Refresh

Job
01015

JOB01015 Name: MFDATA

Release Class: A

Update User: JESUSER

Status: **Output Hold**

Priority: 00

COND: 00000

JCLCM0188I JOB01015 MFDATA JOB STARTED 09:20:44

JCLCM0182I JOB01015 MFDATA JOB ENDED - COND CODE 0000 09:20:44

Status	Class	DD Name	Step	Nbr.	Proc Step	Records
Hold	A	JESYSMSG		0		45

Details

【一連ネットワークの実行】

MFSTART を実行して、後続ネットワークを起動させます。

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)

【ネットワークの実行状態】

MFSTART を実行して正常終了後、後続ネットワークが実行されます。MF101、MF102、MF103 は並行処理なので同時に実行されています。

	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		2015/02/16		14:00:41	0000	正常終了	
<input type="checkbox"/>	MF101		2015/02/16		14:00:46	0000	実行中	パラレルラン
<input type="checkbox"/>	MF102		2015/02/16		14:00:46	0000	実行中	パラレルラン
<input type="checkbox"/>	MF103		2015/02/16		14:00:46	0000	実行中	パラレルラン
<input type="checkbox"/>	MF201		2015/02/16			0000	キューイング済み	
<input type="checkbox"/>	MF202		2015/02/16			0000	キューイング済み	
<input type="checkbox"/>	MF203		2015/02/16			0000	キューイング済み	
<input type="checkbox"/>	MFDATA		2015/02/16		09:20:43	0000	正常終了	
<input type="checkbox"/>	MFEND		2015/02/16			0000	キューイング済み	
<input type="checkbox"/>	MFSTART		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。

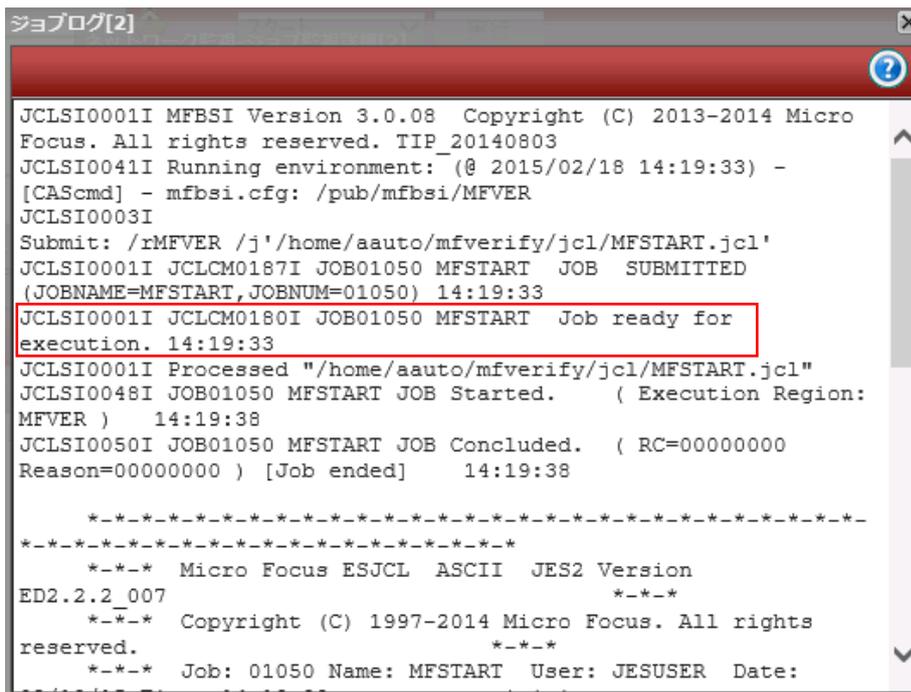
	ネットワーク	ネットワーク略称	スケジュール日付	予定スタート時刻	実績スタート時刻	処理時間	ステータス	属性
<input type="checkbox"/>	MF001		2015/02/17		09:32:16	0000	正常終了	
<input type="checkbox"/>	MF101		2015/02/17		09:32:22	0000	正常終了	パラレルラン
<input type="checkbox"/>	MF102		2015/02/17		09:32:22	0000	正常終了	パラレルラン
<input type="checkbox"/>	MF103		2015/02/17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

【ジョブログの確認】

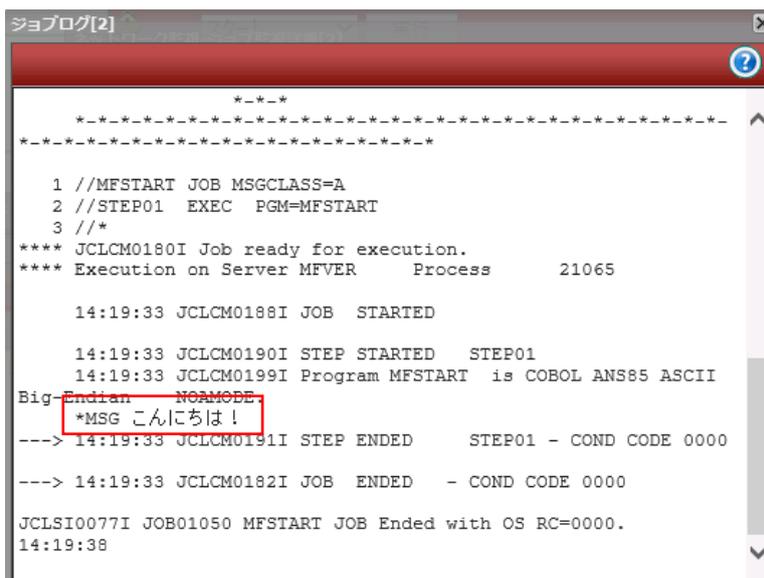
MFBSI 設定ファイルへジョブログを標準出力へ出力するように指定しているため、A-AUTO から各ジョブのジョブログが確認可能です。ジョブログ表示ボタンにて内容を表示します。



【ジョブログ】 Enterprise Server で実行されたジョブ番号が確認できます。



【ジョブログ】 表示文字コードを Shift-JIS にしているため、プログラムで出力した日本語も確認できます。



【Enterprise Server の実行状態】

Enterprise Server のスプールを表示すると、A-AUTO で実行指示された JCL が COND CODE ゼロで返却されていることが確認できます。

Name	JobID	C	User	Cond	Submitted
MF202	JOB01030	A	JESUSER	00000	2015/02/17 09:29:00.54
MF203	JOB01031	A	JESUSER	00000	2015/02/17 09:30:06.55
MFSTART	JOB01032	A	JESUSER	00000	2015/02/17 09:31:46.90
MF001	JOB01033	A	JESUSER	00000	2015/02/17 09:32:16.99
MF101	JOB01034	A	JESUSER	00000	2015/02/17 09:32:22.23
MF102	JOB01035	A	JESUSER	00000	2015/02/17 09:32:23.24
MF103	JOB01036	A	JESUSER	00000	2015/02/17 09:32:24.24
MF201	JOB01037	A	JESUSER	00000	2015/02/17 09:32:41.15
MFEND	JOB01038	A	JESUSER	00000	2015/02/17 09:49:32.02
MFENDNR	JOB01039	A	JESUSER	00000	2015/02/17 09:54:08.74

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。

- 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) ダミー(S)

ジョブコード(J)

マックスリターンコード(MAX-RC)(M)

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)

分岐設定

リターンコード分岐(C)

先行ジョブ番号	分岐設定
1 010(MF000000) ▼	RC<1
2	▼
3	▼
4	▼

- 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) ダミー(S)

ジョブコード(J)

マックスリターンコード(MAX-RC)(M)

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)

分岐設定

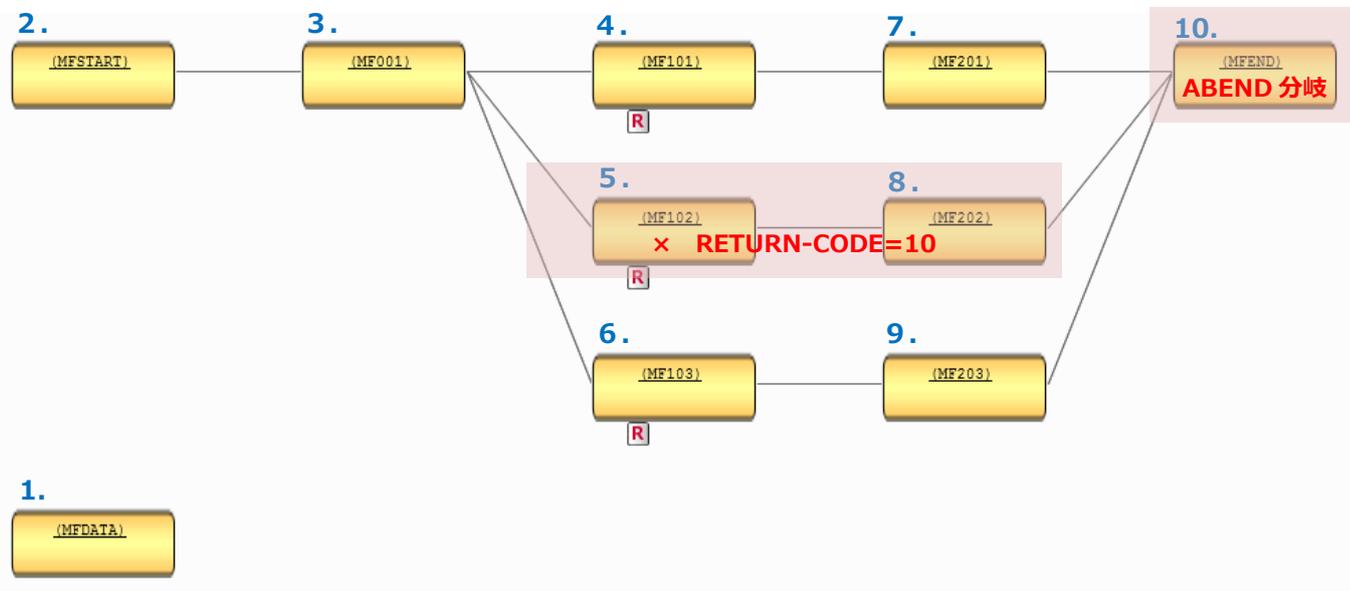
リターンコード分岐(C)

先行ジョブ番号	分岐設定
1 010(MF000000) ▼	RC>0
2	▼
3	▼
4	▼

5.2 計画済の異常終了時

【ネットワークの設定】

下記のように一部のプログラムからのリターンコードを変更します。



- 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

【ネットワークの初期状態】

MFDATA は終了していますので、前回終了時と変化はありません。

【Enterprise Server の初期状態】

前回終了時と変化はありません。

【一連ネットワークの実行】

変更プログラムが影響する、MF102、MF202、MFEND を再度実行します。

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)

【ネットワークの実行状態】

ネットワークの MF102、MF202、MFEND は計画済として Max リターンコード 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へ継続されていますが、MFEND ではリターンコード分岐により MFENDAB が実行され、リターンコードへ 99 が返却されているゆえ異常終了となっています。

	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	

異常終了時のアラート一覧



The screenshot shows a software interface for alert management. At the top, there are buttons for 'アラート一覧' (Alert List), 'スタート' (Start), and '実行' (Execute). Below these is a red bar with a '削除' (Delete) button. The main area is a table with the following data:

削除	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"/>	MFEND	2015/02/17	AUTO02	異常終了

アラートからの実績情報の表示

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	0099
異常終了ステップ	
OS-JOB 番号	UNIX

【Enterprise Server の実行状態】

Enterprise Server のスプールを表示すると、A-AUTO で実行指示された JCL が、プログラムのリターンコードを COND CODE で受け取っていることが確認できます。

<input type="checkbox"/>	<u>MF102</u>	JOB01040	A	JESUSER 00010	2015/02/17 10:38:35.04
<input type="checkbox"/>	<u>MF202</u>	JOB01041	A	JESUSER 00010	2015/02/17 10:38:43.92
<input type="checkbox"/>	<u>MFEND</u>	JOB01042	A	JESUSER 00010	2015/02/17 10:38:56.49
<input type="checkbox"/>	<u>MFENDAB</u>	JOB01043	A	JESUSER 00099	2015/02/17 10:39:01.61
<input type="checkbox"/>			Display		Delete

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

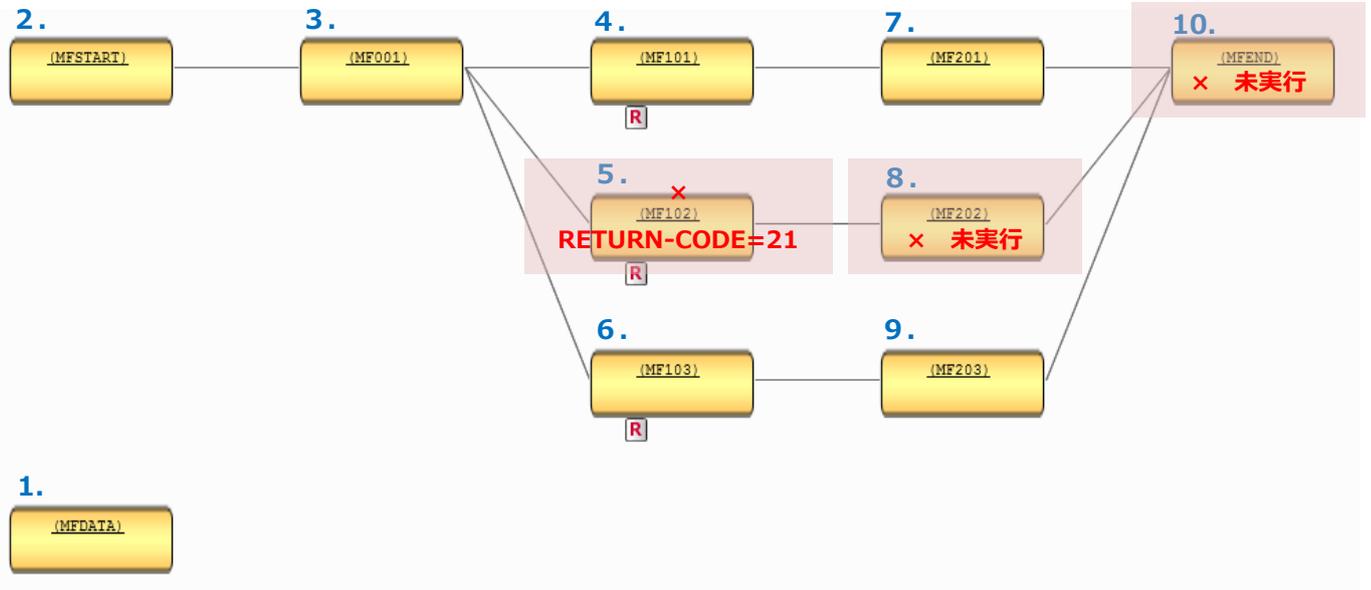
MFENDAB → MFEND リターンコードが 0 より大きい場合に実行

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) 先行ジョブ番号 分岐設定 <table border="1" style="width: 100%;"> <tr> <td>1</td> <td>010(MF000000) ▼</td> <td>RC>0</td> </tr> <tr> <td>2</td> <td>▼</td> <td></td> </tr> <tr> <td>3</td> <td>▼</td> <td></td> </tr> <tr> <td>4</td> <td>▼</td> <td></td> </tr> </table>	1	010(MF000000) ▼	RC>0	2	▼		3	▼		4	▼	
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" value="MFENDAB"/>													

5.3 未計画の異常終了時

【ネットワークの設定】

下記のように一部のプログラムからのリターンコードを変更します。



- 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

【ネットワークの初期状態】

MFDATA は終了していますので、前回終了時と変化はありません。

【Enterprise Server の初期状態】

前回終了時と変化はありません。

【一連ネットワークの実行】

変更プログラムが影響する、MF102、MF202、MFEND を再度実行します。

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)

【ネットワークの実行状態】

ネットワークの MF102 は未計画のリターンコード 21 を返却している設定している異常終了となっており、後続ネットワークは継続されません。時刻を確認すると後続ネットワークの MF202、MFEND は前回終了時のままです。

	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"/>	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	

異常終了時のアラート一覧



削除	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

アラートからの実績情報の表示

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

【Enterprise Server の実行状態】

Enterprise Server のスプールを表示すると、A-AUTO で実行指示された JCL が、プログラムのリターンコードを COND CODE で受け取っていることが確認できます。



<input type="checkbox"/>	MF102	JOB01044	A JESUSER 00021	2015/02/17 11:23:43.03
<input type="checkbox"/>	Display		Delete	

また、MF102 が計画外の異常終了になったため、後続ジョブである MF202、MFEND が実行されていないことが確認できます。

6. 結論

以上の結果から OPEN 環境における、Micro Focus Enterprise Sever と A-AUTO の連携には特筆する問題点は存在しませんでした。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 からシェルまたはスクリプトをキックし、Enterprise Server へ JCL の実行が可能であること
2. A-AUTO へ設定したジョブパラメータが正常にシェルまたはスクリプトへ引き渡され、展開が可能であること
3. A-AUTO へ先行ネットワーク、先行ジョブを設定し、リターンコードによるジョブ分岐の設定と実行が可能であること
4. A-AUTO へネットワーク並行処理の設定と実行が可能であること
5. Web 画面にて、A-AUTO、Enterprise Server 実行状況が確認可能であること
6. MFBSIJCL コマンドを利用してジョブを実行することにより、JCL の COND CODE を A-AUTO へ返却可能であること
7. MFBSI 設定ファイル値設定により、ジョブログが A-AUTO 側へ表示可能であること
8. COBOL プログラムで出力している日本語が A-AUTO へ正常に表示可能であること
9. MFBSIJCL コマンドにコンバートコマンドを組み合わせることにより、UTF 8 に変換されたジョブログが A-AUTO で確認可能であること
10. Linux や Windows で実行するジョブが別 OS 上で稼働していることを意識することなく Windows 上で運用でき、結果確認も同一管理画面から確認可能であること
11. Enterprise Server と A-AUTO で管理しているジョブ実行結果に相違がないこと

以上で連携動作検証を終了いたします。