

# スクリプトデバッグモジュール

## ドライラン (dry) 機能仕様書

富士通株式会社  
2011 年 3 月 1 日

## 更新履歴

日付	版数	担当	備考
2011/3/1	1.0 版	FST)	

## まえがき

本書は、ドライラン（dry）に求められる機能要件まとめたものです。

また、本書はプロト版の為、機能・性能改善するのに当たり、予告なしに変更する場合があります。

➤ 未実装、仕様未確定

機能が未実装であったり、仕様が未確定の部分は、本文中で網駆け（機能）表記しています。  
実装や性能改善するのに当たり、予告なしに変更する場合があります。

➤ 表記上規則

記号	意味
{ <u>ABC</u>   EFG}	{ }内の文字列の 1 つを選択することを示します。省略した場合、“_”(アンダーライン)の文字列が選択されたことを示します。
[ABC]	[ ]で囲まれた文字列は省略できることを示します。

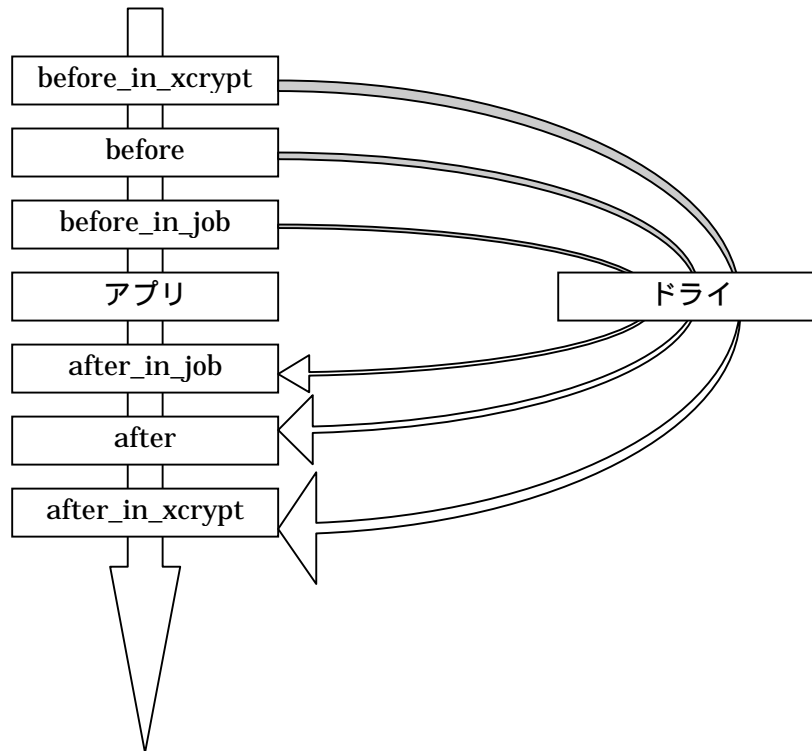
## 目次

第1章 機能要件.....	1
1.1. 利用要件.....	2
第2章 機能.....	3
2.1. 実行レベル.....	3
2.2. 実行環境.....	3
2.3. 実行アプリ.....	3
第3章 ドライコマンド.....	4
3.1. ドライ定義コマンド.....	5
3.1.1. 書式.....	5
3.1.2. パラメタ.....	5
3.2. ジョブリファレンス定義.....	7
3.2.1. 書式.....	7
3.2.2. パラメタ.....	7
3.2.3. 記述例.....	7

## 第1章 機能要件

既存にもドライラン機能はあるが、現在の xcrypt に対応していない為、期待通りのドライランができない。xcrypt の機能に合ったドライランにしたい。

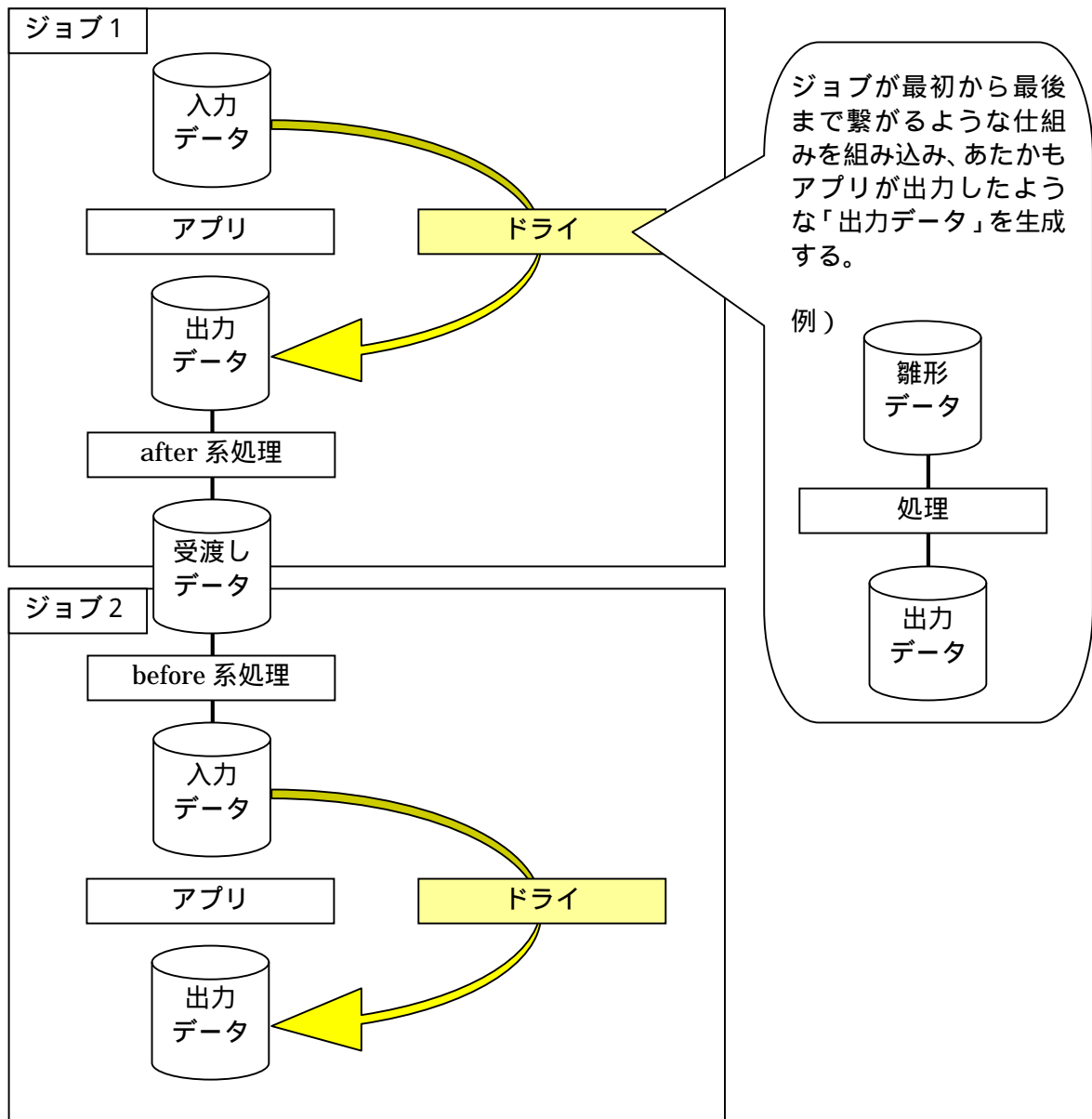
また、ユーザの用途に合う様、複数のドライランモードを用意したい。



## 1.1. 利用要件

これまでのドライランは、単純にアプリを実行しないだけの機能であった為、後続処理に対してファイルを渡す(before や after でアプリが扱うデータを生成したり抽出したりする) ようなジョブのドライランの実施は困難であった。ファイルの受渡しがあるようなジョブであっても処理できるよう、何らかの仕組みを用意したい。

動作イメージは次のようになる。



## 第2章 機能

1章で記述した要件を満たすために、実現する機能について述べる。

### 2.1. 実行レベル

利用シーンに応じ、実行レベルを切替えて実行できるようにする。

- テンプレートの記述チェックがしたい
- before\_in\_xcrypt や after\_in\_xcrypt の動作確認がしたい
- before や after の動作確認がしたい
- before\_in\_job や after\_in\_job の動作確認がしたい

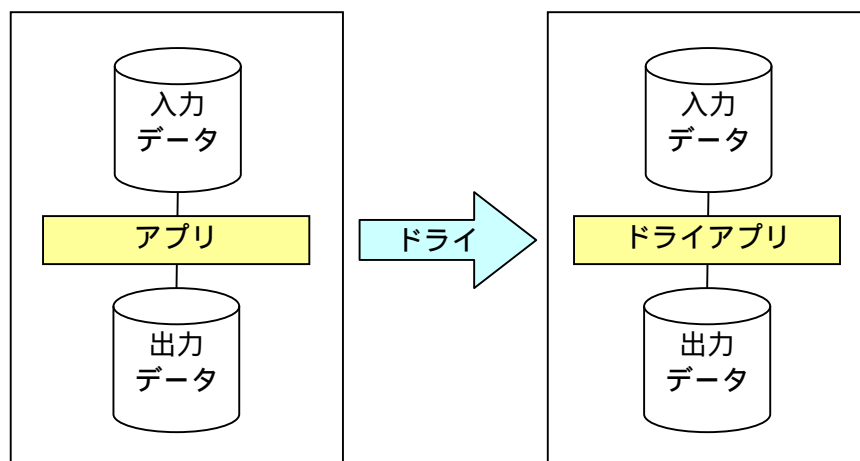
### 2.2. 実行環境

ドライランの実行レベルによっては、バッチサーバへの qsub(ジョブ投入)が不要となる。課金対象ユーザやバッチサーバを利用している他ユーザのことを考慮し、ローカルサーバ(管理サーバ)での実行をユーザが選択できるようにする。

- バッチサーバで実行
- ローカルサーバ(管理サーバ)で実行

### 2.3. 実行アプリ

アプリ(ジョブリファレンス定義の'exe'に記述したアプリ)は実行しない。アプリを実行しないと出力データが作成されず、後続処理が正常に動作しなくなる可能性が高い為、アプリが実行されるべきタイミングにユーザが指定したドライアプリ(アプリまたはコマンドや関数)を実行させることで、ユーザが期待する出力データを自由に作成できるようにする。



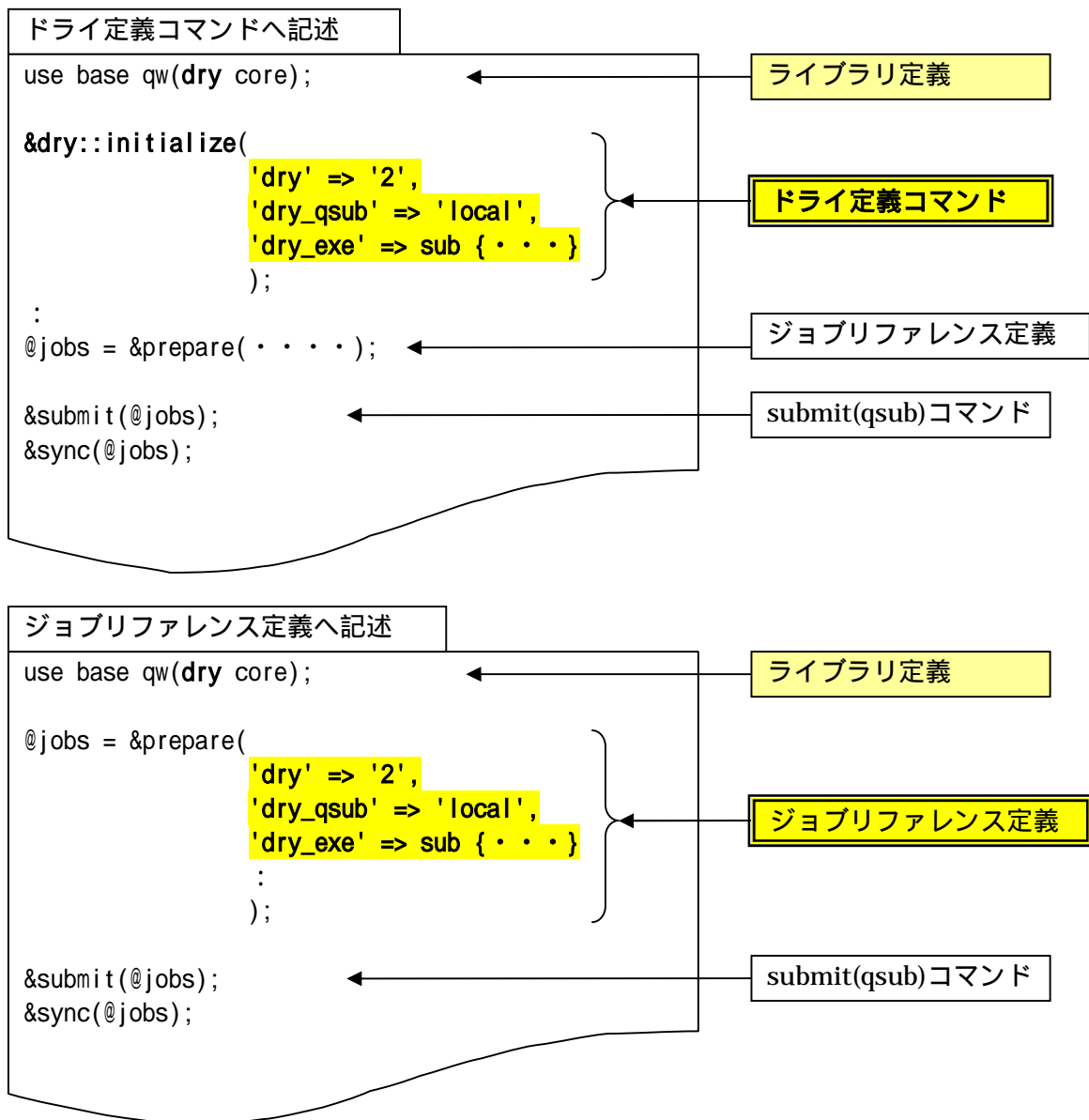
### 第3章 ドライコマンド

ドライランの定義と指示は、ドライライブラリ（ドライモジュール）のドライ定義コマンド、又はジョブリファレンス定義にて記述する。ドライコマンドは、スクリプトの所定場所に記述する。

< 定義 >

- ドライ定義コマンド
- ジョブリファレンス定義

両方に記述した場合、そのジョブはジョブリファレンス定義の指示に従い動作する。





### 3.1. ドライ定義コマンド

#### 3.1.1. 書式

```
&dry::initialize(
    ['dry' => ['ドライレベル']][,]
    ['dry_qsub' => ['ドライ実行環境']][,]
    ['dry_exe' => ドライアプリ[,]
);
```

#### 3.1.2. パラメタ

- ドライレベル  
ドライランの実行レベル。  
0 ~ 3 を指定。(省略値 = 0 : 通常動作)

< ドライランの挙動 >

	dry=0	dry=1	dry=2	dry=3
before_in_xcrypt 動作				×
before 動作				×
before_in_job 動作			×	×
アプリ動作		×	×	×
after_in_job 動作			×	×
after 動作				×
after_in_xcrypt 動作				×

: 実行する、× : 実行しない

- ドライ実行環境  
ドライランの実行環境。  
local または host を指定。(省略値 = host)

< 実行環境 >

local	実行形式を sh に切り替えて管理サーバで実行
host	バッチサーバの指定通りに実行

➤ ドライアプリ

アプリの代わりに実行する処理（アプリやコマンド、又は関数）。

例 1 ) アプリやコマンドを実行

```
&dry::initialize(  
  :  
  'dry_exe' => 'アプリやコマンド',  
  :  
);
```

例 2 ) 関数を実行

```
&dry::initialize(  
  :  
  'dry_exe' => sub {処理・・・},  
  :  
);
```

## 3.2. ジョブリファレンス定義

### 3.2.1. 書式

```
@jobs = &prepare(
    :
    ['dry' => ['ドライレベル']][,]
    ['dry_qsub' => ['ドライ実行環境']][,]
    ['dry_exe' => ドライアプリ[,]
    :
);
```

### 3.2.2. パラメタ

「4.1.2 パラメタ」(ドライ定義コマンドのパラメタ説明)を参照。

### 3.2.3. 記述例

```
use base qw(dry core);

&dry::initialize(
    'dry' => '1',
    'dry_qsub' => 'local',
);

@jobs1 = &prepare(
    'dry_qsub' => 'host',
    'dry_exe' => sub {処理 1}
    :
);

@jobs2 = &prepare(
    'dry' => '2',
    'dry_exe' => sub {処理 2}
    :
);

&submit(@jobs);
```

@jobs1 は下記の設定で動作

- dry='1'
- dry\_qsub='host'
- dry\_exe=sub {処理 1}

@jobs2 は下記の設定で動作

- dry='2'
- dry\_qsub='local'
- dry\_exe=sub {処理 2}