

エンジンのバージョン管理の方法を以下のとおり 規定する。

A .管理対象ファイル規定

1.バージョン管理するファイル

以下のファイル、フォルダについて、バージョン管理を行う。
javadoc や classes など、既存のリソースより作成できるものは、含まない。
ただし、エンジンとして提供する lib 以下の jar ファイルは、そのものをコピーしておく。

```
%APPS%\%java%\tomcat%\conf%\server.xml
%UAP%\%webapps%\dbdef%\src
%UAP%\%webapps%\dbdef%\jsp
%UAP%\%webapps%\dbdef%\WEB-INF%\lib
%UAP%\%webapps%\dbdef%\WEB-INF%\mis-taglib.tld
%UAP%\%webapps%\dbdef%\WEB-INF%\web.xml
```

2.バージョンアップ時に変更する、必須ファイル

バージョンアップ時には、以下のファイルをすべて更新する。
よって、開発者 (エンジン利用者)は、以下のファイルの変更を禁止する。

```
%UAP%\%webapps%\dbdef%\jsp%\common
%UAP%\%webapps%\dbdef%\WEB-INF%\lib%\hayabusa****.jar
%UAP%\%webapps%\dbdef%\WEB-INF%\mis-taglib.tld
%UAP%\%webapps%\dbdef%\WEB-INF%\web.xml
```

3.バージョンアップ時に個別に変更する必要がある可能性があるファイル。

基本的には、システム固有のファイルなので、システム毎にセッティングしてもよいファイルであるが、エンジンと密接な関係があるため、修正が必要になる場合がある。
そのようなケースを想定して、カスタマイズ時には、変更個所が判るように注意して修正しておいてください。

```
%UAP%\%webapps%\dbdef%\jsp%\index.jsp ~ index6.jsp
%UAP%\%webapps%\dbdef%\jsp%\menu.jsp
%UAP%\%webapps%\dbdef%\jsp%\custom
%UAP%\%webapps%\dbdef%\jsp%\debug
%UAP%\%webapps%\dbdef%\jsp%\image
%UAP%\%webapps%\dbdef%\jsp%\menu
%UAP%\%webapps%\dbdef%\jsp%\treebom
```

APPS とUAP は、init.bat 中で、設定します。

通常は、

APPS H:ドライブ

UAP G:ドライブ

となります。

B.バージョン管理規定

1. リリース形態

バージョン管理は、3つのリリース形態を持つこととする。

Release Builds	正式リリース版 本番環境では、このリリースをご使用ください。
Milestone Builds	新機能評価やバグFIXの確認等にご使用ください。
Nightly Builds	ほぼ、毎日更新されます。(Use at your own risk!)

また、それぞれ管理ファイル、時期が異なります。

Release Builds	バージョン管理対象の全てのファイルを復元可能とします。
Milestone Builds	バージョンアップ時の必須ファイルのみ、保存します。
Nightly Builds	最新バージョンのみ、保存します。(履歴管理しません。)

2. バグ修正バージョン/パッチファイル

バグ修正バージョンや、パッチファイルの更新等は、行いません。

必ず、Release Builds として、正式リリースします。

また、本番環境では、正式リリースにて運用ください。

Milestone Builds を元にした改造/改善は、お受けできません。

3. バージョン管理番号

バージョン管理は、4個のバージョン番号で表します。

1桁目	アプリケーション上位互換
2桁目	外部インターフェース上位互換
3桁目	内部インターフェース上位互換
4桁目	バグフィックス、内部リファクタリング、軽微な変更

1桁目 (アプリケーション上位互換)

この桁が同一バージョン間では、アプリケーション互換が保証されます。

同一バージョンでは、jsp/common, hayabusa.jar, mis-taglib.tld, web.xml の置換え (上書きコピー) のみで、アップデートできます。

2桁目 (外部インターフェース上位互換)

この桁が同一バージョン間では、外部インターフェース互換が保証されます。

同一バージョンでは、hayabusa.jar, mis-taglib.tld の置換え (上書きコピー) のみで、アップデートできます。

3桁目 (内部インターフェース上位互換)

この桁が同一バージョン間では、内部インターフェース互換が保証されます。

同一バージョンでは、hayabusa.jar の置換え (上書きコピー) のみで、アップデートできます。

4桁目 (バグフィックス等)

この桁が同一のバージョン違いは、Release Builds には存在しません。

この桁が異なる場合は、(3桁目が同一であれば) hayabusa.jar の置換え (上書きコピー) のみで、アップデートできます。

4. バージョンの確認方法

個々のファイルを総合してバージョン管理しているため、配布用 (アーカイブ用) ファイルにバージョン番号を持たせ、各ファイルには、直接持たせません。

アーカイブを展開したファイルのREADME.TXT を確認してください。

なお、管理ツールより バージョン情報も表示できます。