



# Enterprise Edition 2016 Backward Compatibility Upgrade: QAD Enterprise Financials 2017

**March 2017**

**Note** This document is accurate as of March 23, 2017. To ensure you have the latest version, check the QAD Document Library at [documentlibrary.qad.com](http://documentlibrary.qad.com), or go directly to the upgrade instructions by clicking:

[Enterprise Edition 2016 Backward Compatibility Upgrade: QAD Enterprise Financials 2017](#)

**Overview** 2

**Prerequisites** 2

**Upgrade Enterprise Financials** 3

## Overview

This document provides the steps required to upgrade QAD Enterprise Financials to a later QAD Enterprise Financials release (also known as a Backward Compatibility installation). Backward Compatibility refers to the upgrade of a module to a higher module release without upgrading the rest of the Enterprise Edition core release. For example, Enterprise Financials can be upgraded independently from Enterprise Edition Core. This capability allows you to implement an Enterprise Financials 2017 instance in an Enterprise Edition 2016 implementation.

The backward compatibility upgrade patch is installed using YAB. For detailed information about YAB installation prerequisites, installation, and use, see [QAD Enterprise Edition Installation Guide](#) for Enterprise Edition 2016.

## Prerequisites

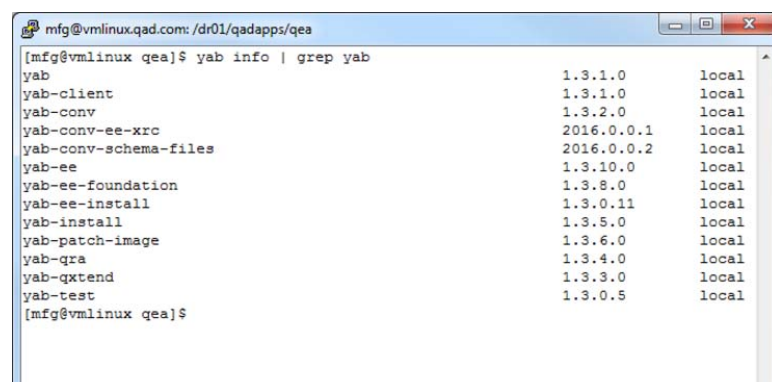
### YAB Requirements

The QAD Enterprise Edition environment in which you are executing the backward compatibility upgrade must be using the latest YAB patch. To check the YAB and YAB patch versions that are installed in your environment, enter:

```
> yab info | grep yab
```

You will see information similar to the following.

**Fig. 1**  
YAB and YAB Patch Versions



```
mfg@vmlinux.qad.com: /dr01/qadapps/qea
[mfg@vmlinux qea]$ yab info | grep yab
yab 1.3.1.0 local
yab-client 1.3.1.0 local
yab-conv 1.3.2.0 local
yab-conv-ee-xrc 2016.0.0.1 local
yab-conv-schema-files 2016.0.0.2 local
yab-ee 1.3.10.0 local
yab-ee-foundation 1.3.8.0 local
yab-ee-install 1.3.0.11 local
yab-install 1.3.5.0 local
yab-patch-image 1.3.6.0 local
yab-qra 1.3.4.0 local
yab-qxtend 1.3.3.0 local
yab-test 1.3.0.5 local
[mfg@vmlinux qea]$
```

Figure 1 shows a system installed with YAB 1.3 and YAB patch 1.3.6.0, called `yab-patch-image`. Ensure that the YAB patch version in your environment is at least equal to the minimum requirement listed in Table 1. Always use the latest patch available for your environment.

**Table 1**  
Minimum YAB Patch Versions

YAB Version	Minimum Patch Version
YAB 1.3	yab-patch-1.3.6.0
YAB 1.4	yab-patch-1.4.3.0

For more information on installing a YAB patch, see the [QAD Enterprise Edition Administration and Configuration Guide](#).

## Progress Requirements

Progress requires that `after-imaging` and `2phase` be disabled for the index rebuild process to complete properly. You must disable these features in your environment before you start the installation process.

## Back Up the Existing System

Back up your production databases before you start the installation process. Previously installed hot fixes can interfere with this patch installation. Ensure that the following folders are backed up and their contents deleted.

- `dist/fin/patch`
- `dist/fin/proxypatch`
- `patches/fin/proxypatch`

For more information on backup procedures, see the *QAD Enterprise Edition Administration and Configuration Guide*.

## Factory Defaults

The QAD Enterprise Financials patch overwrites any changes you have made to the factory defaults. If you want to keep your changes, back them up.

To save your current storage search and report variant settings, run the Stored Search Export Initial Settings and Report Variant Export Initial Settings programs from the QAD .NET UI. The programs generate two files, `factorydefaults.xml` and `reportvariantfactorydefaults.xml`. Save these two files in the `DefaultDataDirectory` as defined in the `server.xml` file. The default location of this directory is `QAD_EE_install_directory>/config/data`. For example:

```
<defaultdata>
  <DefaultDataDirectory> /dr01/qadapps/config/data/<DefaultDataDirectory>
</defaultdata>
```

Back up these files by saving them in a different location.

## Upgrade Enterprise Financials

- 1 Download the following packages from the [QAD Store Download Center](#). Always use the latest version available for your environment.

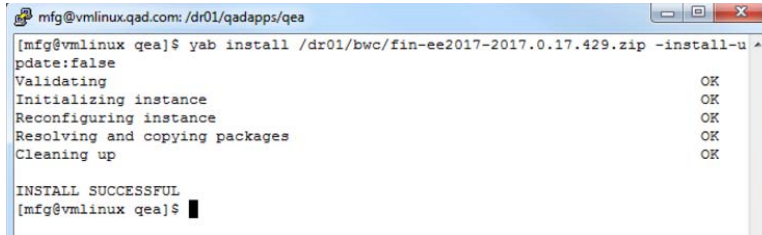
**Table 2**  
Required Packages

Name	Package Name	Minimum Version
2017 Enterprise Financials	qad-ee-fin-patch	2017.0.4.0
2017 Financials Schema	fin-ee2017	2017.0.17.429
2017 Financials Conversions	yab-conv-ee-xrc	2017.0.0.1

- 2 Install the Enterprise Financials Schema package. Ensure that you do not apply the update by including `-install-update:false`, as shown in the following example

```
> yab install /<download-dir>/fin-ee2017-2017.0.x.x.zip -install-update:false
```

**Fig. 2**  
Enterprise Financials Schema Package



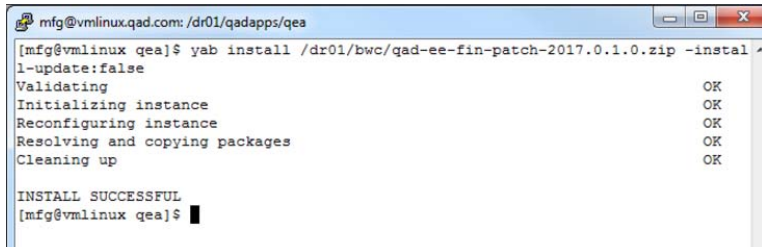
```
mfg@vmlinux.qad.com: /dr01/qadapps/qea
[mfg@vmlinux qea]$ yab install /dr01/bwc/fin-ee2017-2017.0.17.429.zip -install-update: false
pdate:false
Validating OK
Initializing instance OK
Reconfiguring instance OK
Resolving and copying packages OK
Cleaning up OK

INSTALL SUCCESSFUL
[mfg@vmlinux qea]$
```

3 Install the Enterprise Financials package, again ensuring that you do not apply the update.

> yab install /<download-dir>/qad-ee-fin-patch-2017.0.x.x.zip -install-update: false

**Fig. 3**  
Enterprise Financials Package



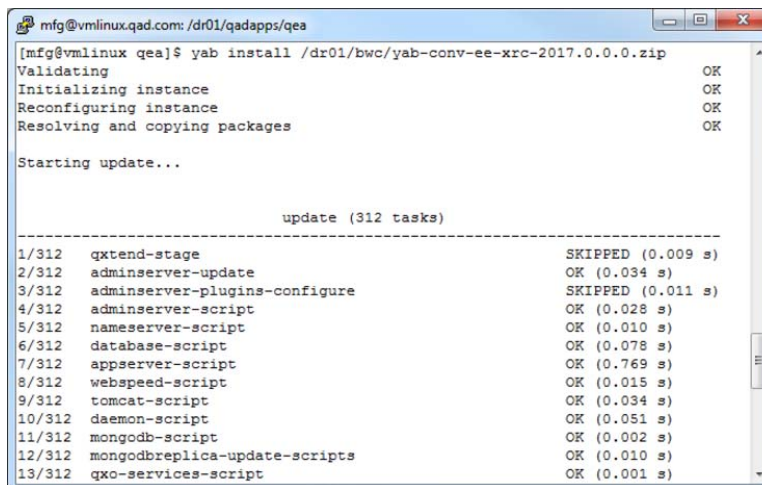
```
mfg@vmlinux.qad.com: /dr01/qadapps/qea
[mfg@vmlinux qea]$ yab install /dr01/bwc/qad-ee-fin-patch-2017.0.1.0.zip -install-update: false
l-update: false
Validating OK
Initializing instance OK
Reconfiguring instance OK
Resolving and copying packages OK
Cleaning up OK

INSTALL SUCCESSFUL
[mfg@vmlinux qea]$
```

4 Install the Financials Conversions package.

> yab install /<download-dir>/yab-conv-ee-xrc-2017.0.x.x.zip -install-update: false

**Fig. 4**  
Enterprise Financials Conversions



```
mfg@vmlinux.qad.com: /dr01/qadapps/qea
[mfg@vmlinux qea]$ yab install /dr01/bwc/yab-conv-ee-xrc-2017.0.0.0.zip
Validating OK
Initializing instance OK
Reconfiguring instance OK
Resolving and copying packages OK

Starting update...

update (312 tasks)
-----
1/312 qxtend-stage SKIPPED (0.009 s)
2/312 adminserver-update OK (0.034 s)
3/312 adminserver-plugins-configure SKIPPED (0.011 s)
4/312 adminserver-script OK (0.028 s)
5/312 nameserver-script OK (0.010 s)
6/312 database-script OK (0.078 s)
7/312 appserver-script OK (0.769 s)
8/312 webspeed-script OK (0.015 s)
9/312 tomcat-script OK (0.034 s)
10/312 daemon-script OK (0.051 s)
11/312 mongodb-script OK (0.002 s)
12/312 mongodbreplica-update-scripts OK (0.010 s)
13/312 qxo-services-script OK (0.001 s)
```

5 Confirm that the package versions are correctly updated. Enter the following command:

> yab info

**Fig. 5**  
YAB info results

```
mfg@vmlinux.qad.com: /dr01/qadapps/qea
[mfg@vmlinux qea]$ yab info

INSTANCE

/dr01/qadapps/qea

MODULES

archiving                               1.0.0.78      local
assistance-help-ee                     2016.0.0.2   local
base-api                               1.1.0.79     local
dde-ant                                2.2.0.1      local
dde-build                              2.2.8.0      local
dde-core                               2.2.0.14     local
dde-registry                           2.2.1.0      local
demo-data-ee2016                       2016.0.16.209 local
fin-apidoc                             2017.0.80.1  local
fin-bin64-proxy                        2017.0.80.1  local
fin-bin64-qadfin                       2017.0.80.1  local
fin-bldata                             2017.0.80.1  local
fin-customization                      2017.0.80.1  local
fin-data                               2017.0.80.0  local
fin-ee2017                             2017.0.17.429 local
fin-src-proxy                          2017.0.80.1  local
```

**6** Update the environment.

> yab update

**Fig. 6**  
YAB Update

```
mfg@vmlinux.qad.com: /dr01/qadapps/qea
293/312 copy-electronic-signature-update      OK (0.127 s)
294/312 daemon-configure                      OK (0.042 s)
295/312 webapp-fin-api-update                 OK (0.268 s)
296/312 webapp-fin-api-configuration          OK (0.002 s)
297/312 chui-client-us-update                 OK (0.004 s)
298/312 aim-client-us-update                  OK (0.006 s)
299/312 chui-client-ch-update                 OK (0.007 s)
300/312 aim-client-ch-update                  OK (0.005 s)
301/312 copy-mfg-configs-update                OK (0.016 s)
302/312 copy-mfg-auditing-progress-configs-update OK (0.036 s)
303/312 copy-mfg-auditing-qad-configs-update  OK (0.080 s)
304/312 report-archiving-update               OK (0.021 s)
305/312 report-cmr-update                     OK (0.028 s)
306/312 report-fin-update                     OK (45.218 s)
307/312 report-mrc-update                     OK (0.024 s)
308/312 report-periodic-costing-update        OK (0.010 s)
309/312 metadata-qracore-update               SKIPPED (0.393 s)
310/312 metadata-fincore-update               SKIPPED (0.033 s)
311/312 metadata-mfgcoreplus-update           SKIPPED (0.039 s)
312/312 qxtend-application-id-update          OK (0.055 s)

-----
BUILD SUCCESSFUL (47:07 m)
[mfg@vmlinux qea]$
```

Ensure that the build completes with a BUILD SUCCESSFUL message. If the build fails, review the build/logs/yab.log file for errors.

**7** Execute the Financials conversion.

> yab -conv.fin.version:93o conversion-fin-upgrade

**Fig. 7**  
Financials Conversion

```
mfg@vmlinux.qad.com: /dr01/qadapps/qea
[mfg@vmlinux qea]$ yab -conv.fin.version:93o conversion-fin-upgrade

-----
conversion-fin-upgrade (2 tasks)
-----
1/2 conversion-stage                        OK (11.741 s)
2/2 conversion-fin-run                      OK (1.066 s)
-----

BUILD SUCCESSFUL (13.298 s)
[mfg@vmlinux qea]$
```

- 8 Validate the conversion execution. You can monitor the conversion's progress in the `build/logs/yab.log` and `build/logs/conversion.log` files.

Once the conversion is complete, perform the following verifications.

- a Ensure that the `conversion-fin-upgrade` process completed successfully. The output should be similar to the following.

**Note** The timing will vary depending on the size of the database and server specification.

**Fig. 8**

Conversion-fin-upgrade Successful

```
> yab conversion-fin-upgrade

-----
conversion-fin-upgrade (2 tasks)
-----
1/2 conversion-stage                OK (6.184 s)
2/2 conversion-fin-run              OK (1.397 s)
-----

BUILD SUCCESSFUL (9.284 s)
```

- b Review each process associated with the `conversion-fin-upgrade` process in the `yab.log` file. You can use `"BuildContext -"` to find each process execution entry. For example, to locate the log content associated with running the financials database conversions step, search the `yab.log` file for:
 

```
"BuildContext - conversion-fin-run"
```
- c Review each conversion process in its corresponding `conversion.log` file. You should search for the items in the following table to ensure that all major areas of the conversion were executed.

**Table 3**

Conversion review items

Conversion.log search items	Description
"Target Version Conversion: Starting"	A sample check to ensure that the qaddb conversions were started.
"Processing Table: YearClosing"	A sample check to ensure that the Financial conversions calling program was executed.
"[Post Upgrade Conversion Updates] Conversion completed (post_upgrade_cv93g.p)."	A sample check to ensure that the last Financial conversion was executed.
"Target Version Conversion: Complete"	A sample check to ensure that the qaddb conversions were completed.

- 9 Execute the Financials synchronization

```
> yab fin-sync-run
```

**Fig. 9**

Financials Synchronization

```
mfg@vmlinux.qad.com: /dr01/qadapps/qea
[mfg@vmlinux qea]$ yab fin-sync-run

-----
fin-sync-run (1 task)
-----
1/1 fin-sync-run                OK (1:49 m)
-----

BUILD SUCCESSFUL (1:49 m)
[mfg@vmlinux qea]$
```

- 10 Apply the non-intrusive customization code to the new customization templates that were shipped with the patch. Then recompile the code.

For more information on customizations, see the *QAD Enterprise Edition Administration and Configuration Guide*.

