Below are some frequently asked questions about using Financial Management in a 64-bit environment:

- What is 64-bit Financial Management?

This is the 64-bit port of Financial Management. It is functionally identical to 32-bit Financial Management and uses the same code base. The first version of 64-bit Financial Management is 11.1.1.x.

- Which Operating Systems are supported?

The x86-64 versions of Microsoft Windows are supported. On the application server, this includes Windows 2003 and subsequent service packs. On the client side, the x86-64 versions of Windows XP and Vista are supported

Which CPUs are supported?
The x86-64 architecture is supported. This includes the AMD64 and Intel 64 (formerly EM64T) processors. The Itanium processor (IA-64 architecture) is currently NOT supported.

- How does one migrate from 32-bit to 64-bit Financial Management? Will 64-bit Financial

Management work with an application created under 32-bit Financial Management?
The schema upgrade path from a previous release running 32-bit to a release running 64-bit Financial Management is similar to an ordinary version upgrade. When upgrading Financial Management from a previous release, use the Schema Upgrade Utility to upgrade the database schema to support the new version of Financial Management (11.1.1.x or newer); schema upgrade is not required when moving from 32-bit Financial Management 11.1.1.x or newer to a 64-bit version of the same release.
NOTE: To install the 64-bit version of Financial Management software, you must select "New Installation" in EPM System Installer.

- Which components of the system need to be 64-bit? In particular, does the relational

database need to be 64-bit?
The database can be either 32-bit or 64-bit as long as it is a supported type and version. See the EPM System Certification Matrix for supported database.

- What are the benefits of 64-bit Financial Management?

The main benefit of 64-bit Financial Management is the ability to hold substantially more data in memory at one time. Depending on the size of the application and its usage profile, the extra memory can lead to significant speed improvements, while simultaneously reducing the load on the relational database.

- What are the memory limitations of 64-bit Financial Management?

The limit of virtual address space is dictated by Microsoft Windows and is 8 TB (8192 GB), compared to a maximum of 3 GB in 32-bit Windows. Since the physical memory in almost any current computer is far smaller than 8 TB, the implication is that, in practical terms, 64-bit Financial Management is limited by physical memory, rather than virtual memory. In other words, 64-bit Financial Management can take advantage of all available physical memory once the proper memory parameter adjustments are made (see below).

- Are there any memory settings that need to be tuned for 64-bit Financial Management?

Yes. Financial Management's default memory settings are appropriate for a small to medium size application in a 32-bit environment. To take advantage of the extra memory in a 64-bit environment, we recommend the following settings for a monthly application. The relevant registry settings are NumDataRecordsinRAM and MaxDataCacheSizeinMB which need to be created or changed in [HKEY_LOCAL_MACHINESOFTWAREHyperion Solutions Hyperion Financial ManagementServer] on each application server's Windows registry. The following table contains suggested values for these parameters depending on available memory. This is done with the assumption that Financial Management is the only memory-intensive process running on the machine and running only a single Financial Management application.
If multiple Financial Management applications will be active, then

divide the Total physical Memory installed on the server by the number of Financial Management applications to arrive at the “Available Physical Memory” for each application.
Available Physical Memory NumDataRecordsinRAM MaxDataCacheSizeinMB
4 GB 4,000,000 500
8 GB 10,000,000 1500
16 GB 30,000,000 4500
32 GB 60,000,000 9000
For a weekly application, divide the NumDataRecordsinRAM by 4, without changing the value in the last column.

- What kind of applications will see the most benefit?

Applications with large memory requirements will see the most benefit. This includes applications with one or more of the following characteristics:

1. Large scenarios (millions of records per year)
2. Dense applications (many large subcubes)
3. Large memory footprint as a result of many scenarios being accessed concurrently
4. Weekly applications

- What kind of applications will see the least benefit?

Small applications, where the total memory footprint of the application, even under load, can fit comfortably in the 32-bit memory space.