

# Managing Common Components DLL Guidelines

**Version:** 1.0

Adopted Date: 08/10/2009

**Background Information:** Currently there are several common components being developed or tested for other applications to consume. These common components help developers to reduce duplicated data as well as functionalities in their applications.

**Challenges:** Just like other software, common components are subject to change to meet the new business rules or technical requirements. Without appropriate version control, communication and release planning, the changes in common components can bring down other applications that are using them.

## Guidelines:

- All common components must follow the agency adopted [architecture review process](#).
- Common components should be built on the agency approved technologies. They should avoid requiring developers to install additional non-supported libraries.
- When common component are tied to the data source, it need to have a “configurable” parameter that allows the consumer to pick the database environment.
- In order for a common component to move to production, it needs to have adequate release notification, instruction to use the component, changes made to the new version and a plan for the old version.
- A common component is deemed as “production ready” only after at least 1 application successfully consumes it and makes it to production.
- When releasing common components for the developers to consume, unless noted as beta, they should be production version for all environments (dev, test, prod...etc)
- Initially developers will consume the common components by copying the DLL from the [Soft Center](#) to their application BIN directory. SAT will give approval for the DLL to register in GAC when the DLL is used widely by many applications. [Soft Center](#) is the central place to house the common components DLL and their documentation.
- When changes are made to a production common component, regardless of how small or large, its version number needs to be changed. Major changes with extra functionalities can use the whole number such as version 2.0. Minor changes such as bugs fixed can use the subset number (e.g. version 1.0.1.) New features will be added to the most current version only.
- Changes to the existing common components should not break the applications that are already consuming the existing one. In case this is unavoidable, the project lead needs to coordinate with SAT and Steering Committee (cross Programs) to work out the release and implementation plan.

## Document History

Date	Version	Editor	Change
08-05 <sup>th</sup> -2009	1 <sup>st</sup> Draft	Son Tran	First proposal
09-10 <sup>th</sup> -2009	1	SAT	SAT approved the guidelines