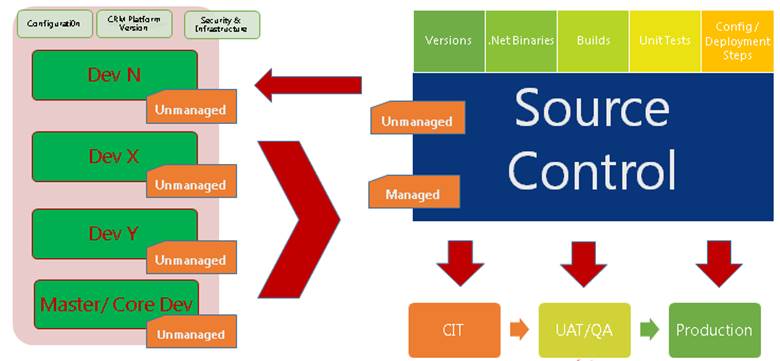
**Managed Vs Unmanaged Solutions**

* March 4, 2016
* Posted by: Raz Dynamics
* Category: Microsoft Dynamics CRM

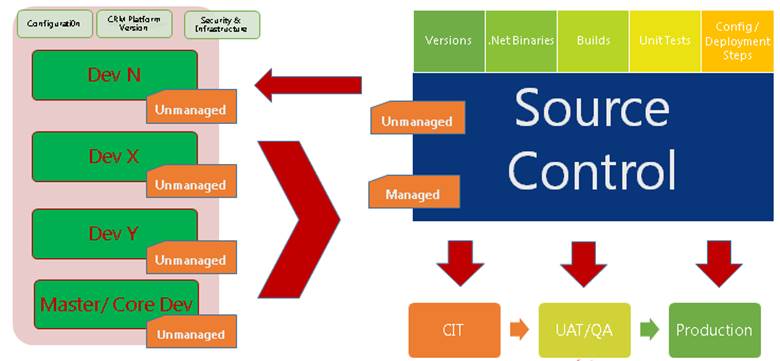
[1 Comment](http://www.crmconsultants.co.uk/managed-vs-unmanaged-solutions/#comments)



The Concept of Managed & Unmanaged Solutions was introduced with MS Dynamics CRM 2011, as a simpler way to package CRM customizations and deploy to multiple CRM environments. This would simplify the release process and enable Microsoft Partners to utilize XRM to deliver ISV Solutions for CRM. This allowed customers to easily install solutions from third party vendors allowing to easily import, update and maintain CRM customizations. Prior to this customization were migrated in the same state as they were in development and which sometimes led to customizations being overwritten or corrupted by the end user, effectively breaking the solution. Managed solution helps to preserve the ISV solution in the state that it was released in as well as the Intellectual property of the ISV to a certain degree.

The ALM for Microsoft CRM officially states that unmanaged solutions are used for development and Managed is released downstream to production. Please see the following reference for more information[www.microsoft.com/en-gb/download/details.aspx?id=39044](https://www.microsoft.com/en-gb/download/details.aspx?id=39044)

This scenario is Ideal for ISVs and works for incidental development or where development is following a fixed release calendar, but what about for agile scenarios where the customer is responsible for developing and releasing their own solutions and deployments? There can be some debate over whether only managed solutions are released to UAT and Production or if unmanaged solutions are more preferred by the customer for the following reasons;



**Unmanaged Solutions**

* All Solutions must begin as Unmanaged, allows the customer to directly control their intellectual property and customizations
* Unmanaged Solution customization are made at the Unmanaged Layer which is also part of the Default Solution
* Allows us to make customizations, register plugin steps etc., before we can export as Managed
* Deleting the Unmanaged Solution only deletes the reference to the solution, not the customization components
* Ensures customer business customizations & data are always part of the default solution
* Rolling back requires manually deleting the customization components from either the solution or Default Solution, or restoring the last Backup of the CRM
* Less risk of losing customizations and data as more difficult to remove
* Depends on how the solution is being delivered. i.e. internally or externally
* Risks associated to how well IT infrastructure / TFS / Version control is maintained
* If the business has procedures in place to disable direct customization to live / security
* Unmanaged Solutions are easier to maintain for customers who are responsible for their own deployment
* Allows to snapshot production and resolve a blocking issue due to the use of unmanaged solutions
* Merging behavior for customization updates is simpler, less complex than managed
* The unmanaged solution can be exported as unmanaged, but not vice versa

**Managed Solutions**

* Managed Solutions need to be exported as Managed from an Unmanaged Solution
* The whole point of Managed is locking down the Component states so they cannot be edited
* This secures the solution in the Target / Production so it keeps the solution feature working and prevents end users from breaking it. Therefore it is maintainable.
* Managed Solutions are installed at the Managed Layer
* Deleting the Managed Solution will remove all its customizations as well as data contained
* Solutions are additive, no build in feature to removing managed customization without losing your data unless a holding solution is used.
* Any Managed components that are customizable will be done at the Unmanaged Layer
* Managed Solutions become read only once deployed so they cannot be manipulated

·        Roll back / removing a Managed Solution after deploying to production is simple

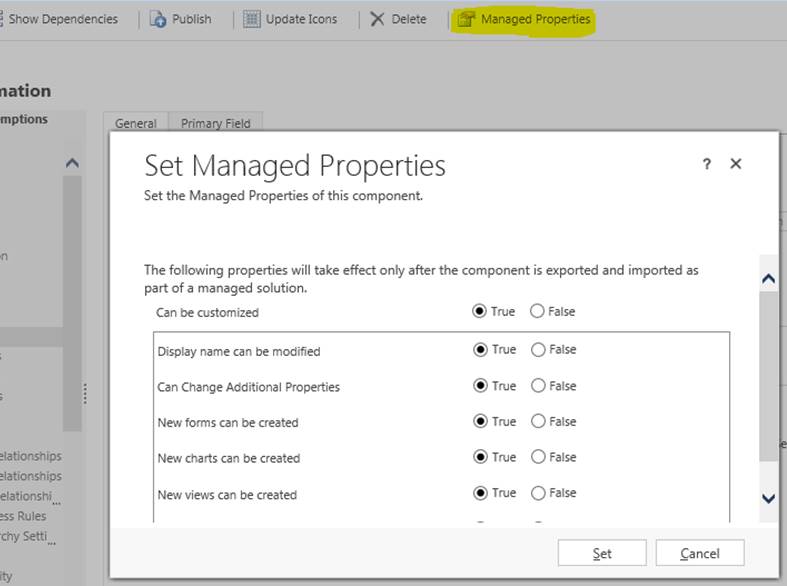
·        Preparing managed solutions for deployment requires more considerations for dependencies & merge behavior

·        Once you deploy as Managed in production you cannot change back to unmanaged once the solution has been deployed

Before your release a managed solution to Production, you should understand how the layers of the solution mechanism work and how they merge

* System Layer – The Default Solution and components that come shipped with all CRM instances, also referred to as Vanilla or OOTB (Out of the Box)
* Unmanaged layer – Unmanaged Solution components
* Manager Layer – Managed Components

You may find that System Components in an Unmanaged Solution have managed state by default, the Managed property would not become effective unless exported as a Managed Solution and keeps System or Business Entities as Customizable as these entities are used repeatedly for many other customizations projects, preventing Managed Solutions from owning the system component.  Managed Properties are useful to prevent clients from deleting or editing customizations and breaking core functionality after being installed on production. In the Unmanaged Solution we can set the managed properties for the Managed Solution Components as shown below;



**Rules for Updating Solutions**

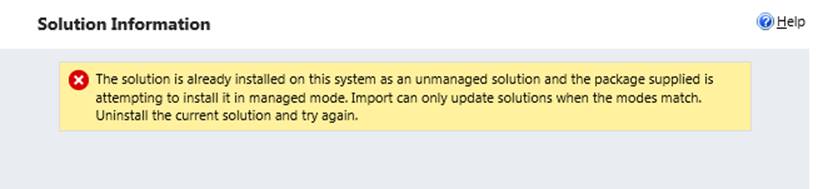
* A solution component needs to belong to either a Managed or Unmanaged Component, Cannot be a Mixture of both!
* Components in the unmanaged solution layer cannot be updated by a managed solution.
* Manually deleting the components that the unmanaged solution installed, you can import that solution again as managed and then install an update for it with a managed solution. However you will lose your data, will need to export the data prior to deletion and then re-import it when the components are installed as managed
* You cannot Directly Edit the Components within a managed solution. If the Managed properties for the solution components are set to allow customization, you can edit them in the Customization Areas (Default Solution) or from another managed Solution. This is useful during upgrades when you may require to make some changes to resolve an error.

When importing a Managed solution you will have the following options;

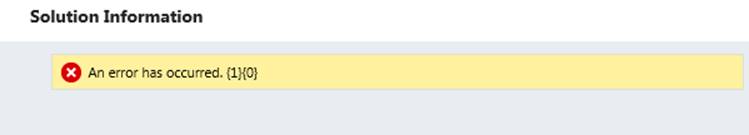
**Maintain customizations (recommended)**– This option maintains any unmanaged customizations performed on components, however some of the updates included in this solution may not take effect.

**Overwrite customizations**– Overwrites any unmanaged customizations previously performed on components included in this solution. All updates included in this solution will take effect. Using this option may become useful when investigating conflict issues to establish if the user has applied any conflicting changes to the solution. They should perform a backup export the existing unmanaged solutions before doing so to re-apply if required.

A Managed Solution cannot be installed Over an Unmanaged Version of the Same Solution



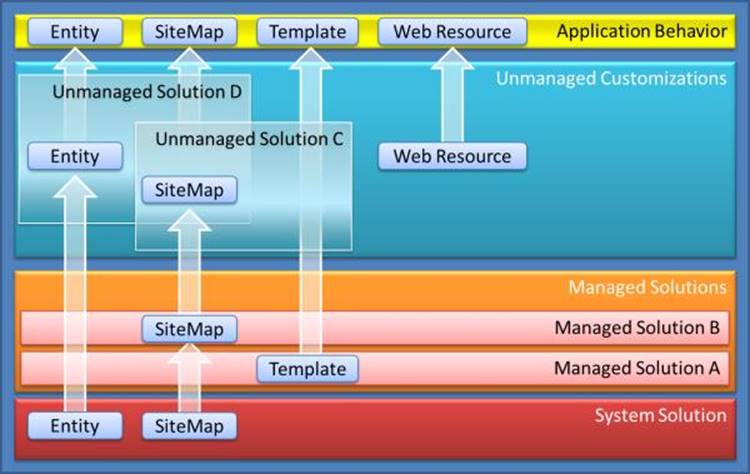
An Unmanaged version of the Solution cannot be installed over Managed Solution



Managed Solution cannot override or replace the existing Unmanaged Solutions. For Custom Fields of an Unmanaged Solution, these will become set to the state of Managed as soon as its exported as a managed solution

**Merging**

Merging allows combination of many Managed and Unmanaged solutions to co-exist in a CRM deployment, Virtually Merging the Behavior of the multiple solutions according to its rules. Please be aware The Installed On date for solution is when it was First Installed and Not Updated. Since Managed Solutions are stacked in the order of Solution Imports, the last installed managed solution update has the most effect the overall behavior of the winning customization and final Application behavior as shown in the following diagram;

[](http://www.crmconsultants.co.uk/wp-content/uploads/2016/03/dynamicscrmsolutionsmerging.jpg)

The solution publisher for a managed solution plays an important factor when releasing an update to your managed solution. Using the same solution publisher you can create a new managed solution to update a managed solution you previously released.

* Defining Publisher allows to track customizations and multiple development teams
* Helps prevent conflicting customizations
* Adds Prefix to Entity and Field Attributes
* Define Integer Range for Picklists/ OptionSet.

**Components in managed layers will be owned by the solution publisher.**

* Publisher owns the component, not the solution.
* Components with same name and publisher will be considered the same thing.
* Removing a solution does not remove a component when it is referenced by another solution using the same shared publisher.
* Default Publisher created for each environment with same default new\_prefix and Integer range for optionsets
* Important when creating development environments

“Creating a new solution with a different name and same publisher, will appear to work until the next version of the solution is deployed. Selecting the option to import without overwriting will import the new version of the solution within a layer directly above the previous version of the core solution. This will be beneath the solution containing the hotfix, which therefore may result in configuration and customization changes not being surfaced.”

* Solution must maintain the same solution publisher and the same solution name
* The Installed Date remains the Same when updating a solution, which can be misleading
* Updating the Solution does not cause loss of data
* Maintain the version number by overwriting the existing solution
* Using the recommended approach to preserve customizations when importing managed solutions, maintaining the version number will automatically replace any components within the solution being imported
* Lack of ability to identify that a hotfix has been applied. Consider adding a note
* Or with an incremented version number of the existing solution and create a new layer
* Incrementing the version number will create a new managed layer for the solution above the old layer.
* Can become misleading, as it suggests that the hotfix can be uninstalled or deleted which is not necessarily true due to the additive nature of the platform.
* Creating a new solution with a different name and same publisher, will appear to work until the next version of the solution is deployed. Selecting the option to import without overwriting will import the new version of the core solution within a layer directly above the previous version of the core solution. This will be beneath the solution containing the hotfix, which therefore may result in configuration and customization changes not being surfaced.
* Creating a New Solution with just the updates will allow you to rollback, but will add an additional solution into your organization, which could potentially cause clutter

Removing Managed Solution Components without losing data

As discussed above, since solutions are additive in nature we need a way of removing a managed solution component after has been deployed without losing our important data and customizations in the process, because uninstalling managed solutions will result in losing all the data stored in the attributes. However there is a technique to achieve this by using a ‘Holding’ solution.

Create a temporary ‘Holding’ managed solution for the customizations you want to keep using the same publisher, but with a Unique Name. This is so the new solution contains the customized components at a separate managed layer, so when you do delete the original managed solution containing the field you want to remove, you will not lose the data for the remaining objects.  You can use XRM toolbox will be able to copy all the components for you, eliminating the risk of missing any components. Now Import the holding solution into the target, and then delete the old solution which contained the fields you wanted to remove.

This deletes the unwanted managed components while the “holding” solution prevents the wanted customizations from being deleted. If you need to keep the original name of the Managed solution you just need to Import the new version of the managed solution and delete the “holding” managed solution this will allow you to keep your Managed customizations without losing the data and customizations you want to keep.

Understanding the mechanics behind Managed and Unmanaged solutions is a very important aspect to solution development and release management which is often overlooked, and Managed solution is  Whether you choose managed or unmanaged in your production should be based on how your organizations IT infrastructure is setup with Source Control and TFS as managed solutions will require a robust release process with a lot more considerations to solution merge behavior, but provides a way to easily remove your customization if done correctly without causing dependencies. I say this because the forums are full of users struggling to remove managed customizations for various reasons.  Unmanaged solutions do not require this level of robustness in the release process, and also provides the flexibility to perform snapshots of the production to resolved blocking issues. Your organization may decide to deploy unmanaged solutions if they feel at risk of losing business critical data because unmanaged solutions can be easily uninstalled in error,  or they may not have the release process maturity in place, they could also be expecting to make changes the deployment such as migrate from on premise to online etc. In such scenarios deploying unmanaged solutions have their benefits, as they have the option to switch to a managed deployment in the future whereas it is a lot more difficult vice versa once you release as Managed you cannot covert it back to Unmanaged in your production without considerable rework.