JCMS Backlog Item Specification

|  |  |
| --- | --- |
| Sprint |  |
| Backlog Item ID |  |
| Title | Technical Design of JCMS Access Installer Upgrader Application |
| Assigned To | Michael Amato |
| Date Last Revision | 1/21/2013 11:49 AM |

Description 1

Requirements 1

Potential Initial Conditions 2

Changes from Previous Installer and Upgrader……………………………………………………………. 3

Steps Involved 4

Technical Design 5

Use Notes 6

Testing Notes 6

# Description

The purpose of this document is to outline the plans for designing an application that will serve as both an installer and an upgrader for JCMS Access. Aside from combining what used to be two applications (the JCMS installer and the JCMS Access upgrader) the installer will now install MySQL as the database associated with the application instead of Microsoft Access.

# Requirements

* The JCMSInstaller/Upgrader can only install/upgrade a JCMS interface (the access portion) that exists on the local machine, but can upgrade a database existing on ANY remote machine.
* User must have ODBC connector installed no matter what (they need it to link tables to MySQL DB)
* User will need to establish ODBC connection to
* If the user doesn’t have ODBC and cancels ODBC installation they should receive a message saying the ODBC is a required software element and not allow them to continue until the ODBC driver is installed.
* The first form that appears should display to the user what they currently have installed on their machine (MySQL and JCMS) and prompt them to make a decision based on the software they have installed. For instance if a user has neither MySQL or JCMS installed in all likelihood they will want to get a fresh install, but it’s completely feasible they could be using a copy of JCMS that was not installed (just moved over from another machine) and connecting to a remote interface.
* User needs to be able to do a fresh installation on their machine OR upgrade an existing installation.
* In order for upgrader to run the user MUST have a MySQL backend already. If they do not they must be prompted to make sure MySQL is installed, give them the location to download MySQL from, and download the converter from the JCMS website and run it on their database.
* If a user is doing a fresh install they must be able to choose whether they would like to install on their local machine (localhost) or a remote machine (jcms-dev, jcms-test for instance).
* User doing a fresh install must be able to choose to install MySQL in the case their database will exist on local machine and they do not already have MySQL installed or not install MySQL in the case that they already have MySQL installed OR they wish to install on a remote system (like jcms-dev or jcms-test).
* If a user elects to install MySQL they must use the GUI windows MySQL installer which will be included in the JCMS installer and started automatically if a user chooses to install MySQL. Note the user needs both MySQL server AND ODBC connector in order to get JCMS running so both need to be installed.
* For a fresh installation after a user either installs MySQL or elects not to they will be given a form that looks a lot like the JCMS Converter form giving the specifics of the MySQL server and Machine on which they wish to put the database and the install directory for their new interface. JCMS will also be added to the directory as an installed program so the user can uninstall it.
* If a user cancels while in the process of doing a fresh installation of JCMS and MySQL has already been installed a warning must appear saying that while JCMS has not been installed MySQL has and give instructions for uninstalling MySQL if they wish to.
* A user will NOT be able to install MySQL on a remote machine using our JCMS installer. They will be able to target a remote machine that ALREADY HAS MySQL installed and install on that.
* If a user elects to upgrade they will be given a form very similar to the JCMS converter form.
* If a user cancels in the middle of an upgrade changes must roll back to original state.
* A BACKUP MUST BE MADE OF THE TARGET DATABASE AND INTERFACE AT THE BEGINNING OF THE UPGRADE PROCESS NO MATTER WHAT!!
* The upgrader must be easy to add to for a new release perhaps limit to making changes to one property file (for things like version etc.) and adding one file for the upgrade (something like 4-6-0\_to\_4-7-0.xml)
* When a user uninstalls JCMS any JCMS database should be backed up in the case that a user unwittingly uninstalls MySQL thinking they won’t lose their databases.

# Potential Initial Conditions

## Neither MySQL or JCMS Installed

In all likelihood this is the case of a brand new user and they should be advised to select a fresh installation unless there is already a JCMS.mdb file they currently use on their local machine.

A user could already have a fully functional JCMS set up as well in the case that they got the interface from another user and just copied it onto their host computer instead of running the JCMS installer and have their database living in a MySQL server installed on another machine that they are connected to.

It is also possible that the user has JCMS but with an Access backend in the case where JCMS was acquired without using the installer (copied from another instance perhaps). In this case a user needs to be prompted to install MySQL and run the converter found at <http://colonymanagement.jax.org/jcms_db_conversion_tool.html>.

## JCMS Installed, MySQL not installed

For a while this will probably be the case you see when a user is operating using an Access backend. If the user says this is the case then they should be prompted to install MySQL and run the converter found at <http://colonymanagement.jax.org/jcms_db_conversion_tool.html>.

Once the Access backend has become less prevalent this case will probably mean that the user is connected to a database on a MySQL server installed on a remote machine (jcms-test or jcms-dev would be examples). If a user believes this is the case the user should see a form similar to the form for the converter and be able to put in the specifics for their database (machine name, MySQL username and PW, a dropdown list containing all the database names for the user to select which they would like to upgrade and the directory where their interface is located).

## MySQL Installed, JCMS not Installed

This case is difficult to predict as it could feasibly be one of two possible scenarios:

1. A user has MySQL installed for some other application and has never used JCMS. This user should follow the fresh install path but skip the MySQL server installation process and just get right to a screen similar to the JCMS Converter where they specify all the machine/MySQL specifics as well as what they would like their database to be named (jcms\_db, jcmsAmato\_db).
2. A user has received a JCMS interface without running the JCMS installer, in this case a user is likely doing an upgrade and if they select upgrade should be able to conduct a standard upgrade. A user may have MySQL installed on their local machine but their JCMS database is in a different machine, this possibility must be allowed as well.

## Both MySQL and JCMS Installed

In this case a user is most likely upgrading an existing installation on his local machine; however a user could also be upgrading an instance that is connected to a remote machine or desire a new installation (essentially another copy of the interface with links to another database in the case they are working with multiple colonies).

# Changes from previous installer and upgrader

## Installer

The previous installer would not let you run if you had already installed JCMS. The new installer will allow you to install multiple JCMS instances. The way it will work is for the first install the installer will add JCMS to the registry and that location will be where the interface is deleted from in the case the user uninstalls it. It would be ideal if the registry were edited on all subsequent installs so that every JCMS instance is deleted when a user uninstalls, but I do not know if this is possible.

The other difference is this installer will also install MySQL if the user desires it.

## Upgrader

At the moment there are two upgraders: JCMS Access Upgrader (JAUP) and JCMS MySQL upgrader (JMyUP). With the release of this product the JAUP will be deprecated and no longer updated past 4.6.0. If a user wishes to upgrade past 4.6.0 then they will have to have a MySQL backend.

JMyUP in its current form is not an application, just a series of scripts kicked off by a .sh file in the case of linux or mac and a .cmd file in the case of windows. This application will be a clickable app with a much greater level of user friendliness than the current method.

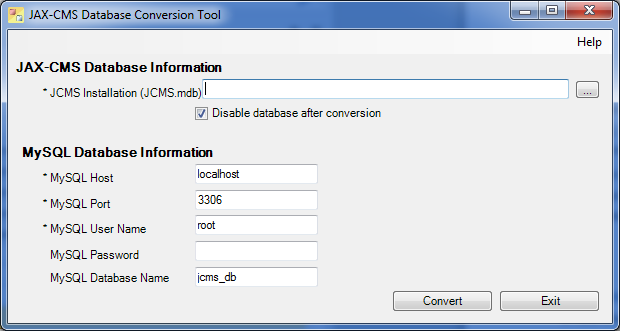
# Steps involved

A flow chart and outline of all the forms made in Microsoft Visio is available in Redmine under the JCMS Access project.

The first step regardless of whether an install or an upgrade is desired is to detect software. When the application has looked for and either found or not found JCMS, MySQL, and the MySQL ODBC driver the user will be presented with a decision on whether they wish to install or upgrade an existing installation

## Installer

The first form will either be a question prompting the user whether they wish to install MySQL locally in the case that MySQL was not found among the installed software on the local machine or in the case that MySQL was found installed locally the user will see a form very similar to what you see below (the main form for the JCMS Converter). The user will proceed to put in the information for their MySQL database/host machine.



In the case where is a user is prompted as to whether or not they wish to install MySQL if the user elects to install MySQL they will install both MySQL and the ODBC driver and then see the above screen. If they choose not to install MySQL and are connecting to a remote machine they will go directly to the above screen.

## Upgrader

The upgrader will take you to a similar screen as above, but the MySQL Database Name field will be a dropdown list instead of an input text field, the list will be populated with all the databases found in the host MySQL server. If there is a problem with the upgrade (database isn’t a JCMS database or was broken by user adding custom fields or whatever) a human readable, helpful error message must be displayed.

# Technical Design

The application will be a C# forms application built in Microsoft Visual Studio.

## Installer

Write to registry, kick off MySQL installation, connect to MySQL create schema, make jcms\_db, link tables.

## Upgrader

Connect to MySQL, run updates parsed from xml files by application, relink tables.

# Use Notes

# Testing Notes

This application will be written exclusively for use in a windows environment (Microsoft Access is only available for Windows) and should be tested for at least the following cases:

1. Fresh install that is installing MySQL, ODBC and JCMS
2. Fresh install that is only installing ODBC and JCMS
3. Fresh install that is only installing MySQL and ODBC (this is case where a database needs to be converted)
4. Fresh install that is only installing JCMS (user already has ODBC)
5. For all the above four cases both placing the db locally when MySQL is installed locally and placing db on different host machine.
6. Upgrading a local instance and installing ODBC
7. Upgrading a remote instance and installing ODBC
8. Upgrading a remote instance with ODBC already installed
9. Upgrading local instance with ODBC already installed