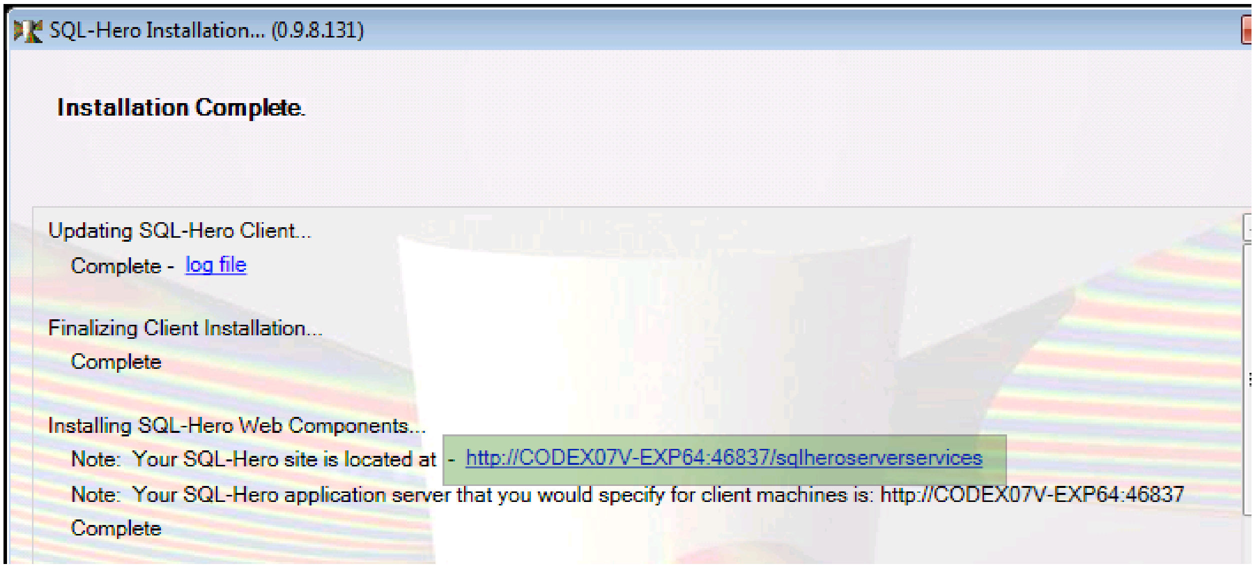
Revision	Description
6/14/2011	Original
9/5/2011	Updates based on changes in portal

SQL-Hero Web Portal

Introduction

The SQL-Hero web portal is installed as part of SQL-Hero web components (it now co-exists in the same location with the exposed SOA/WCF API for SQL-Hero). During installation, you'll see a link which takes you to the site:


SQL-Hero Installation... (0.9.8.131)

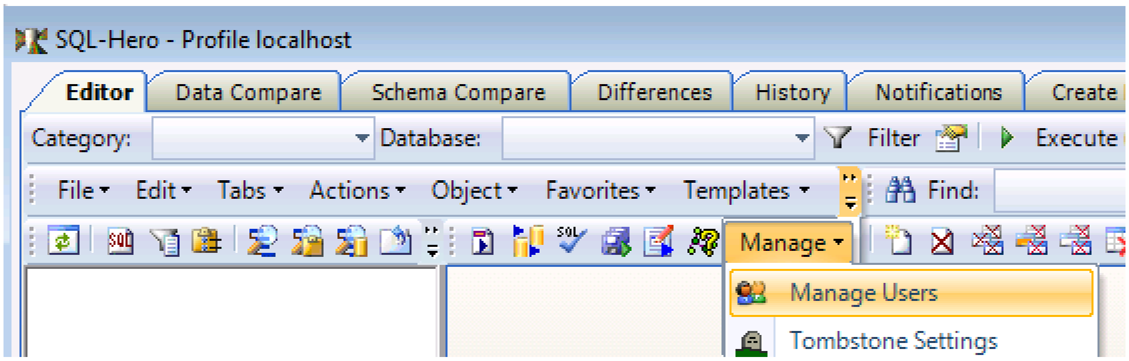
Installation Complete.

Updating SQL-Hero Client...
Complete - log file

Finalizing Client Installation...
Complete

Installing SQL-Hero Web Components...
Note: Your SQL-Hero site is located at - http://CODEX07V-EXP64:46837/sqlheroserverservices
Note: Your SQL-Hero application server that you would specify for client machines is: http://CODEX07V-EXP64:46837
Complete

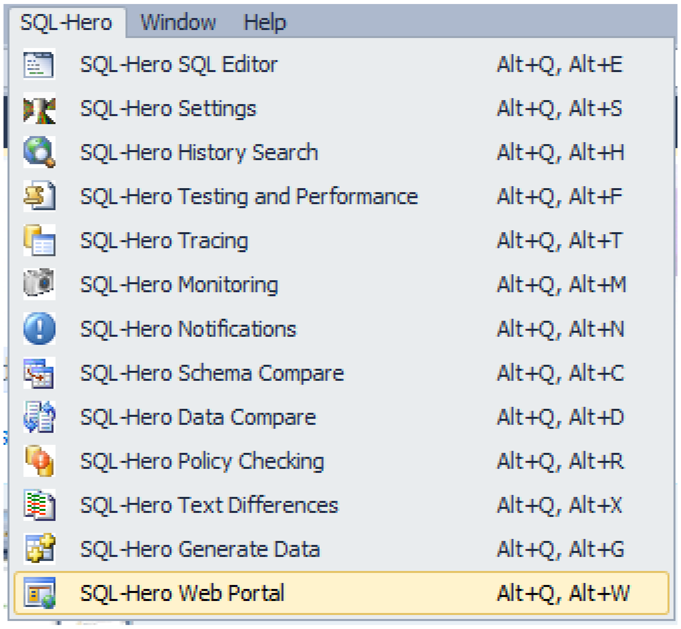
The site serves important functions, for both reporting and tracking changes. Any developer on your team can access the portal, so long as they are registered in the SQL-Hero repository. Users are automatically registered if they make changes in tracked databases (using Windows Authentication), or they can be added explicitly using the Manage -> Manage Users menu option:



If you are using IIS 7.0+, have HTTP redirection enabled, and have installed SQL-Hero as a *site* (not just an application underneath an existing site), you will be able to reference the portal using just the server name and port number – for example, "http://myhost:46837". The typical way to create a new site is to use a port number that's not assigned to an existing site. The default port SQL-Hero uses is 46837 although this can be altered during installation, if desired.

If you have Windows authentication installed, when you connect to the portal, it will try to log you in using your Windows credentials – only if this fails will you be asked to log in using a login page.

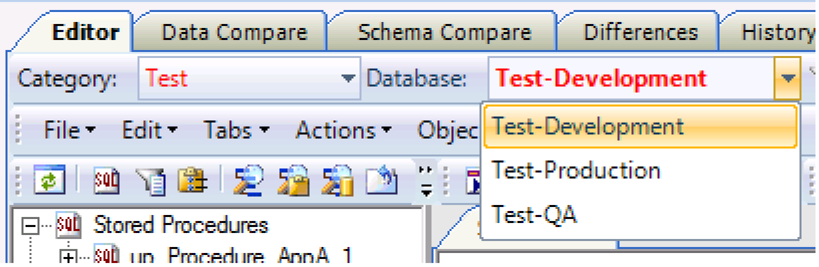
We've added two additional ways to reach the web portal: from SQL-Hero itself, and when using Visual Studio integration. From SQL-Hero, the Portal tab is effectively an embedded browser that uses your current application server setting to determine the portal location. (If you don't have server components installed, you get some additional details about what server components are and the overall SQL-Hero architecture.) In Visual Studio, the Portal is available as a tool window accessible from the main menu:


SQL-Hero
Window
Help
SQL-Hero SQL Editor Alt+Q, Alt+E
SQL-Hero Settings Alt+Q, Alt+S
SQL-Hero History Search Alt+Q, Alt+H
SQL-Hero Testing and Performance Alt+Q, Alt+F
SQL-Hero Tracing Alt+Q, Alt+T
SQL-Hero Monitoring Alt+Q, Alt+M
SQL-Hero Notifications Alt+Q, Alt+N
SQL-Hero Schema Compare Alt+Q, Alt+C
SQL-Hero Data Compare Alt+Q, Alt+D
SQL-Hero Policy Checking Alt+Q, Alt+R
SQL-Hero Text Differences Alt+Q, Alt+X
SQL-Hero Generate Data Alt+Q, Alt+G
SQL-Hero Web Portal Alt+Q, Alt+W

Claiming Objects

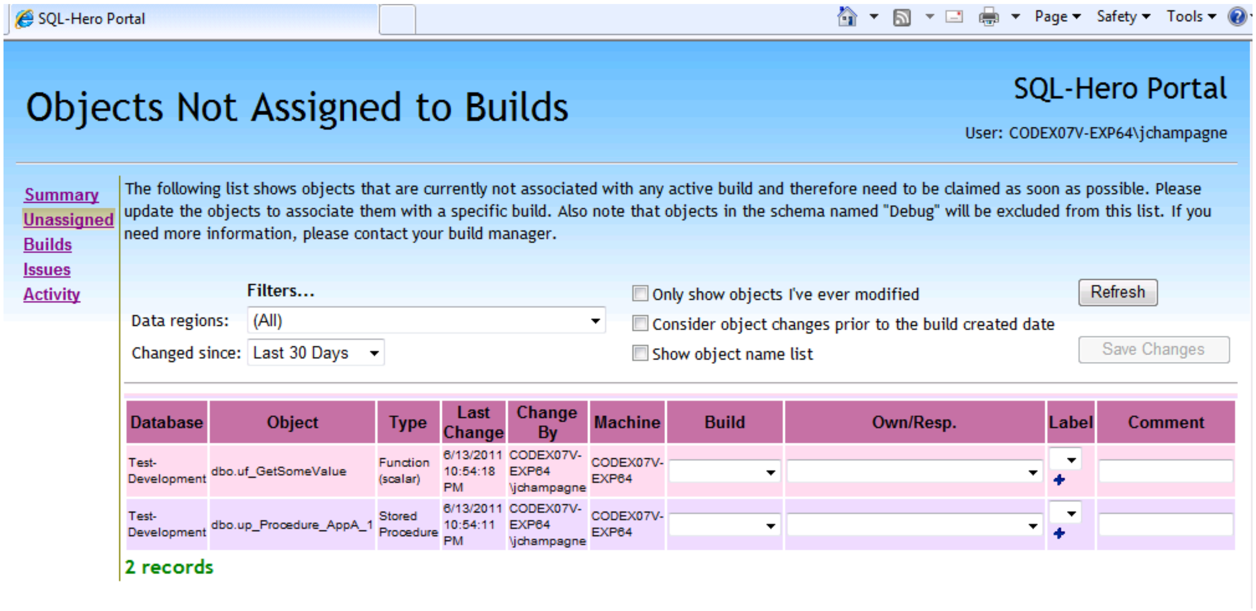
Some organizations will appreciate a "non-intrusive" database development process which allows developers to work with either a shared development database or individual sandbox databases. By "non-intrusive", we mean that developers are free to make changes in the live development database without having to pay a lot of attention to details like "what objects did I end up changing in this session?" or "now I need to commit all the work I've done to a source control system." Those who are source control purists may say "Tough! Deal with it: it's part of a robust process." However, we've observed that streamlining the process isn't all that hard while leveraging some of SQL-Hero's strengths to claw back a level of robustness. SQL-Hero does integrate with external source control systems such as TFS as well, but the process discussed here tends to be suited to organizations that feel the benefits of simplified but more manual change management outweigh the costs. Since SQL-Hero supports multiple models, the choice is largely organizational.

One way to achieve streamlining is to use the new SQL-Hero web portal's build pages. Let's start by looking at a scenario where we have developers using a shared development database. This database may serve a single application or many applications – and we need to worry about handling builds for all applications independently. Let's say we have a development, QA and production database:



Let's assume a developer does some work in the development database and changes a stored procedure and a UDF. It's not hard to imagine a busy developer perhaps remembering the procedure but forgetting that the UDF was changed. Of course different tools make it harder: such as TFS and Database Projects, where your objects would be checked out, awaiting check-in as part of a possibly larger change set. That topic is covered in a different whitepaper.

Assuming you wanted a simpler tracking system, if one were to visit the SQL-Hero web portal and pick the "Unassigned" menu option, it might look like this:



Objects Not Assigned to Builds

SQL-Hero Portal

User: CODEX07V-EXP64\jchampagne

The following list shows objects that are currently not associated with any active build and therefore need to be claimed as soon as possible. Please update the objects to associate them with a specific build. Also note that objects in the schema named "Debug" will be excluded from this list. If you need more information, please contact your build manager.

Filters...

Data regions: (All)

Changed since: Last 30 Days

☐ Only show objects I've ever modified

☐ Consider object changes prior to the build created date

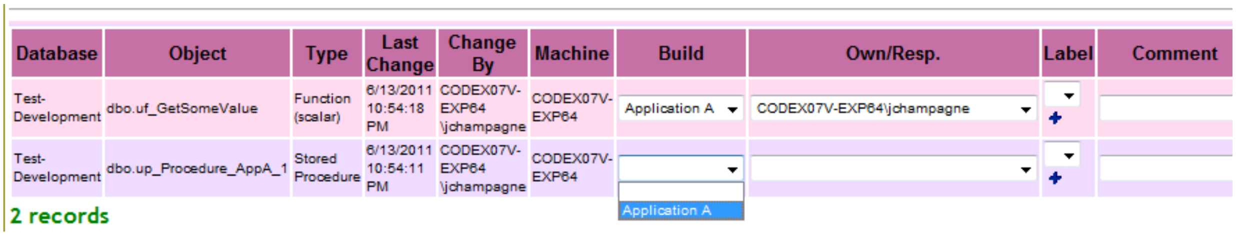
☐ Show object name list

Database	Object	Type	Last Change	Change By	Machine	Build	Own/Resp.	Label	Comment
Test-Development	dbo.uf_GetSomeValue	Function (scalar)	6/13/2011 10:54:18 PM	CODEX07V-EXP64 \jchampagne	CODEX07V-EXP64				
Test-Development	dbo.up_Procedure_AppA_1	Stored Procedure	6/13/2011 10:54:11 PM	CODEX07V-EXP64 \jchampagne	CODEX07V-EXP64				

2 records

Notice how the portal "knows" what objects have been changed but are not associated with a particular build. We can perform that association directly on the site – simply pick the desired application and "Save Changes". A "responsible user" is also tracked and defaulted to the logged in user. The "label" drop-down and add (+) button allow you to pick an existing reference value or add a new one. Reference values can be bug numbers, or anything else you prefer to use to track work.

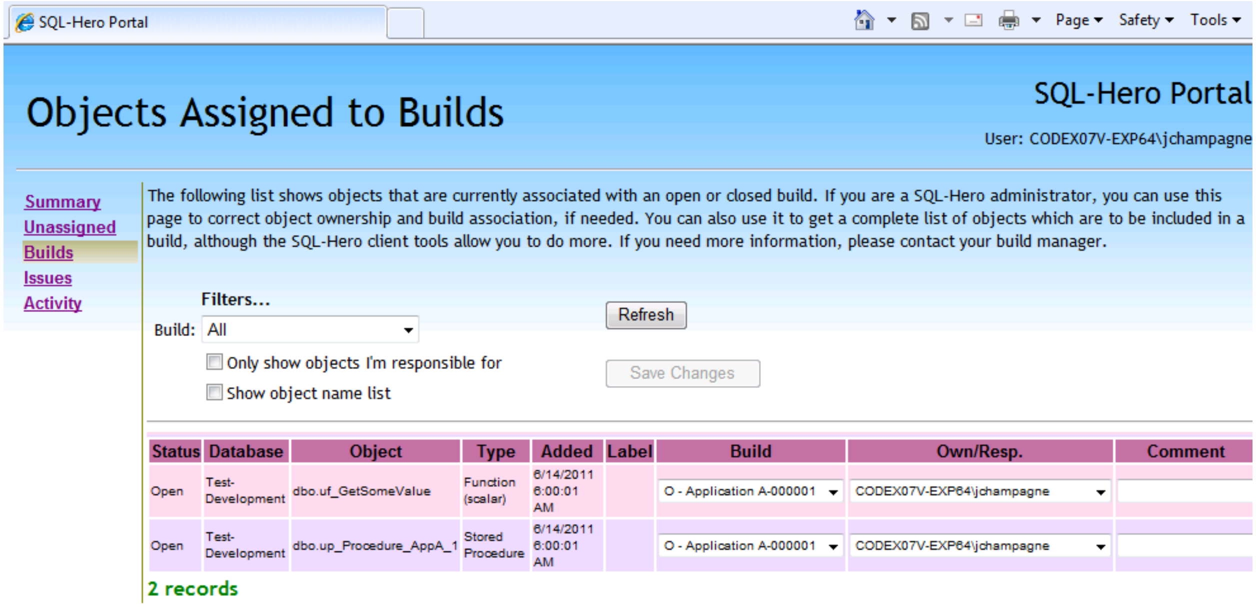
Also note that objects which have published region difference compliance failures are *excluded* from the "not assigned to builds" page's results. Including them could result in "lost updates" in the target region. The last section covers a report which shows these compliance issues separately.



Database	Object	Type	Last Change	Change By	Machine	Build	Own/Resp.	Label	Comment
Test-Development	dbo.uf_GetSomeValue	Function (scalar)	6/13/2011 10:54:18 PM	CODEX07V-EXP64 \jchampagne	CODEX07V-EXP64	Application A	CODEX07V-EXP64\jchampagne		
Test-Development	dbo.up_Procedure_AppA_1	Stored Procedure	6/13/2011 10:54:11 PM	CODEX07V-EXP64 \jchampagne	CODEX07V-EXP64				

2 records

Once these values have been assigned, clicking on the Save Changes button clears away the unassigned list, and the same objects now appear on the assigned list:



Objects Assigned to Builds

User: CODEX07V-EXP64\jchampagne

The following list shows objects that are currently associated with an open or closed build. If you are a SQL-Hero administrator, you can use this page to correct object ownership and build association, if needed. You can also use it to get a complete list of objects which are to be included in a build, although the SQL-Hero client tools allow you to do more. If you need more information, please contact your build manager.

Filters...

Build: All

□ Only show objects I'm responsible for

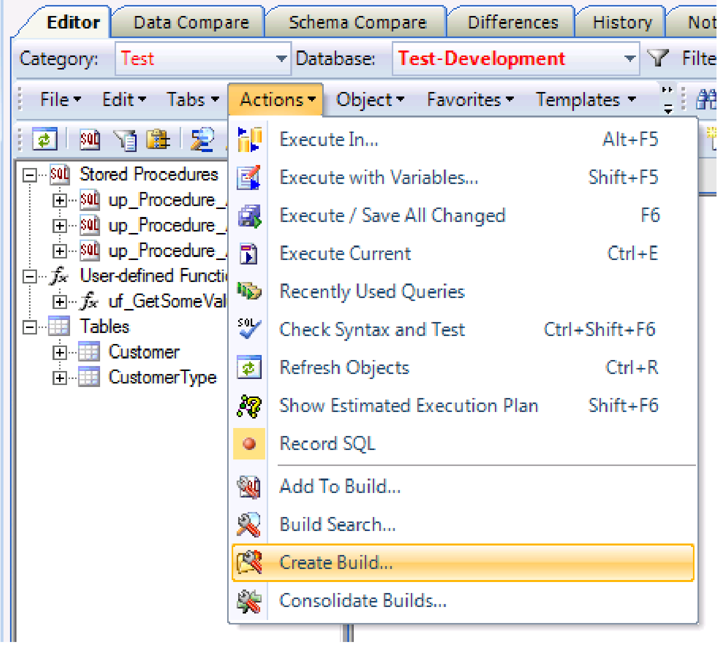
□ Show object name list

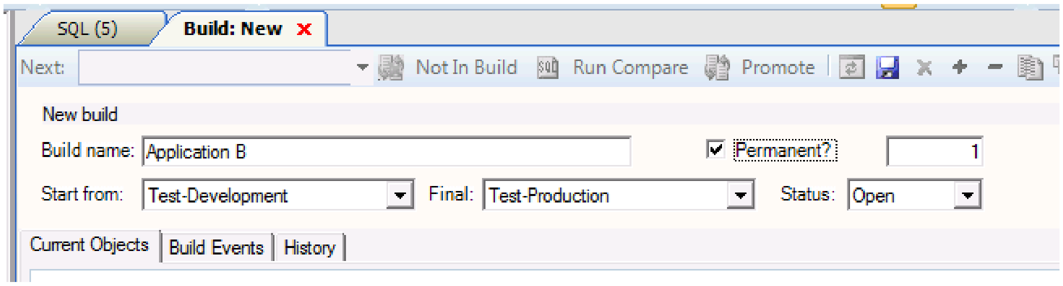
Status	Database	Object	Type	Added	Label	Build	Own/Resp.	Comment
Open	Test-Development	dbo.uf_GetSomeValue	Function (scalar)	6/14/2011 6:00:01 AM		O - Application A-000001	CODEX07V-EXP64\jchampagne	
Open	Test-Development	dbo.up_Procedure_AppA_1	Stored Procedure	6/14/2011 6:00:01 AM		O - Application A-000001	CODEX07V-EXP64\jchampagne	

2 records

(The assigned list is editable only for SQL-Hero administrators, mainly to correct mistakes – we trust that most assignments will be correct!)

But how does one define the values in the build drop-down? Given this is intended to be a one-time task, it's something that can be done using the SQL-Hero client application, SQL Editor tool. In this example, we used "Create Build…":





New build

Build name: Application B

☑ Permanent? 1

Start from: Test-Development

Final: Test-Production

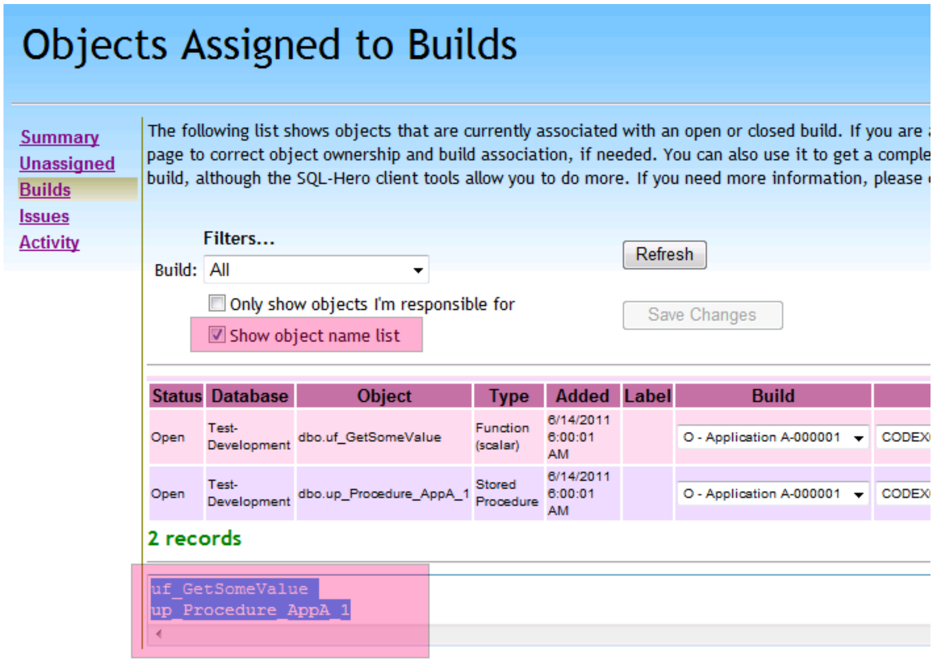
Status: Open

We provide a friendly build name, specify which database serves as the starting point for object movement, and the final database where, when objects reach it, the build is considered complete. The status of "Open" implies we are free to add new objects to the build. When a build is promoted to any other database, the status automatically changes to "Closed" which means we can no longer add new objects, but the build is not yet complete. The Permanent check-box, when checked, implies that if the build changes from Open to Closed or Completed, a new build with the same name will be created in Open status, meaning we'll always be able to add new objects using the build in the starting database. Because of this, a sequence number is used to identify the build – the combined name and sequence number must be unique.

With objects now easily matched to applications, let's look at how easy it is to actually perform builds.

Modified Build Process

With objects now "claimed" by developers and associated to builds, we have a way to reliably know what objects to promote from one database region to the next logical region. The build process itself has streamlined support using the SQL-Hero client application (or Visual Studio integration), although technically one can work with raw object lists right from the portal itself. For example, you can copy-and-paste an object name list if you check "Show object name list":



Objects Assigned to Builds

The following list shows objects that are currently associated with an open or closed build. If you are
page to correct object ownership and build association, if needed. You can also use it to get a comple
build, although the SQL-Hero client tools allow you to do more. If you need more information, please

Filters...

Build: All

□ Only show objects I'm responsible for

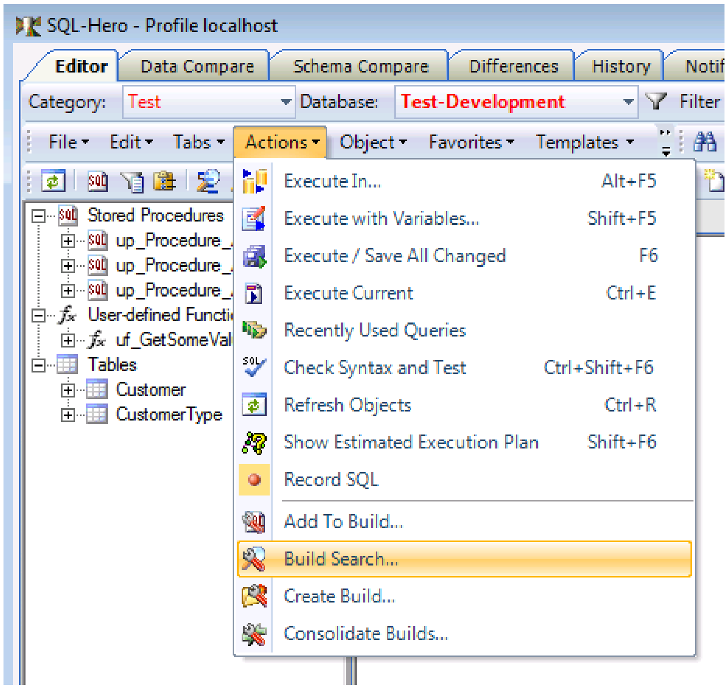
☑ Show object name list

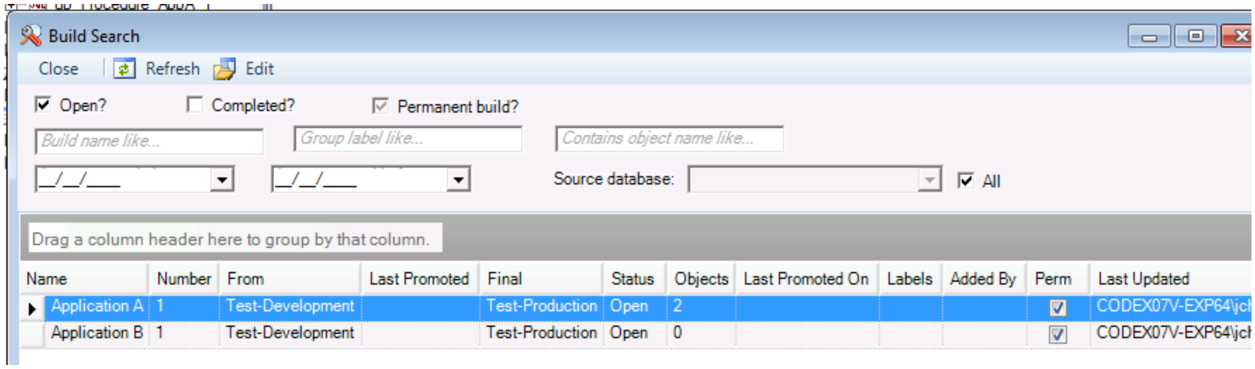
Status	Database	Object	Type	Added	Label	Build	
Open	Test-Development	dbo.uf_GetSomeValue	Function (scalar)	6/14/2011 6:00:01 AM		O - Application A-000001	CODEX
Open	Test-Development	dbo.up_Procedure_AppA_1	Stored Procedure	6/14/2011 6:00:01 AM		O - Application A-000001	CODEX

2 records

```
uf_GetSomeValue
up_Procedure_AppA_1
```

In the SQL-Hero client tool, we can search for open and closed builds:



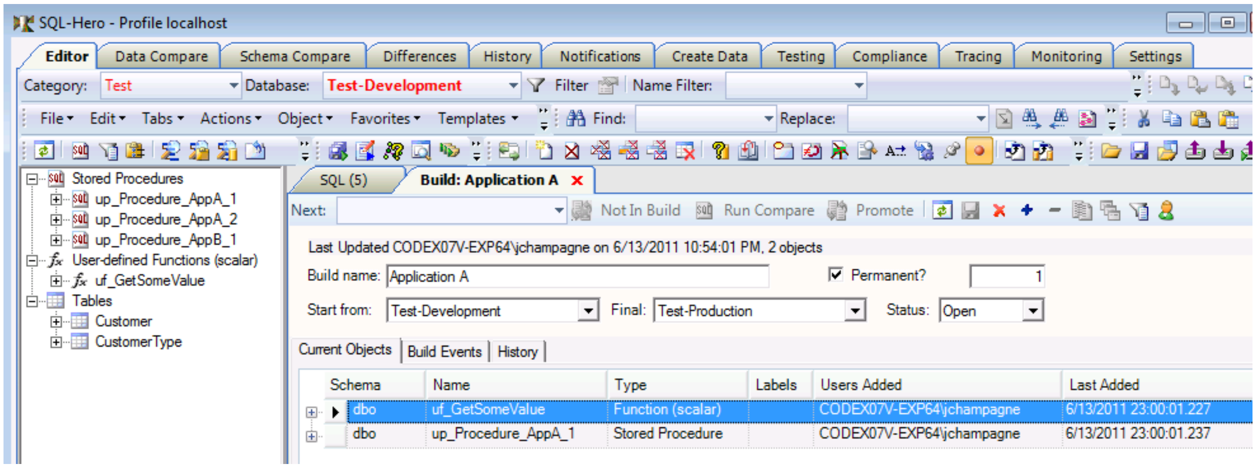

Build Search
Close
Refresh
Edit
Build name like...
Group label like...
Contains object name like...
Drag a column header here to group by that column.

☑ Open? ☐ Completed? ☑ Permanent build?

Source database: ☑ All

Name	Number	From	Last Promoted	Final	Status	Objects	Last Promoted On	Labels	Added By	Perm	Last Updated
Application A	1	Test-Development		Test-Production	Open	2				☑	CODEX07V-EXP64\jc
Application B	1	Test-Development		Test-Production	Open	0				☑	CODEX07V-EXP64\jc

If we double-click the first grid row (or click Edit), you're brought to the build details panel:



Category: Test Database: Test-Development

Next:

Last Updated CODEX07V-EXP64\jchampagne on 6/13/2011 10:54:01 PM, 2 objects

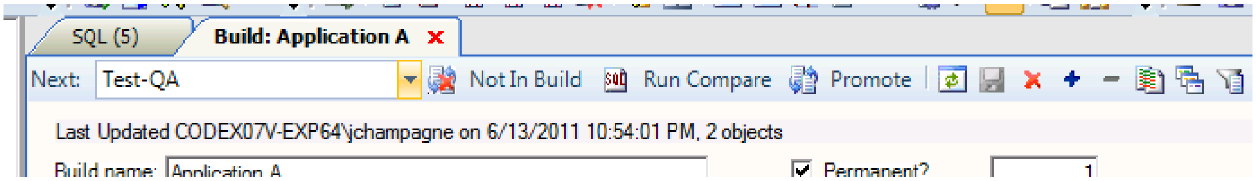
Build name: Application A ☑ Permanent? 1

Start from: Test-Development Final: Test-Production Status: Open

Schema	Name	Type	Labels	Users Added	Last Added
dbo	uf_GetSomeValue	Function (scalar)		CODEX07V-EXP64\jchampagne	6/13/2011 23:00:01.227
dbo	up_Procedure_AppA_1	Stored Procedure		CODEX07V-EXP64\jchampagne	6/13/2011 23:00:01.237

Here we see the objects that comprise the build, any labels (e.g. bug numbers) that apply to each, and a log of events that relate to the build.

To actually promote objects from one database to another, pick the "next" database to move to in the drop-down in the panel tool bar:

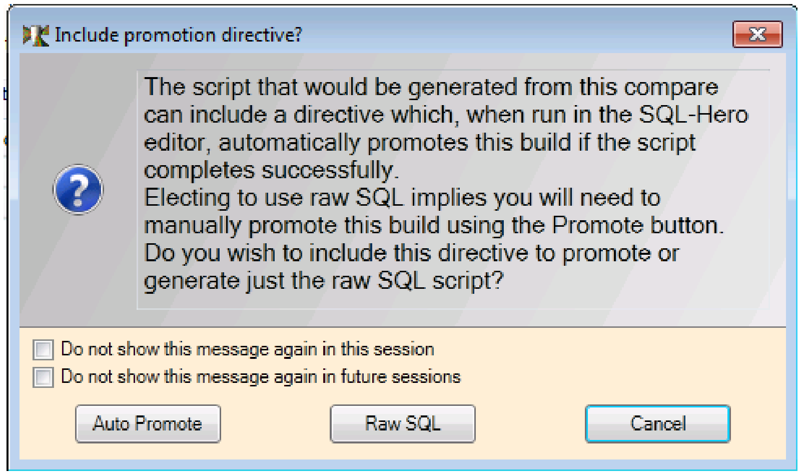


Next: Test-QA

Last Updated CODEX07V-EXP64\jchampagne on 6/13/2011 10:54:01 PM, 2 objects

Build name: Application A ☑ Permanent? 1

Notice how some tool bar buttons are now enabled. In the simplest scenario, we now click "Run Compare". We're asked if the script that gets generated should include a directive that will cause SQL-Hero to update the build information in the repository when the script executes successfully.



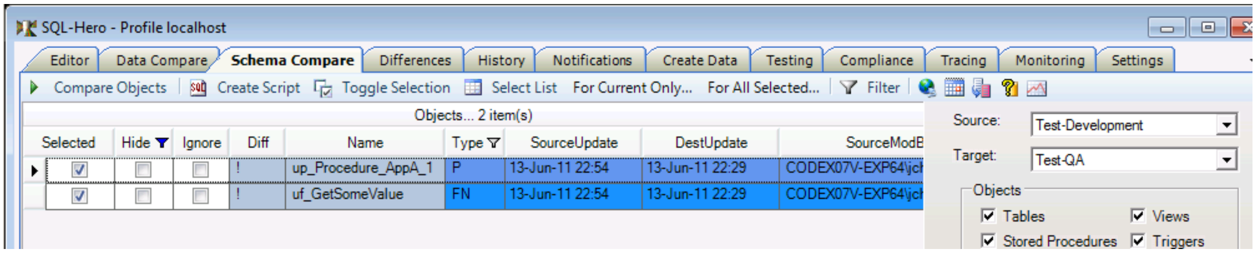
Include promotion directive?

The script that would be generated from this compare can include a directive which, when run in the SQL-Hero editor, automatically promotes this build if the script completes successfully.
Electing to use raw SQL implies you will need to manually promote this build using the Promote button.
Do you wish to include this directive to promote or generate just the raw SQL script?

☐ Do not show this message again in this session
☐ Do not show this message again in future sessions

If you intend to run the generated script outside of SQL-Hero, then Raw SQL may be a valid choice, although the directive that's added is added inside a SQL comment, so using Auto Promote produces script that can be executed anywhere. If you do not use Auto Promote or will not run the script in SQL-Hero, you'll need to manually click on the Promote button after you've applied the change script, as an extra step. (The promotion step changes a build's status from Open to Closed, so it's an important thing to do and should not be skipped.)

Let's assume we click Auto Promote. Now we're brought to the Schema Compare tool and the compare is issued automatically for us. We'll see the difference list, filtered to the objects involved in the build:



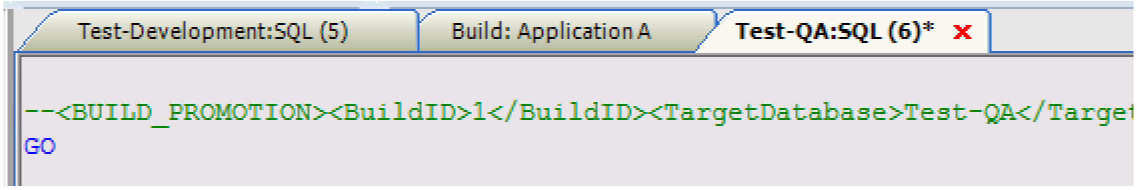
Objects... 2 item(s)

Selected	Hide	Ignore	Diff	Name	Type	SourceUpdate	DestUpdate	SourceModB
☑	☐	☐	!	up_Procedure_AppA_1	P	13-Jun-11 22:54	13-Jun-11 22:29	CODEX07V-EXP64\jcl
☑	☐	☐	!	uf_GetSomeValue	FN	13-Jun-11 22:54	13-Jun-11 22:29	CODEX07V-EXP64\jcl

Source: Test-Development
Target: Test-QA

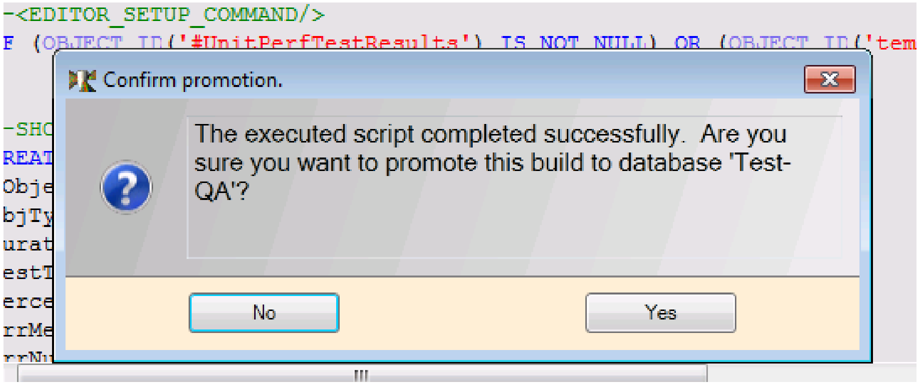
Objects
☑ Tables ☑ Views
☑ Stored Procedures ☑ Triggers

Clicking "Create Script" gives us the actual change script to apply. Note that it includes the promotion directive, with the target database named, and the script's database set to QA. At this point we only need to hit F5 to execute the script:



```
--<BUILD_PROMOTION><BuildID>1</BuildID><TargetDatabase>Test-QA</Target
GO
```

With the successful completion of the script, we're given a message that gets us to confirm the promotion operation:



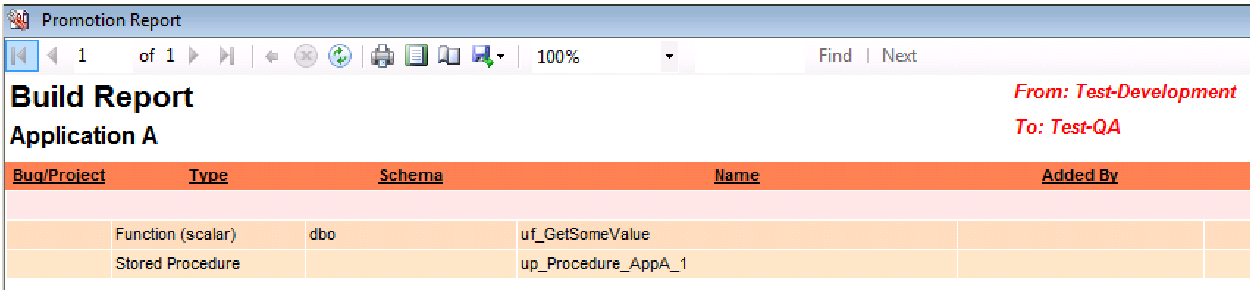
```
-<EDITOR_SETUP_COMMAND/>
F  (OBJECT_ID('#UnitPerfTestResults') IS NOT NULL) OR (OBJECT_ID('tem


-SHO
REAT
Obje
bjTy
urat
estT
erce
rrMe
rrNu
```

Confirm promotion.

The executed script completed successfully. Are you sure you want to promote this build to database 'Test-QA'?

After the promotion completes, a build report appears listing objects involved in the build. The report can be exported to PDF or Excel if the intent is to share the content.



Build Report

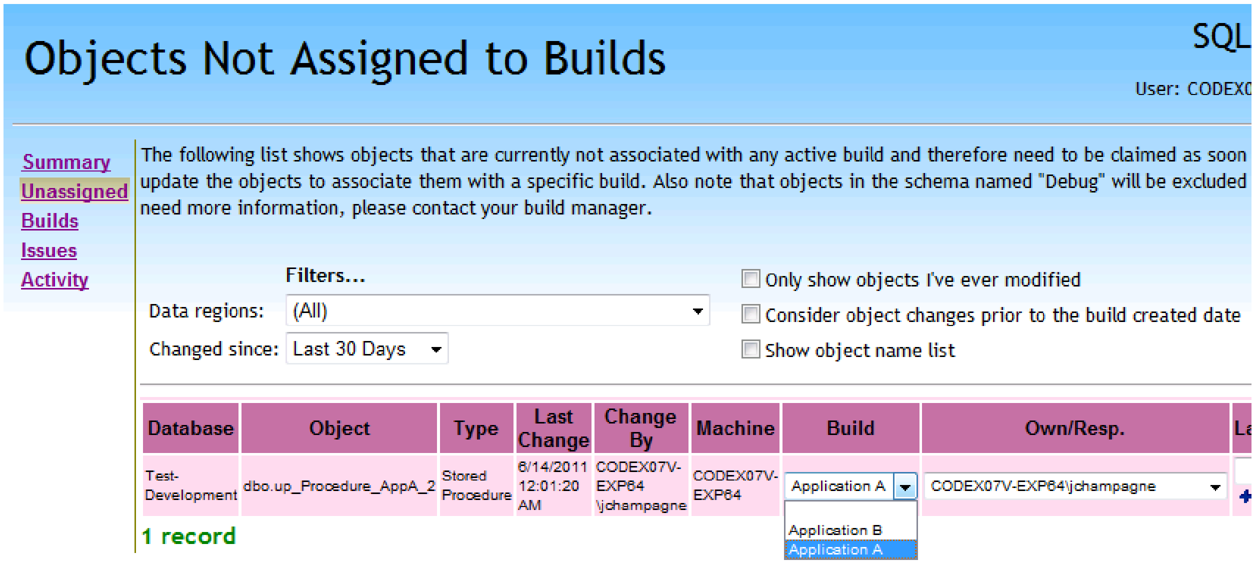
Application A

From: Test-Development

To: Test-QA

Bug/Project	Type	Schema	Name	Added By
	Function (scalar)	dbo	uf_GetSomeValue	
	Stored Procedure		up_Procedure_AppA_1	

Let's assume now that the user makes a change in another object in development. Back on the Unassigned list on the SQL-Hero portal, we can assign this object to the proper application:



Objects Not Assigned to Builds

The following list shows objects that are currently not associated with any active build and therefore need to be claimed as soon update the objects to associate them with a specific build. Also note that objects in the schema named "Debug" will be excluded need more information, please contact your build manager.

Filters...

Data regions: (All)

Changed since: Last 30 Days

□ Only show objects I've ever modified

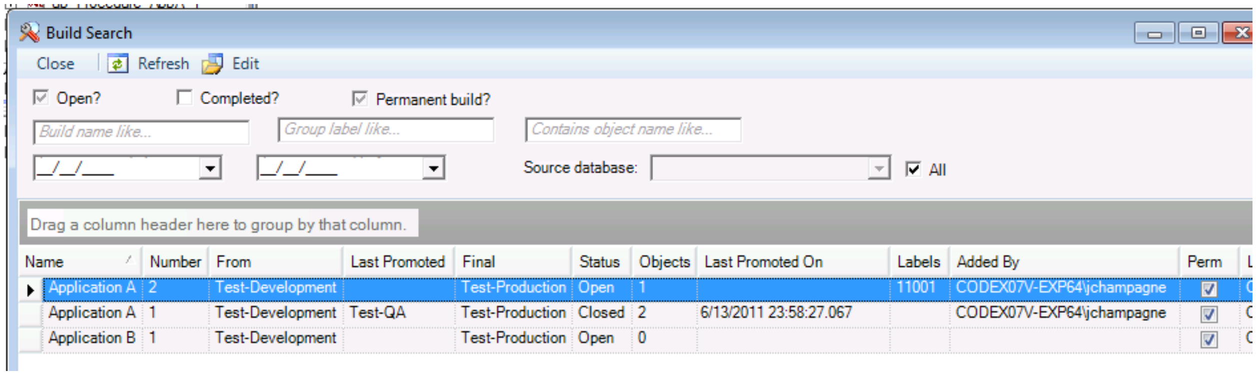
□ Consider object changes prior to the build created date

□ Show object name list

Database	Object	Type	Last Change	Change By	Machine	Build	Own/Resp.	La
Test-Development	dbo.up_Procedure_AppA_2	Stored Procedure	6/14/2011 12:01:20 AM	CODEX07V-EXP64 \jchampagne	CODEX07V-EXP64	Application A	CODEX07V-EXP64\jchampagne	

1 record

Now performing a build search, we can see how "Application A" has two on-going builds: numbers 1 and 2. #1 has 2 objects and was the one we just promoted (hence the status of Closed). #2 is the Open build we just added the above object to.



☑ Open? ☐ Completed? ☑ Permanent build?

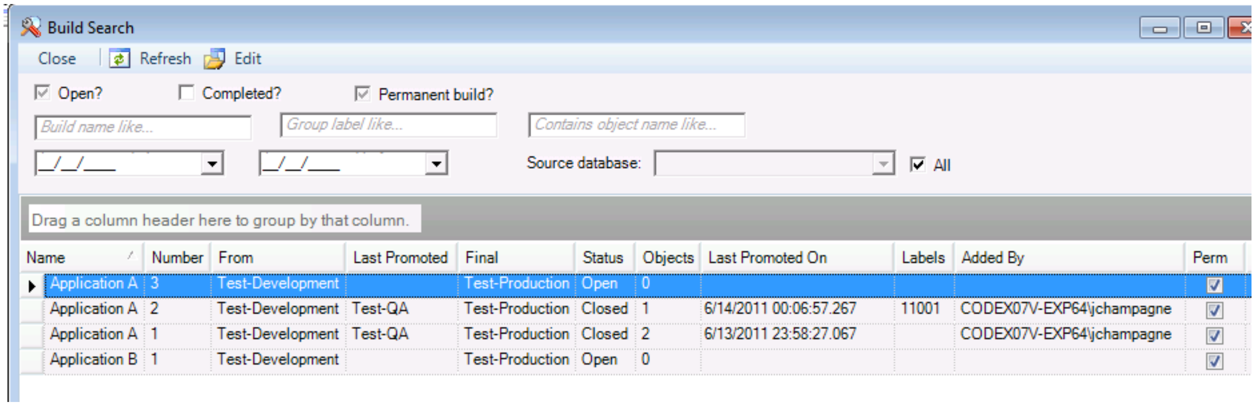
Build name like... Group label like... Contains object name like...

__/__/____ __/__/____ Source database: ☑ All

Drag a column header here to group by that column.

Name	Number	From	Last Promoted	Final	Status	Objects	Last Promoted On	Labels	Added By	Perm	L
Application A	2	Test-Development		Test-Production	Open	1		11001	CODEX07V-EXP64\jchampagne	☑	C
Application A	1	Test-Development	Test-QA	Test-Production	Closed	2	6/13/2011 23:58:27.067		CODEX07V-EXP64\jchampagne	☑	C
Application B	1	Test-Development		Test-Production	Open	0				☑	C

After another promotion to QA, we'd see:



☑ Open? ☐ Completed? ☑ Permanent build?

Build name like... Group label like... Contains object name like...

__/__/____ __/__/____ Source database: ☑ All

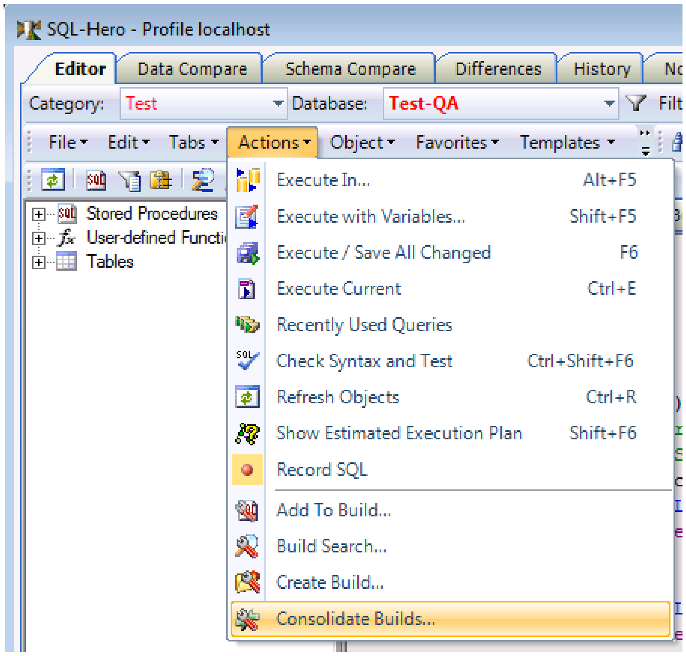
Drag a column header here to group by that column.

Name	Number	From	Last Promoted	Final	Status	Objects	Last Promoted On	Labels	Added By	Perm
Application A	3	Test-Development		Test-Production	Open	0				☑
Application A	2	Test-Development	Test-QA	Test-Production	Closed	1	6/14/2011 00:06:57.267	11001	CODEX07V-EXP64\jchampagne	☑
Application A	1	Test-Development	Test-QA	Test-Production	Closed	2	6/13/2011 23:58:27.067		CODEX07V-EXP64\jchampagne	☑
Application B	1	Test-Development		Test-Production	Open	0				☑

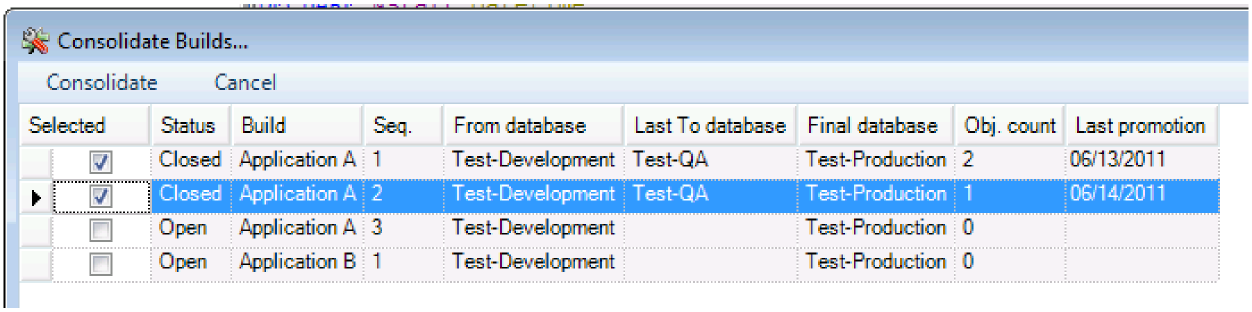
One can easily imagine a process by which regular builds are performed from development to QA, but much less frequent movements happen from QA to production. Do we really need to promote a large number of small builds in this case? Build consolidation is an alternative that can be used to combine multiple open or closed builds into one new build.

Build Consolidation

Combining builds is a straightforward process, launched from the SQL-Hero client tool:

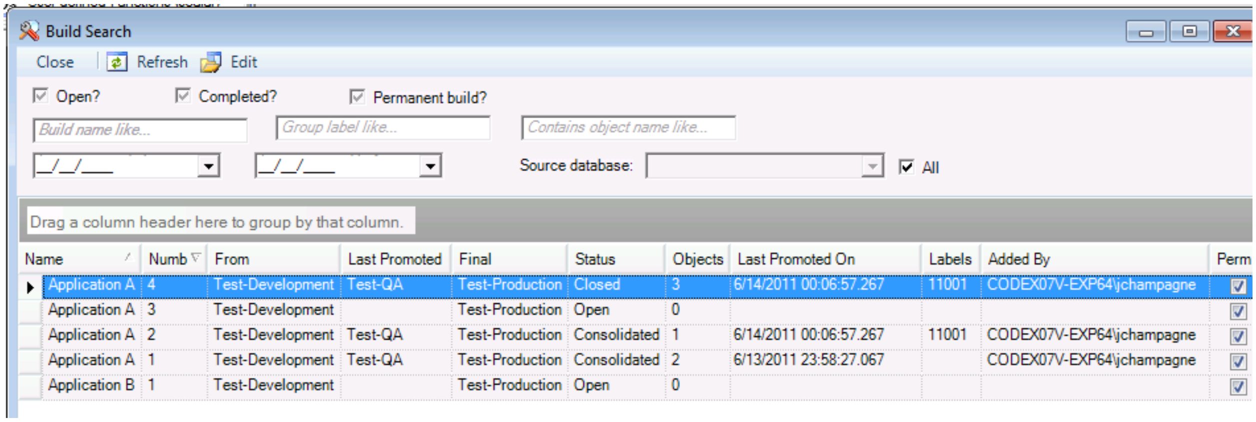


The option presents all open and closed builds. It's legal to combine both open and closed builds, but all selections must be for the same "From database", "Final database" and "Build name". If any build is in closed status, the new build will also be closed. It's also illegal to include an open build if it's flagged as being permanent: such a build should be changed to closed first (either manually or through promotion). The new consolidated build contains all unique objects from selected source builds.



Selected	Status	Build	Seq.	From database	Last To database	Final database	Obj. count	Last promotion
☑	Closed	Application A	1	Test-Development	Test-QA	Test-Production	2	06/13/2011
☑	Closed	Application A	2	Test-Development	Test-QA	Test-Production	1	06/14/2011
☐	Open	Application A	3	Test-Development		Test-Production	0	
☐	Open	Application B	1	Test-Development		Test-Production	0	

After the above consolidation is performed, a build search where we've elected to show completed builds would show this:



☑ Open? ☑ Completed? ☑ Permanent build?

Build name like... Group label like... Contains object name like...

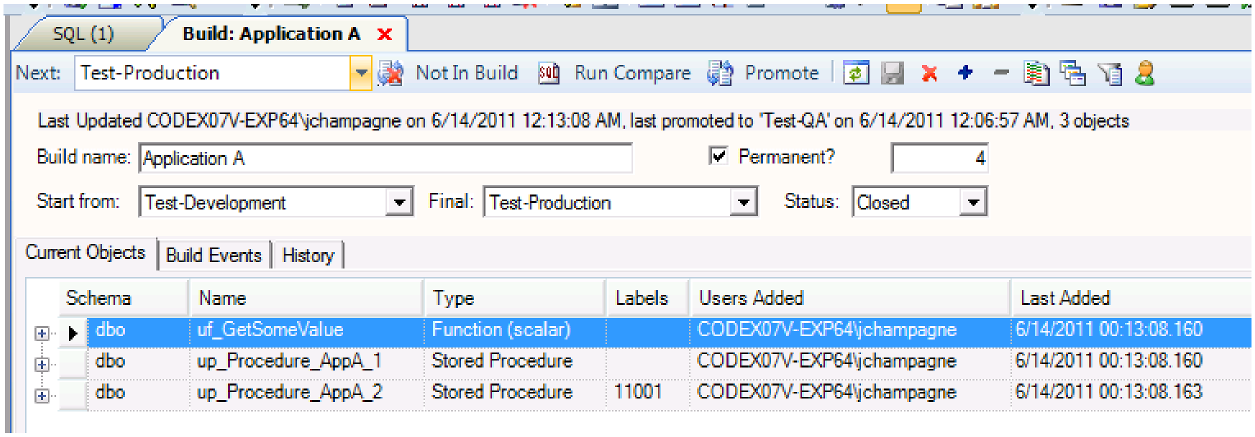
__/__/____ __/__/____ Source database: ☑ All

Drag a column header here to group by that column.

Name	Numb	From	Last Promoted	Final	Status	Objects	Last Promoted On	Labels	Added By	Perm
Application A	4	Test-Development	Test-QA	Test-Production	Closed	3	6/14/2011 00:06:57.267	11001	CODEX07V-EXP64\jchampagne	☑
Application A	3	Test-Development		Test-Production	Open	0				☑
Application A	2	Test-Development	Test-QA	Test-Production	Consolidated	1	6/14/2011 00:06:57.267	11001	CODEX07V-EXP64\jchampagne	☑
Application A	1	Test-Development	Test-QA	Test-Production	Consolidated	2	6/13/2011 23:58:27.067		CODEX07V-EXP64\jchampagne	☑
Application B	1	Test-Development		Test-Production	Open	0				☑

Notice how builds #1 and #2 for “Application A” now have a status of “Consolidated.” This status is treated the same as “Complete” for the most part: these builds can’t be used in further promotions. Instead we can use build #4, which is now in closed status and contains 3 objects: 2 from #1 and 1 from #2. Why does it show “Last Promoted On” with a date? Because build events are copied from the source builds to ensure the new build will “know” where objects it contains were last promoted *from* – an important piece of information when we want to next promote this build.

Let’s assume the 3 objects covered by the new consolidated build are ready to be moved to production. After picking the build on the search screen, we’d see:



Next: Test-Production

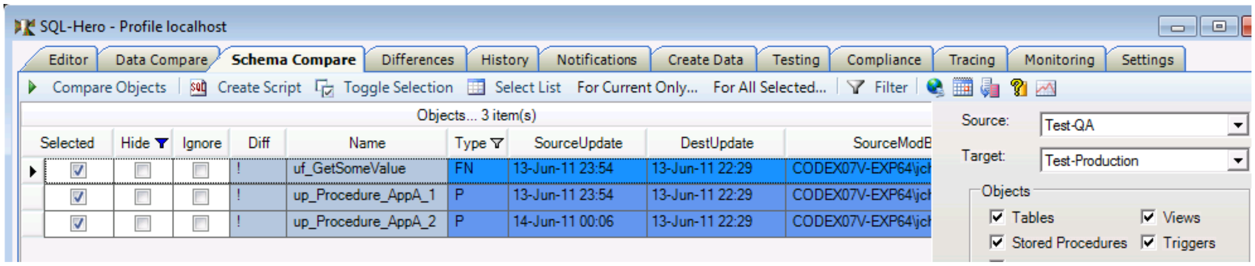
Last Updated CODEX07V-EXP64\jchampagne on 6/14/2011 12:13:08 AM, last promoted to 'Test-QA' on 6/14/2011 12:06:57 AM, 3 objects

Build name: Application A ☑ Permanent? 4

Start from: Test-Development Final: Test-Production Status: Closed

Schema	Name	Type	Labels	Users Added	Last Added
dbo	uf_GetSomeValue	Function (scalar)		CODEX07V-EXP64\jchampagne	6/14/2011 00:13:08.160
dbo	up_Procedure_AppA_1	Stored Procedure		CODEX07V-EXP64\jchampagne	6/14/2011 00:13:08.160
dbo	up_Procedure_AppA_2	Stored Procedure	11001	CODEX07V-EXP64\jchampagne	6/14/2011 00:13:08.163

After clicking on “Run Compare”, we’d see (notice that the Source database has correctly been set to QA automatically):



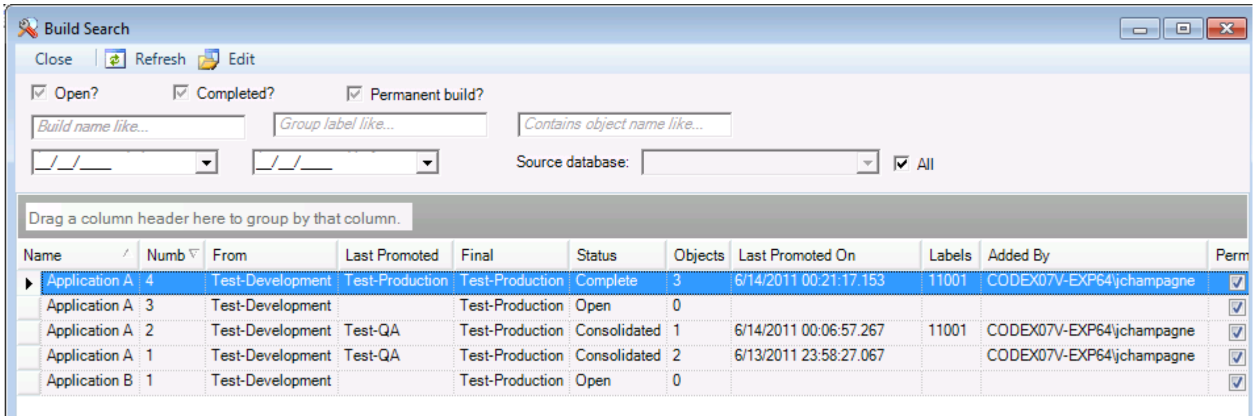
Objects... 3 item(s)

Selected	Hide	Ignore	Diff	Name	Type	SourceUpdate	DestUpdate	SourceModB
☑	□	□	!	uf_GetSomeValue	FN	13-Jun-11 23:54	13-Jun-11 22:29	CODEX07V-EXP64\jcl
☑	□	□	!	up_Procedure_AppA_1	P	13-Jun-11 23:54	13-Jun-11 22:29	CODEX07V-EXP64\jcl
☑	□	□	!	up_Procedure_AppA_2	P	14-Jun-11 00:06	13-Jun-11 22:29	CODEX07V-EXP64\jcl

Source: Test-QA
Target: Test-Production

Objects
☑ Tables ☑ Views
☑ Stored Procedures ☑ Triggers

After running the change script and confirming the promotion, we'd see this in a subsequent build search:



☑ Open? ☑ Completed? ☑ Permanent build?

Build name like... | Group label like... | Contains object name like...

_/_/___ | _/_/___ | Source database: | ☑ All

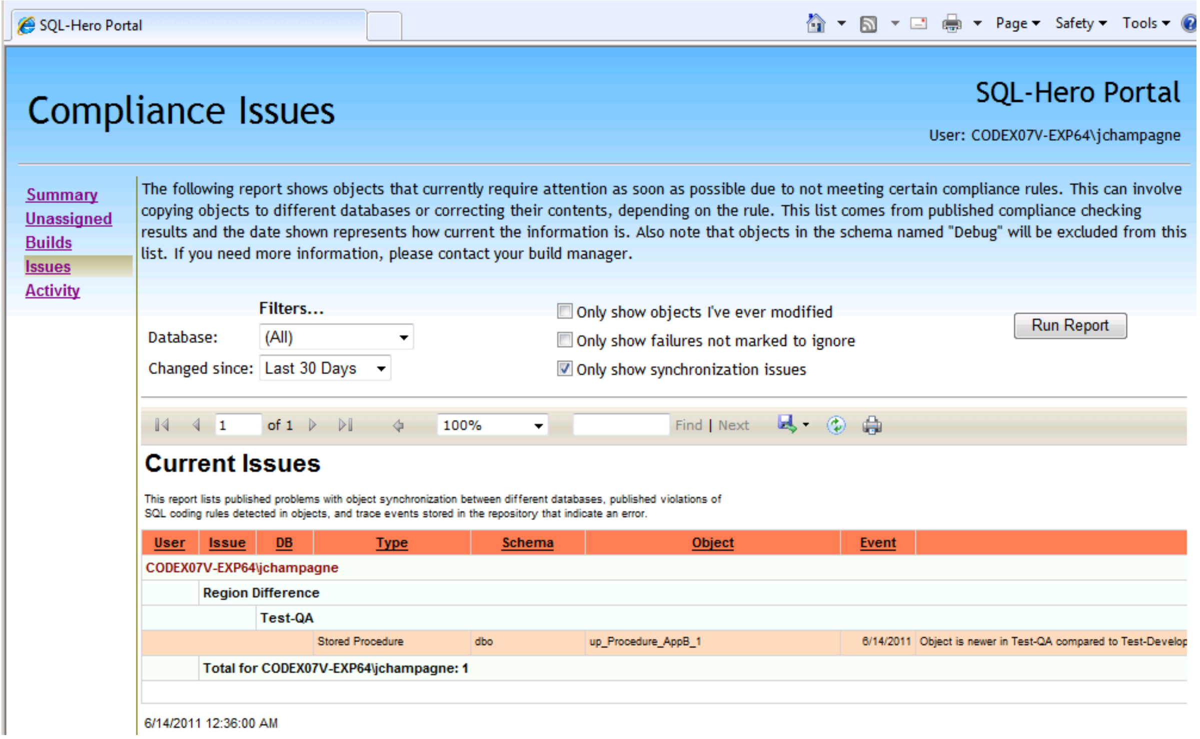
Drag a column header here to group by that column.

Name	Numb	From	Last Promoted	Final	Status	Objects	Last Promoted On	Labels	Added By	Perm
Application A	4	Test-Development	Test-Production	Test-Production	Complete	3	6/14/2011 00:21:17.153	11001	CODEX07V-EXP64\jchampagne	☑
Application A	3	Test-Development		Test-Production	Open	0				☑
Application A	2	Test-Development	Test-QA	Test-Production	Consolidated	1	6/14/2011 00:06:57.267	11001	CODEX07V-EXP64\jchampagne	☑
Application A	1	Test-Development	Test-QA	Test-Production	Consolidated	2	6/13/2011 23:58:27.067		CODEX07V-EXP64\jchampagne	☑
Application B	1	Test-Development		Test-Production	Open	0				☑

Note that the status of #4 is now "Complete". Build #3 is still open and available to support on-going development.

Other Portal Reporting

The web site includes a menu option for "Issues". This report shows a combination of published region difference failures, published compliance check failures, and collected trace records that indicate an error condition. In the example below, we've made a change in a procedure in QA without ensuring it was updated in development as well:



Compliance Issues

SQL-Hero Portal

User: CODEX07V-EXP64\jchampagne

The following report shows objects that currently require attention as soon as possible due to not meeting certain compliance rules. This can involve copying objects to different databases or correcting their contents, depending on the rule. This list comes from published compliance checking results and the date shown represents how current the information is. Also note that objects in the schema named "Debug" will be excluded from this list. If you need more information, please contact your build manager.

Filters...

Database: (All)

Changed since: Last 30 Days

□ Only show objects I've ever modified

□ Only show failures not marked to ignore

☑ Only show synchronization issues

Current Issues

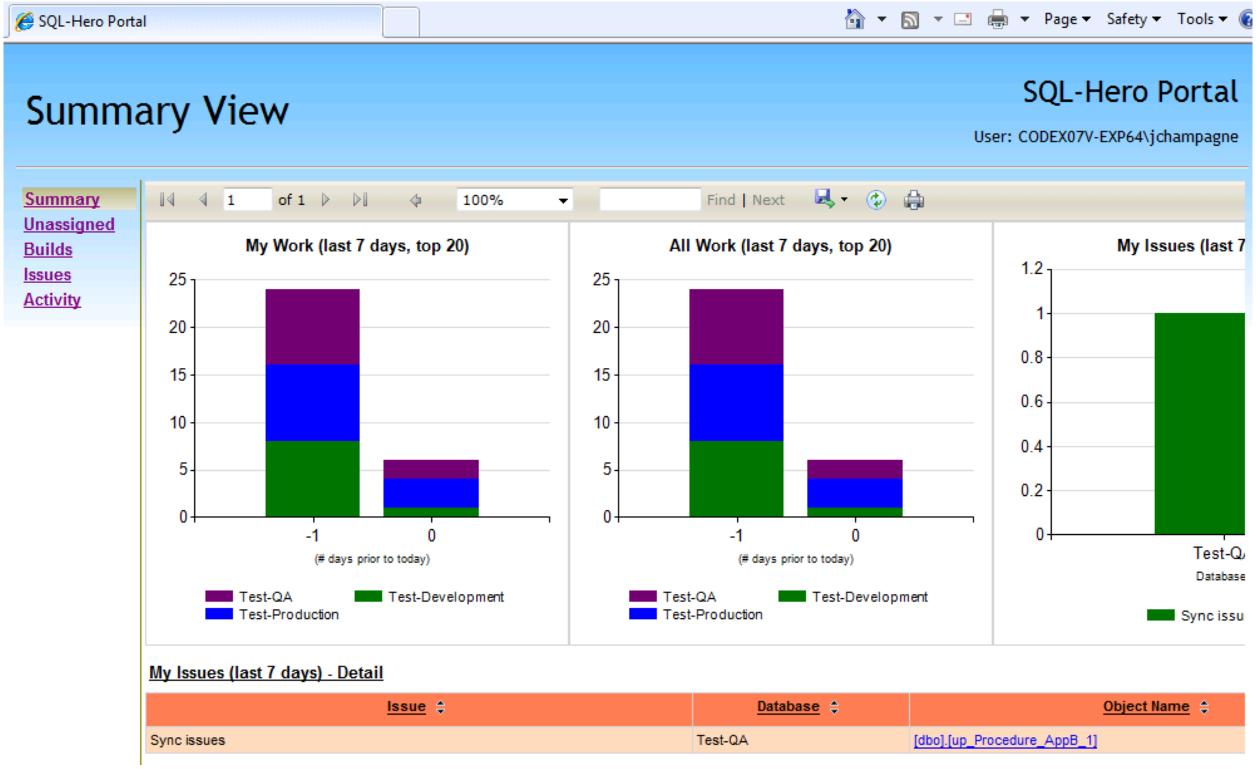
This report lists published problems with object synchronization between different databases, published violations of SQL coding rules detected in objects, and trace events stored in the repository that indicate an error.

User	Issue	DB	Type	Schema	Object	Event	
CODEX07V-EXP64\jchampagne							
	Region Difference						
		Test-QA					
			Stored Procedure	dbo	up_Procedure_AppB_1	6/14/2011	Object is newer in Test-QA compared to Test-Develop
	Total for CODEX07V-EXP64\jchampagne: 1						

6/14/2011 12:36:00 AM

This report is able to distinguish "legitimate differences" using SQL-Hero audit tracking – meaning one needs to be using DML tracking on any databases which participate in region difference checking. Source control integration also comes into play in deriving an accurate picture, so if TFS is being used, the TFS plug-in for SQL-Hero should be enabled. Note that the report is not performing a real-time analysis for compliance issues: it's looking at *published* issues. One typically schedules compliance checking with a frequency that depends on expected object turn-over for a given database – this topic is covered in another whitepaper.

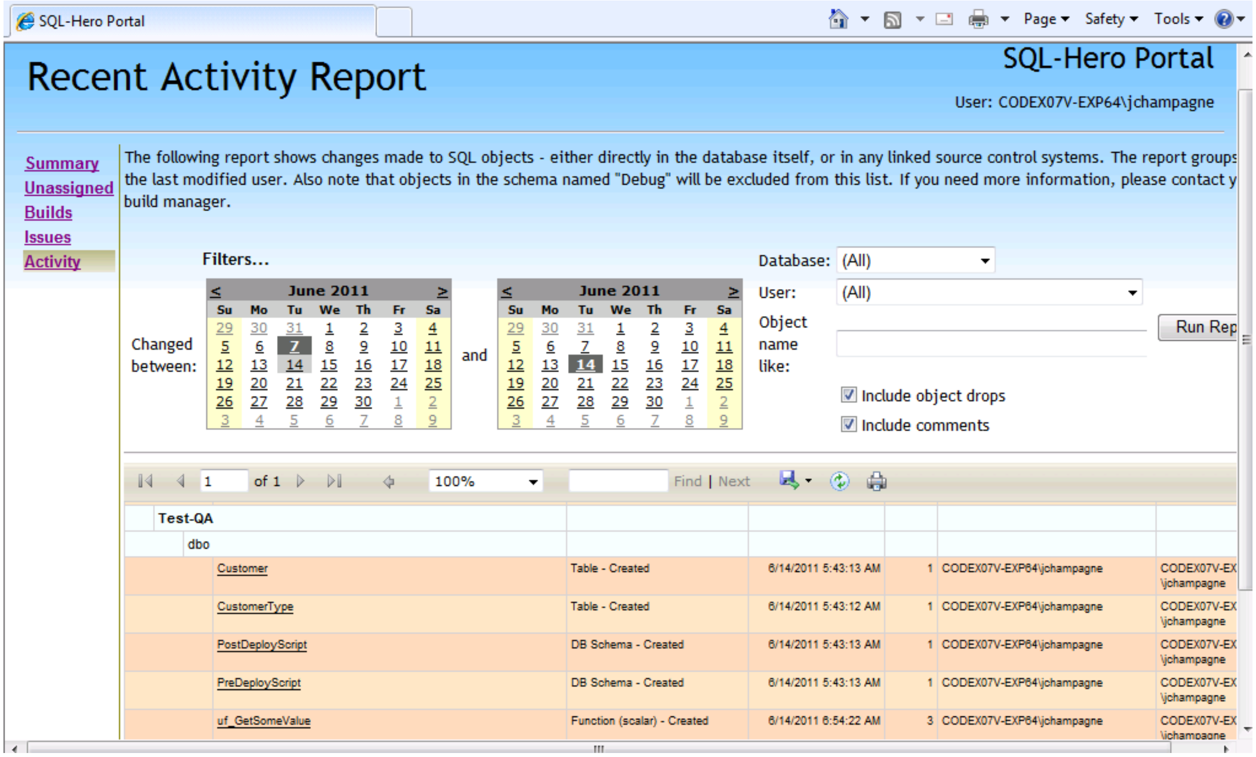
The Summary report is a quick view of some development metrics, filtered to the logged in user:



My Issues (last 7 days) - Detail

Issue	Database	Object Name
Sync issues	Test-QA	[dbo].[up_Procedure_AppB_1]

The Activity report lets you query the object change log using a date range and some additional filter parameters. The report groups by "last modified" user and a description of the last activity is included (e.g. "Created" versus "Updated"). This is the same report that's available in the SQL-Hero client tool on the History window. Note that actions from external source control systems can be included here as well to give a complete and consolidated picture of change.



Recent Activity Report

SQL-Hero Portal

User: CODEX07V-EXP64\jchampagne

The following report shows changes made to SQL objects - either directly in the database itself, or in any linked source control systems. The report groups the last modified user. Also note that objects in the schema named "Debug" will be excluded from this list. If you need more information, please contact y build manager.

Filters...

Changed between:

<			June 2011			>
Su	Mo	Tu	We	Th	Fr	Sa
29	30	31	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	1	2
3	4	5	6	7	8	9

and

<			June 2011			>
Su	Mo	Tu	We	Th	Fr	Sa
29	30	31	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	1	2
3	4	5	6	7	8	9

Database: (All)

User: (All)

Object name like:

☑ Include object drops

☑ Include comments

Test-QA					
dbo					
Customer	Table - Created	6/14/2011 5:43:13 AM	1	CODEX07V-EXP64\jchampagne	CODEX07V-EX \jchampagne
CustomerType	Table - Created	6/14/2011 5:43:12 AM	1	CODEX07V-EXP64\jchampagne	CODEX07V-EX \jchampagne
PostDeployScript	DB Schema - Created	6/14/2011 5:43:13 AM	1	CODEX07V-EXP64\jchampagne	CODEX07V-EX \jchampagne
PreDeployScript	DB Schema - Created	6/14/2011 5:43:13 AM	1	CODEX07V-EXP64\jchampagne	CODEX07V-EX \jchampagne
uf_GetSomeValue	Function (scalar) - Created	6/14/2011 6:54:22 AM	3	CODEX07V-EXP64\jchampagne	CODEX07V-EX \jchampagne