

# Database Feed of Timesheet Data

## **DISCLAIMER**

The populating of timesheet data directly in the database is **NOT** supported by EPK Group. As this data avoids all of the validations and limit checks build into the user interface it may contain unexpected values. If you encounter problems in EPK-Suite that may be attributable to incorrect data we will ask for repro steps in a database containing data entered through the UI before we will act upon it under a support agreement.

<b>DATABASE FEED OF TIMESHEET DATA .....</b>	<b>1</b>
DISCLAIMER .....	1
REQUIREMENTS .....	1
DATABASE SCHEMA .....	2
<i>EPK_TS_TIMESHEETS</i> .....	2
<i>EPK_TS_CHARGES</i> .....	3
<i>EPK_TS_ACTUALHOURS</i> .....	3
<i>EPK_PERIODS</i> .....	3
<i>EPK_PROJ_TASKS</i> .....	4
EXAMPLE .....	4
Entering Data through UI.....	4
Entering Timesheet Data into Database.....	8
Timesheet .....	8
Timesheet Charge Rows.....	9
Hours Reported per Day .....	9

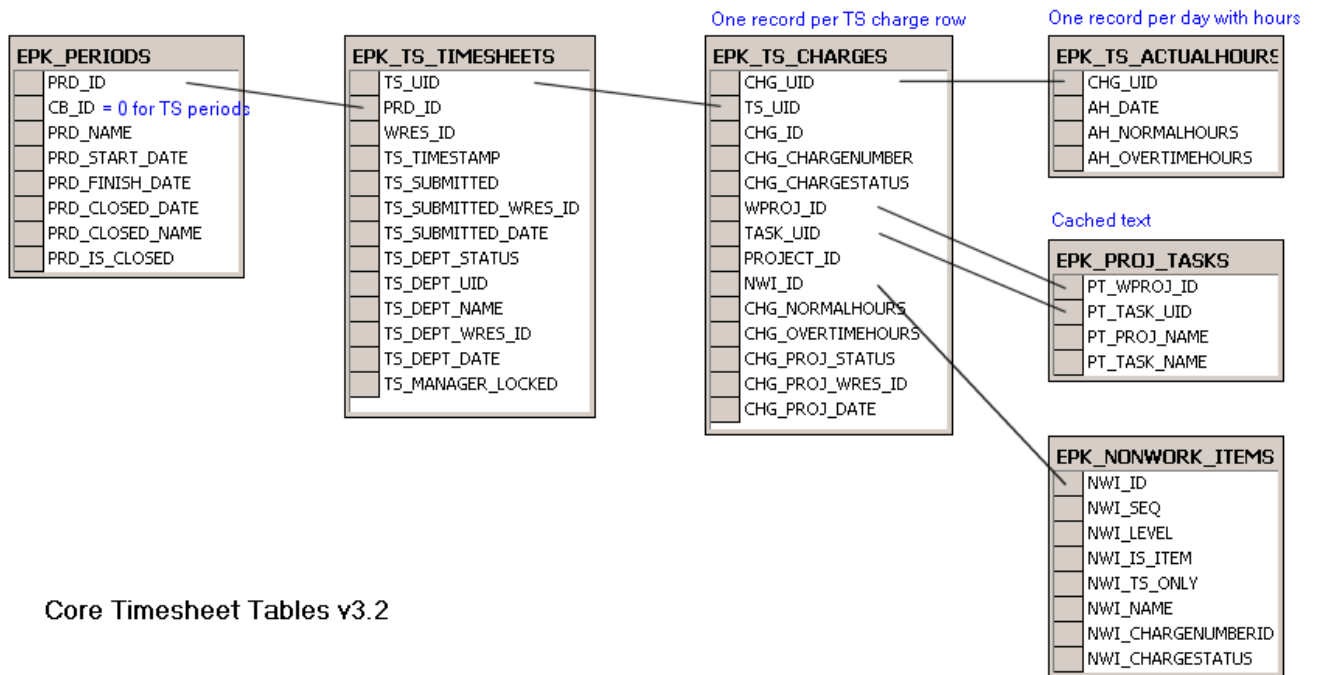
## **Requirements**

As well as period actuals, managers often want to know actuals to date or project to date numbers. When migrating to Timesheets in EPK-Suite, past data is in a separate system/database. Some sites have transferred past timesheet actuals into the EPK tables in the following way:

- defined the first period in the timesheet calendar as an exception to the usual timesheet period length. For launch of weekly timesheets on Jan 2, 2005 that prior period might be named 2004 and cover the whole prior year. Or just be called Prior.
- this predefined timesheet period can be Closed by the Timesheet Administrator so that no timesheet can be opened for editing. This avoids triggering the client software to validate the data that cause problems on save.
- data is written to the database tables down to the day level if it is to be displayed in View Timesheet Actuals pages
- careful testing is required to ensure that the database feed has not compromised system functionality

## Database Schema

The core database tables involved are detailed below.



Core Timesheet Tables v3.2

More complete information on EPK database tables can be found in the EPK-Suite Database Tables.pdf document installed to the application server during setup or downloaded from the [www.epkgroup.com](http://www.epkgroup.com) web site.

### EPK\_TS\_TIMESHEETS

A record for each timesheet with charges for any period

Field	Type	Content
<b>TS_UID</b>	int	identity
<b>PRD_ID</b>	int	
<b>WRES_ID</b>	int	Timesheet owner
<b>TS_TIMESTAMP</b>	datetime	
<b>TS_SUBMITTED</b>	tinyint	
<b>TS_SUBMITTED_WRES_ID</b>	int	Different from WRES_ID if submitted by delegate
<b>TS_SUBMITTED_DATE</b>	datetime	
<b>TS_DEPT_STATUS</b>	tinyint	Dept approved: 4 = Rejected, 5 = Yes
<b>TS_DEPT_WRES_ID</b>	int	Approver
<b>TS_DEPT_DATE</b>	datetime	
<b>TS_DEPT_UID</b>	int	
<b>TS_DEPT_NAME</b>	nvarchar(255)	

<b>TS_MANAGER_LOCKED</b>	tinyint
--------------------------	---------

### EPK\_TS\_CHARGES

A record for each timesheet charge in any period

Field	Type	Content
<b>CHG_UID</b>	int	identity
<b>TS_UID</b>	int	
<b>CHG_ID</b>	int	Established sequence of lines on timesheet
<b>WPROJ_ID</b>	int	MSP Project ID, Null for a portfolio Item
<b>TASK_UID</b>	int	MSP Task Unique ID, Null for Project level or Portfolio charges
<b>PROJECT_ID</b>	int	Portfolio Item Unique ID, Null for charges to MSP Projects
<b>NWI_ID</b>	int	Non-Work Item Unique ID, Null for charges to MSP Projects and to Portfolio Items
<b>CHG_CHARGENUMBER</b>	nvarchar(255)	
<b>CHG_NORMALHOURS</b>	int	
<b>CHG_OVERTIMEHOURS</b>	int	
<b>CHG_PROJ_STATUS</b>	tinyint	9 = Approved by PM
<b>CHG_PROJ_WRES_ID</b>	int	
<b>CHG_PROJ_DATE</b>	datetime	
<b>CHG_CHARGESTATUS</b>	nvarchar(255)	

### EPK\_TS\_ACTUALHOURS

A record for each timesheet charge on any particular day

Field	Type	Content
<b>CHG_UID</b>	int	
<b>AH_DATE</b>	datetime	
<b>AH_NORMALHOURS</b>	int	
<b>AH_OVERTIMEHOURS</b>	int	

### EPK\_PERIODS

A record for each fiscal calendar

Field	Type	Content
<b>PRD_ID</b>	int	Period ID
<b>CB_ID</b>	int	Cost Breakdown ID (0 = timesheet)

		periods)
<b>PRD_NAME</b>	nvarchar(255)	Period Name
<b>PRD_START_DATE</b>	datetime	
<b>PRD_FINISH_DATE</b>	datetime	
<b>PRD_CLOSED_DATE</b>	datetime	Date that this period was closed (timesheet periods only)
<b>PRD_CLOSED_NAME</b>	nvarchar(255)	Name of user who closed this period
<b>PRD_IS_CLOSED</b>	tinyint	Is this period closed (0 or 1)

### EPK\_PROJ\_TASKS

A record for each field unique combination of Project and Task used in the timesheet charge table (as a reference in case of changes/deletions)

Field	Type	Content
<b>PT_WPROJ_ID</b>	int	WPROJ_ID of used project
<b>PT_TASK_UID</b>	int	TASK_UID of used task (0 for project charge)
<b>PT_PROJ_NAME</b>	nvarchar(255)	
<b>PT_TASK_NAME</b>	nvarchar(255)	

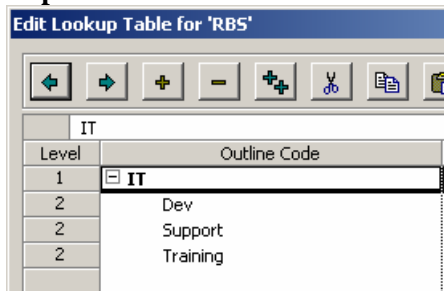
### Example

You will have portfolio item, resource, department and other data already but for the purposes of this example we'll start with an empty database. After entering data manually we'll look at how it is stored internally so that we can pick up the IDs needed to populate the timesheets.

### Entering Data through UI

Let's set up some simple data for use in example:

**Department** as EROC30 or RBS



### Assign resources to Departments

		Resource Name	Type	RBS
1		Abby	Work	IT.Dev
2		Bob	Work	IT.Support
3		Carol	Work	IT.Training
4		Contractor ABC	Work	
5		David	Work	IT

SQL Server Enterprise Manager - [Data in Table 'MSP\_CODE\_FIELDS' in 'ProjectServer' o...]

```

SELECT MWR.WRES_ID, MP.RES_NAME, MCF.CODE_UID
FROM MSP_CODE_FIELDS MCF INNER JOIN
MSP_RESOURCES MP ON MCF.CODE_REF_UID = MP.RES_UID AND MP.PROJ_ID = 1 INNER JOIN
MSP_WEB_RESOURCES MWR ON MP.RES_UID = MWR.RES_EUID
    
```

WRES_ID	RES_NAME	CODE_UID
101	Abby	11
102	Bob	12
103	Carol	13
104	David	10

**And timesheet approvers**  
**Configure Departments**

Save

Department	Manager
IT	David
Dev	David
Support	David
Training	David

SQL Server Enterprise Manager - [Data in Table 'EPK\_TS\_APPROVERS' .

CODE_UID	WRES_ID	OCA_TEXT1	OCA_TEXT2
10	104		
11	104		
12	104		
13	104		
*			

**Setup charge numbers**

**Portfolio Custom Field: Charge Number**

Use a Lookup list for this field:

Restrict field entries to those in the list:

Restrict selection to Leaf entries in list:

Using list from field: Project CN (Enterprise Project Outline Code30)

Admin
99910
99920
Eng
66610
66620
IT
44410
44420

SQL Server Enterprise Manager - [Data in Table 'MSP\_OUTLINE\_CODES' in 'ProjectServer' on ...]

```

SELECT *
FROM MSP_OUTLINE_CODES
WHERE (OC_FIELD_ID = 188744618)
    
```

RESERVED_DATA	PROJ_ID	CODE_UID	OC_PARENT	OC_FIELD_ID	OC_NAME
0	5	1	0	188744618	Admin
0	5	2	1	188744618	99910
0	5	3	1	188744618	99920
0	5	4	0	188744618	IT
0	5	5	4	188744618	44410
0	5	6	4	188744618	44420
0	5	7	0	188744618	Eng
0	5	8	7	188744618	66610
0	5	9	7	188744618	66620

Timesheet **periods** (note that Prior is closed)

### Manage Timesheet Periods

Save

Periods

Period	Name	Start	Finish	Status
1	Prior	1/2/2005	5/28/2005	Closed on 6/3/2005 by Administrator
2	wc 5/29	5/29/2005	6/4/2005	not used
3	wc 6/05	6/5/2005	6/11/2005	not used
4	wc 6/12	6/12/2005	6/18/2005	not used

SQL Server Enterprise Manager - [Data in Table 'EPK\_PERIODS' in 'ProjectServer' on 'BOLFORDLAP']

PRD_ID	CB_ID	PRD_NAME	PRD_START_DATE	PRD_FINISH_DATE	PRD_CLOSED_DATE	PRD_CLOSED_NAME	PRD_IS_CLOSED
1	0	Prior	1/2/2005	5/28/2005	6/3/2005 9:01:28 AM	Administrator	1
2	0	wc 5/29	5/29/2005	6/4/2005	<NULL>	<NULL>	0
3	0	wc 6/05	6/5/2005	6/11/2005	<NULL>	<NULL>	0
4	0	wc 6/12	6/12/2005	6/18/2005	<NULL>	<NULL>	0

### Add some **portfolio** Items

Name	Charge Number	Charge Status	2005												2006	
			J	F	M	A	M	J	J	A	S	O	N	D	J	F
Admin PI	Admin.99910	2 - Open, Charge to PI or W/s	[Gantt bar from Jan to May]													
Complex Project	Eng.66610	1 - Open, Charge to PI only	[Gantt bar from May to Oct]													
Operations PI	IT.44410	1 - Open, Charge to PI only	[Gantt bar from Jan to Dec]													
Support PI	IT.44420	2 - Open, Charge to PI or W/s	[Gantt bar from Jan to Dec]													

PROJECT_ID	PROJECT_NAME	PROJECT_START_DATE	PROJECT_FINISH_DATE	PROJECT_STATUS	PROJECT_STAGE
1	Admin PI	1/2/2005	12/31/2005	<NULL>	1
2	Support PI	1/2/2005	12/31/2005	<NULL>	1
3	Operations PI	1/2/2005	12/31/2005	<NULL>	1
4	Complex Project	3/7/2005	9/23/2005	<NULL>	1

Some with **Work Items**

Name	Charge Number	Charge Status	2005	2006
1 Admin PI	99910	2 - Open, Charge to PI or WIs	J A J O J	J
2 Meetings	99910	1 - Open		
3 Mentoring	99910	1 - Open		
4 HR	99910	1 - Open		
5 PR	99910	1 - Open		

PROJECT_ID	WORKITEM_ID	WORKITEM_NAME	WORKITEM_CHARGENUMBERID	WORKITEM_CHARGESTATUS
1	1	Meetings	2	1 - Open
1	2	Mentoring	2	1 - Open
1	3	HR	2	1 - Open
1	4	PR	2	1 - Open
2	1	Support	6	1 - Open
2	2	ERs	6	1 - Open
2	3	Operations	6	1 - Open

**Non-work Items**

**Manage Non-Work Items**

Save

← → ↕ ↶ ↷

Name	Charge Number	Charge Status	TS Only
Public Holiday	99910	Open	<input type="checkbox"/>
Vacation	99910	Open	<input type="checkbox"/>
Sick	99920	Open	<input checked="" type="checkbox"/>
Civic Duty	99910	Open	<input type="checkbox"/>
Training	99910	Open	<input type="checkbox"/>

NWI_ID	NWI_IS_ITEM	NWI_TS_ONLY	NWI_NAME	NWI_CHARGENUMBERID	NWI_CHARGESTATUS
1	1	0	Public Holiday	2	1
2	1	0	Vacation	2	1
3	1	1	Sick	3	1
4	1	0	Civic Duty	2	1
5	1	0	Training	2	1

## Entering Timesheet Data into Database

Migrated data from the old timesheet system will not be aware of the IDs in EPK so it you will need to convert it. Let's follow some data for Abby, whose IDs are:

WRES\_ID = 101  
 Dept IT.Dev, UID = 11  
 Approver David WRES\_ID = 104

PI	WI	ID	Charge No	Charge Status	Hours
Support		PI = 2	44420	2	345
Admin	Meetings	PI = 1, WI = 1	99910	2	24
Vacation		NWI = 3	99920		40

## Timesheet

### SQL script

```
insert into EPK_TS_TIMESHEETS (PRD_ID,WRES_ID, TS_TIMESTAMP, TS_SUBMITTED,
TS_SUBMITTED_WRES_ID, TS_SUBMITTED_DATE, TS_DEPT_STATUS, TS_DEPT_UID, TS_DEPT_NAME,
TS_DEPT_WRES_ID, TS_DEPT_DATE, TS_MANAGER_LOCKED)
values (1, 101, 0, 1, 1, GETDATE(), 5, 11, 'Dev', 104, GETDATE(), 1)
```

## Timesheet Register - Prior



Ok	Per	Dept	Proj	Name	WI	Charge Number	Hours	Total Hours
		✓	Key	Dev				
		✓	Key	Abby				
			+	IT				
			+	Support				
			+	Training				

## Timesheet Charge Rows

### SQL

```
SET IDENTITY_INSERT EPK_TS_CHARGES ON
```

```
insert into EPK_TS_CHARGES (CHG_UID, TS_UID, CHG_ID, CHG_CHARGENUMBER, CHG_CHARGESTATUS,
WPROJ_ID, TASK_UID, PROJECT_ID, NWI_ID, CHG_NORMALHOURS, CHG_PROJ_STATUS,
CHG_PROJ_WRES_ID, CHG_PROJ_DATE)
values (1, 1, 1, 44420, 2, NULL, NULL, 2, NULL, 345*60000, 9, 10, GETDATE())
```

```
insert into EPK_TS_CHARGES (CHG_UID, TS_UID, CHG_ID, CHG_CHARGENUMBER, CHG_CHARGESTATUS,
WPROJ_ID, TASK_UID, PROJECT_ID, NWI_ID, CHG_NORMALHOURS, CHG_PROJ_STATUS,
CHG_PROJ_WRES_ID, CHG_PROJ_DATE)
values (2, 1, 2, 99910, 2, NULL, 1, 1, NULL, 24*60000, 9, 10, GETDATE())
```

```
insert into EPK_TS_CHARGES (CHG_UID, TS_UID, CHG_ID, CHG_CHARGENUMBER, CHG_CHARGESTATUS,
WPROJ_ID, TASK_UID, PROJECT_ID, NWI_ID, CHG_NORMALHOURS, CHG_PROJ_STATUS,
CHG_PROJ_WRES_ID, CHG_PROJ_DATE)
values (3, 1, 3, 99920, 1, NULL, NULL, NULL, 2, 40*60000, 9, NULL, GETDATE())
```

```
SET IDENTITY_INSERT EPK_TS_CHARGES OFF
```

## Timesheet Register - Prior



Options										
1/2/2005 - 5/28/2005   Open Period   Expand All   Expand Row   Collapse All   [Refresh]   [Print]										
	Ok	Per	Dept	Proj	Name	Wl	Charge Number	Hours	Total Hours	
			✓	✓	Dev			409		409
			✓	✓	Abby			409		409
				✓	Support PI		44420	345		345
				✓	Admin PI	Meetings	99910	24		24
				✓	Vacation		99920	40		40
				+	IT					
				+	Support					
				+	Training					

## Hours Reported per Day

Rather than spread the hours, we enter the whole period's hours in one day as we know that as the period is closed the validations set by Limits will highlight out-of-scope values but will have no other effect.

### SQL

```
insert into EPK_TS_ACTUALHOURS (CHG_UID, AH_DATE, AH_NORMALHOURS)
values (1, '2005/05/27', 345*60000)
```

```
insert into EPK_TS_ACTUALHOURS (CHG_UID, AH_DATE, AH_NORMALHOURS)
values (2, '2005/05/27', 24*60000)
```

```
insert into EPK_TS_ACTUALHOURS (CHG_UID, AH_DATE, AH_NORMALHOURS)
values (3, '2005/05/27', 40*60000)
```

The **Timesheet Register** display is unchanged but now that data by day has been added **View Timesheet Actuals** can be used

### Timesheet Actuals

Expand All   Expand Row   Collapse All   [Print] [Refresh] [Save]								
Filters Options								
	Name	Resource Name	Project Name	Task Name	Charge No.	Row Type	Prior	Totals
	IT						409	409
	Dev						409	409
		Abby	Vacation		99920	Non-Work Item	40	40
		Abby	Admin PI	Meetings	99910	Portfolio Item	24	24
		Abby	Support PI		44420	Portfolio Item	345	345
	Support							
	Training							

**Timesheet User** – the 409 hour total is colored red as it is above the limits but as there is no editing or saving there are no consequences.

### Dev Timesheet - Abby



**Timesheet - enter actual hours worked**

1/2/2005 - 5/28/2005 | Non-Work | My Work | Go To | Status | [Up] | [Down] | [Refresh] | [Print]

	Project Name	Task Name	Charge Number	Mon	Tue	Wed	Thu	Fri	TOTALS
	Support PI		44420						345
	Admin PI	Meetings	99910						24
	Vacation		99920						40
			TOTALS						409

**Progressor - enter status information**

**Explorer - view and select items**

Options Filter Search

Add Project | Expand Row | [Print] | [Refresh]

	Name	Charge Number	Resource Names	2005											
				J	F	M	A	M	J	J	A	S	O	N	D
	Support PI	44420		[Gantt bars]											
	Support	44420		[Gantt bars]											
	ERs	44420		[Gantt bars]											
	Operations	44420		[Gantt bars]											

Department Manager View – the Portfolio Item/Project Manager’s view is similar.

**Department Approval - David**



1/2/2005 - 5/28/2005								
Per	Dept	Proj	Name	Description	Charge Number	Hours	Total Hours	
	✓	✓	[-] Dev			409	409	
🕒	✓	✓	[-] Abby			409	409	
🕒		✓	Support PI		44420	345	345	
🕒		✓	Admin PI	Meetings	99910	24	24	
🕒		✓	Vacation		99920	40	40	
			[-] IT					
			David					
			[-] Support					
			Bob					
			[-] Training					
			Carol					

06/03/2005     Drafted  
 06/07/2005     Published