

## Sys-Events

1	Was sind SysEvents? .....	1
2	tsSysEvents.....	2
3	Standard-Events aus easyFramework.....	2

### 1 Was sind SysEvents?

SysEvents sind Ereignisse die ausgelöst werden, und auf die Sie reagieren können. Das heißt, dass Sie die Möglichkeit haben, dass Anwender Ihres Produktes bequem auf bestimmte Ereignisse reagieren können und entsprechende Vorgänge starten können.

Außerdem haben Sie eine Übersicht über das, was in Ihrer Anwendung geschieht.

Die Entitäten verwenden z.B. die SysEvents standardmäßig um auf ein Löschen-Ereignis zu reagieren. Es werden dann sämtliche Untergeordnete Entitäten zuvor gelöscht.

Sie können beliebige benutzerdefinierte Ereignisse definieren und in eigenen Assemblies ablegen und zu Ihrer Anwendung einfach dazulegen.

Dreh- und Angelpunkt ist die Tabelle tsSysEvents die zunächst beschrieben wird.

## 2 tsSysEvents

In dieser Tabelle wird auf ein bestimmtes Ereignis reagiert. Im easyFramework sind einige Ereignisse standardmäßig vorgegeben. Sie bitte unter [Standard-Events aus easyFramework](#) nach.

### Tabellenfelder mit Beispielwerten

Sys_category	Sys_name	Sys_ActionType	Sys_Index	Sys_Task	Sys_Assembly
EntityDelete	Orders	SQL	100	DELETE FROM tdOrderItems WHERE OrderID=\$1	<NULL>
EntityInsert	Orders	CLASS	100	easyFrameWork.Project.EntityEventHandler	mypEventHandler

### 2.1 Feld Sys\_Category

Kann beliebig definiert werden. Jede selbst geschriebene Komponente kann eigene Kategorien aufrufen.

Das Entitätenmodell implementiert die Ereignisse **EntityDelete**, **EntityInsert** und **EntityUpdate**.

### 2.2 Feld Sys\_Name

Beliebig definierter Name. Meist eine genauere Beschreibung, z.B. bei EntityDelete: Name der Entity

### 2.3 Feld Sys\_ActionType

Zwei Werte möglich:

1. SQL – direkter SQL-Befehl, z.B: „EXEC spXXX“, „DELETE FROM ...“, „INSERT INTO ...
2. CLASS – Klassenname einer .NET-Klasse; In Assembly muss dann der Name der Assembly angegeben werden, in der sich die Klasse befindet.

Die Klasse muss die Schnittstelle ISysEvents implementieren.

### 2.4 Feld Sys\_Index

Negative Werte werden vor dem eigentlich Event ausgeführt. Positive Werte danach. Der Indexwert 0 wird NACH dem eigentlichen Event ausgeführt.

### 2.5 Feld Sys\_Task

Bei ActionType = SQL die SQL-Anweisung. Bei ActionType = CLASS die .NET-Klasse.

Anmerkung: bei Entitäten wird in Task als SQL der Parameter \$1 mit dem Schlüsselwert der Entität ersetzt.

## 2.6 Feld Sys\_Assembly

Nur relevant, wenn ActionType = class

## 3 Standard-Events aus easyFramework

Für Entitäten gelten folgende Ereignisse:

- **EntityDelete** - wird beim Löschen einer Entität gefeuert
- **EntityUpdate** – wird beim Update gefeuert (auch bei Neuanlage)
- **EntityCreate** – wird nur bei Neuanlage gefeuert

Anmerkung: bei Entitäten wird in tsSysEvents.Task als SQL der Parameter \$1 mit dem Schlüsselwert der Entität ersetzt.