

# Transparenz für alle

mogular GmbH | Newsletter März 2012

Statusmonitore werden in Unternehmen immer beliebter. Hauptverantwortlich dafür ist der sogenannte *Hawthorne*-Effekt, wonach die Beobachtung oder der Vergleich mit anderen und mit einer Zielvorgabe sowohl die Motivation als auch die Leistungsfähigkeit der am Prozess beteiligten Personen steigert. Ein Statusmonitor zeigt allen Mitarbeitern in der Produktion oder im Lager an, wie viele Aufträge aktuell anstehen, wie viele im Rückstand sind, wie gut die aktuelle Liefertreue ist – KPIs eben. Neben der reinen Motivation haben Statusmonitore auch operative Funktion - beispielsweise können Sie Auskunft geben, wo aktuell ein nicht abgearbeiteter Arbeitsvorrat zur Verfügung steht. Hinzu kommt, dass Bildschirme für graphische Großanzeigen heute keine große Investition mehr darstellen.

## Statusmonitor versus Leitstand

Häufig bieten Standardsysteme Leitstände an – die jedoch Benutzerinteraktion fordern. Selbstaktualisierende Statusmonitore sind die Ausnahme – schon weil die Anforderungen sehr kundenindividuell sind. Eine individuelle Anpassung ist mit hohen Kosten verbunden, außerdem müssen alle Kennzahlen als Kundenvorgabe vor Entwicklung definiert werden. Gerade bei Statusmonitoren ist aber ein iterativer Ansatz zu empfehlen, so dass aus der gewonnenen Erfahrung gelernt werden kann.

Schön wäre es hier einen Werkzeugkasten zu haben, mit dem man schnell und kreativ seinen eigenen Statusmonitor bauen kann – und das Ganze mit bereits vorhandenen Bordmitteln.

## Excel? Excel!

Excel hat sich seit der Markteinführung vor über 25 Jahren immer mehr zu *dem* Werkzeug für Auswertungen oder Kalkulationen entwickelt. Können Daten in Excel zur Verfügung gestellt werden, sind die Mitarbeiter selbstständig in der Lage diese mittels Pivot-Tabellen oder Diagrammen aufzubereiten um Schlüsse aus den Daten zu ziehen.

## Real Time Data – Daten in Echtzeit

Kaum beachtet wurde mit Einführung der Version 2002 das Real-Time-Data Feature von Excel. Einfach gesagt bietet Excel damit eine Funktion, von beliebigen Datenquellen (Datenbanken, WeBServices etc.) Zahlen in einem Spreadsheet anzuzeigen und automatisch zu aktualisieren, sofern sich an der Datengrundlage etwas geändert hat.

## Mehrere unabhängige Datenquellen

Seit der Version 2007 ist Excel auch in der Lage parallel mehrere unabhängige Aktualisierungen durchzuführen (durch die Einführung der *Multithreaded Recalculation*). Dadurch wird es möglich verschiedene Zahlen mehrerer Systeme gleichzeitig anzuzeigen und unabhängig auf der Anzeige aktualisieren zu lassen – ein einfaches Beispiel dafür sind internationale Börsendaten, die über Online-Dienste in einem Excel-Spreadsheet automatisch aktualisiert und dargestellt werden.

### Multithreading

Unter Multithreading versteht man die Möglichkeit eines Systems, quasi parallel und unabhängig voneinander Berechnungsaufgaben auszuführen – beispielsweise Aktualisierungen von mehreren Datenquellen. Unterstützt ein System kein MT bedeutet das, dass Berechnungen nacheinander ausgeführt werden – damit kann eine blockierende Berechnung (beispielsweise wenn eine Datenquelle gerade nicht erreichbar ist) nachfolgende Berechnungen blockieren.

## Statusmonitore in der Praxis

In einem mittelständischen produzierenden Unternehmen werden Produktionsaufträge in einem ERP-System angelegt. Entweder durch Kundenvorgabe oder berechnet aus dem Eingangsdatum erhalten Aufträge ein geplantes Fertigstellungsdatum. Somit kann dargestellt werden, wie viele Aufträge terminlich eingehalten werden können, wie viele schon an der Grenze und wie viele definitiv verspätet sind. In einem anhängigen Lager kann wiederum die Produktivität im Vergleich zu den letzten 30 Tagen dargestellt



werden – beispielsweise mittels der Anzahl der heute abgearbeiteten Bestellpositionen im Vergleich zu den letzten 30 Tagen zur selben Uhrzeit.

## Technik

Während der ursprüngliche RTD-Server noch viel C++ Implementierung und Registrierung von DLLs verlangte, kann heute mittels dem Open Source Project *Excel DNA* in C# entwickelt werden. Excel DNA agiert dabei als Excel AddIn mit vielen nützlichen Funktionen, u.a. wird die dynamische Übersetzung von C# Klassen unterstützt. RTD-Server werden automatisch registriert, so dass diese in Excel direkt zur Verfügung stehen.

### Unendliche Möglichkeiten in .NET

Dem Entwickler steht durch den Einsatz *Excel DNA* die gesamte Welt von C#, also inklusive aller Bibliotheken, zur Verfügung – von dem Zugriff auf SQL-Datenbanken mittels LINQ oder dem Abruf von WebServices via WCF bis hin zu Dateizugriffen, E-Mail-Funktionen oder SMS Versand.

## Ausblick

Excel RTD verbunden mit Excel DNA für einen Statusmonitor zu verwenden ist eine Anwendungsmöglichkeit – viele weitere Anwendungen lassen sich vorstellen. So kann der Statusmonitor erweitert werden zu einem Monitor für Mitarbeiter, der via Touch Screen verschiedene Informationen bereithält; oder via C# völlig neue Funktionen für Excel zur Verfügung gestellt werden.

### Beispiel RTD-Server: Aktuelle Uhrzeit

```
public class TimeServer : IRtdServer{

    private IRTDUpdateEvent callback;
    private Timer timer;
    private int topicId;

    #region IRtdServer Members

    public object ConnectData(int topicId, ref Array Strings,
                             ref bool GetNewValues){
        _topicId = topicId;
        _timer.Start();
        return GetTime();
    }

    public void DisconnectData(int topicId){
        _timer.Stop();
    }

    public int Heartbeat(){
        return 1;
    }

    public Array RefreshData(ref int topicCount){
        object[,] results = new object[2, 1];
        results[0, 0] = _topicId;
        results[1, 0] = GetTime();
        topicCount = 1;
        _timer.Start();
        return results;
    }

    public int ServerStart(IRTDUpdateEvent CallbackObject){
        _callback = CallbackObject;
        _timer = new Timer();
        _timer.Tick += Callback;
        _timer.Interval = 500;
        return 1;
    }

    public void ServerTerminate(){
        if (_timer!=null){
            _timer.Dispose();
            _timer = null;
        }
    }

    #endregion
}
```

## Excel Entwicklung ohne VBA

Mit Excel DNA muss man sich nicht mehr der eher unangenehmen Sprache VBA widmen – alles kann in C# abgewickelt werden. In der Welt von C# stehen natürlich ganz andere Möglichkeiten offen. So könnte in der Serverüberwachung mittels automatischen Pings auf Serverfarmen deren Heartbeat getestet und die Ergebnisse dargestellt werden. Im Fehlerfall könnten sogar automatisch SMS oder E-Mails versandt werden.

Nach Zawinski's Law of Software Envelopment wird es wohl damit enden: *Every program attempts to expand until it can read mail. Those programs which cannot so expand are replaced by ones which can.*

## Zum Weiterlesen und Ausprobieren

- Hawthorne Effekt | <http://de.wikipedia.org/wiki/Hawthorne-Effekt>
- Andon Displays | <http://de.wikipedia.org/wiki/Andon>
- Zawinski's Law | [http://en.wikipedia.org/wiki/Jamie\\_Zawinski#Zawinski.27s\\_law\\_of\\_software\\_envelopment](http://en.wikipedia.org/wiki/Jamie_Zawinski#Zawinski.27s_law_of_software_envelopment)
- Excel RTD | [http://msdn.microsoft.com/en-us/library/aa140059\(v=office.10\).aspx](http://msdn.microsoft.com/en-us/library/aa140059(v=office.10).aspx)
- Excel DNA | <http://excel-dna.net>

Fragen Sie doch einfach mal bei uns nach, wenn Sie sich mit dem Thema Statusmonitore beschäftigen.



A handwritten signature in black ink, appearing to read "H. Gubbels". The signature is fluid and cursive.



A handwritten signature in black ink, appearing to read "A. Moosbrugger". The signature is fluid and cursive.

Büroadresse:  
mogular GmbH  
Ossietzkystr. 8  
70174 Stuttgart

[www.mogular.com](http://www.mogular.com)  
[mail@mogular.com](mailto:mail@mogular.com)  
Tel +49 (0) 711 / 217 28 78 - 0  
Fax +49 (0) 711 / 217 28 78 - 9

Firmensitz:  
mogular GmbH  
Zamenhofstr. 64  
70197 Stuttgart

Stuttgart Volksbank e.G.  
BLZ 600 901 00  
Konto 115 075 003  
IBAN DE79600901000115075003

Registergericht: Stuttgart HRB 738119  
Gerichtsstand: Stuttgart  
Geschäftsführer: Holger Gubbels,  
Andreas Moosbrugger