Unit-Verwaltung

Aufgabe

Units sind programmtechnische Einheiten, die dynamisch in das CBA-Framework eingebunden werden. Die Units implementieren Methoden, die die Funktionalität realisieren. Sie stellen neben Konfiguration und deskriptiver Beschreibung den dritten Weg zur Implementierung von Software-Anwendungen dar - die programmtechnische Implementierung. Die CBA-Units sind dabei selbst in hohem Maße parametrierbar, sodass die Implementierung wieder verwendbarer programmtechnischer Einheiten nicht nur möglich, sondern auch sehr gut unterstützt wird.

Das Unit-Konzept ist en Kernstück von CBA und wurde auf der Basis eines bewährten Service-Design-Pattern für service-orientierte Anwendungen konzipiert. Seine programmtechnische Umsetzung mit Hilfe des CBA-Frameworks war ein konseqünte Fortführung dieses Gedankens. Ziel ist es, kleine, methodisch sauber entwickelbare und effizient umsetzbare programmtechnische Einheiten zu schaffen, die nach einheitlichen Prinzipien zusammenarbeiten. Hier spielten auch insbesondere Aspekte der Losen Kopplung zwischen programmtechnischen Einheiten eine wichtige Rolle. Die Units selbst werden parametriert und der Informationsfluß zwischen ihnen konfiguriert.

Das CBA-Unit-Konzept baut auf einem 3-Ebenen-Konzept auf. In jeder Ebene gibt es eine Art programmtechnischer Einheiten mit definierten Interfaces

- Data Access Layer
- Business Logic Layer
- User Interface Layer

Die Unit-Verwaltung kennt die konfigurierten Einheiten und instanziiert die Units dynamisch zu dem Zeitpunkt, zu dem sie das erste mal aufgerufen werden.

BILD UNIT FEHLT

Das CBA-Framework bietet für jede Ebene ebenenspezifische Zugriffsfunktionen, die die richtige Unit ermitteln und die gewünschte Funktion aufrufen. Das CBA-Framework und sein Vorgänger-Produkt ADT, das nach den gleichen Prinzipien arbeitet, haben sich in langjähriger Arbeit und in vielen Projekten bewährt. Es ist stabil und praxiserprobt, ermöglicht aber dennoch jede Art von Erweiterung. Diese Offenheit ermöglicht es, dass vielfältige Applikationen auf der Basis des CBA-Frameworks implementiert werden können.

Die starke Entkopplung programmtechnischer Einheiten zwingt einerseits zu einer klaren Strukturierung der Software, unterstützt jedoch andererseits den Software-Entwicklungs-Prozess sehr gut methodisch. Durch die Vorgabe von programmtechnischen Design-Pattern ist der Code leicht verständlich. Die verschiedenen Methoden der Einheiten haben wohldefinierte Aufgaben, sodass schon im Entwurf beginnend eine einfache und zugleich eindeutige Beschreibung der Programm-Logik machbar ist. Häufig kann auch ohne eine weitere formale Beschreibung der Code direkt implementiert werden. Diese Eigenschaft ist gewollt und fördert agile Methoden. Alternativ hat sich auch die Beschreibung in Form von Pseudo-Code in den programmtechnischen Rahmen als vorteilhaft erwiesen. Für größere Aufgaben können z.B. Modelle mit UML erstellt und in einem getrennten Schritt implementiert werden.

Architektur

Das CBA-Framework kennt drei Unit-Typen:

- DAU - Data Access Unit
- BLU - Business Logic Unit
- UIU - User Interface Unit

Prinzipiell kann die CBA-Unit-Verwaltung auch weitere Unit-Typen verwalten. Das Bild gibt einen Überblick über die Struktur der Interfaces und Klassen. Bei der Implementierung neuer Unit-Typen sind die beschriebenen Architekturprinzipien einzuhalten.

BILD FEHLT

Jeder Unit-Typ realisiert ein typspezifisches Interface, das von dem Interface **ICbaUnit** abgeleitet ist.**ICbaUnit** definiert die Funktionen, die für die dynamische Verwaltung der Units notwendig sind. Von dem Interface **ICbaUnit** bzw. der dazu gelieferten Basis-Implementierung **CbaUnit** werden die Interfaces bzw. Implementierungen abgeleitet. Dazu definiert CBA die Interfaces bzw. Basis-Implementierungen

- Data Access Units: **ICbaDau** bzw. **DauBasic**
- Business Logic Unit: **ICbaBlu** bzw. **BluBasic**
- User Interface Unit: **ICbaUiu** bzw. **UiuBasic**

Jede Unit wird durch eine Klasse implementiert. Die Unit-Verwaltung instanziiert Objekte der konfigurierten Klasse. In der Konfiguration der Units wird unter **Assembly** die DLL angegeben, in der die **Klasse** enthalten ist. Die Angaben zu den Assemblies erfolgen ohne die Dateierweiterung *.DLL. Die Klasse wird mit dem Namespace angegeben.

Im CBA-Lieferumfang ist eine Menge von Standard-Implementierungen enthalten. Diese systemseitig angebotenen Units sind jeweils von einer typspezifischen Basisklasse abgeleitet worden.

Konfiguration

Neue Units können folgendermassen implementiert und zu CBA hinzugefügt werden:

- Implementierung der gewünschten Funktionen durch Ableitung von der Basisklasse
- Einfügen eines neün Datensatzes in die Unit-Verwaltung unter Menüpunkt Konfiguration - Einheiten unter Angabe aller erforderlicher Konfigurations-Parameter
- Upload des Assembly und der Dokumentation

Zu Laufzeit instanziiert die Unit-Verwaltung die Objekte der konfigurierten Klasse als Unit und bindet sie in den Informations- und Steuerfluss dynamisch ein.

Funktionen

Für Unit-Verwaltung stehen folgende Funktionen des CBA-Frameworks zur Verfügung:

- **getUnit:**instanziiert die Unit des angegeben Typs und der angegebenen ID
- **getDau:**ermittelt bzw. instanziiert die konfigurierte DAU für die Tabelle. Jede DAU wird nur einmalig instanziiert, auch wenn sie für mehrere Tabellen konfiguriert ist.
- **getBlu:**ermittelt bzw. instanziiert die konfigurierte BLU für die Tabelle. Jede BLU wird pro Tabelle instanziiert.
- **getUiu:**ermittelt bzw. instanziiert die konfigurierte UIU für das User Interface. Jede UIU wird pro User Interface instanziiert.

getUnit wird nur für interne Zwecke genutzt, bietet aber auch die Möglichkeit, andere Unit-Typen mit der CBA Unit-Verwaltung zu verwalten. Zu diesem Zwecke würde sinnvollerweise eine ähnliche Funktion wie getDau, getBlu oder getUiu implementiert werden, die dann von aussen zugegriffen wird.

Die Funktionen der Unit-Verwaltung werden hauptsächlich aus dem CBA-Framework aufgerufen. Für die Erweiterung des Framework sind sie jedoch sinnvoll.

From:
https://wiki.tim-solutions.de/ - **TIM Wiki / NEW TIM 6 Documentation**

Permanent link:
https://wiki.tim-solutions.de/doku.php?id=software:cba:unit_management&rev=1416842618

Last update: **2021/07/01 10:01**