Análisis: Múltiples instancias de Bigbang single dispatcher

Para que varios bigbans con bases de datos diferentes funcionen en una instancia de Dispatcher (Local,web,remoto) se debe especificar un parámetro mas que es un identificador para ubicar la cadena de conexión.

El problema lo tenemos entonces divididos según las capas de la arquitectura

1. **Front End** : Se debe atacar a las llamadas de los servicios generalmente una clase llamada Controller y pasarle el identificador de la aplicación al Request.ContextInformation:
2. Este identificador se podría sacar de un appSetting o del Fwk.ConfigManager. El ultimo es el más optimo ya que se cambia desde un solo lugar.-
3. Opcionalmente se puede definir un estándar de esta ubicación. Si lo hacemos entonces no será necesario pasar esta data desde la clase controller ya que lo estaría haciendo directamente el **wrapper**.-

La ubicación podría ser directamente en el configmanager donde apuntan los clientes con una clave y grupo ya definido.. si esta no existe se podría pasar directamente “”. (1)

1. BackEnd.-
2. Componente SVC: Los servicios en la capa SVC deberán pasar esta información a la BC o DAC según que se esté utilizando directamente.- (2)
3. Componente BC hay que realizar los siguientes pasos:
4. Debido a que las BC se instancian, la solución es a través de una modificación de su constructor en la clase BaseBC.- Exigiendo que se pase un parámetro que identifique CompanyId en cada una de las clases hijas(3.1).- Debido a la existencia de una clase base es fácil la detección de la falta del constructor en tiempo de compilación.- (3.2)
5. Pasar el valor **CompanyId** al componente DAC desde la BC.-
6. Con respecto a las capa DAC que son estáticas la solución podría ser una de las siguientes:
7. Hacer todas las llamadas dinámicas a las DAC y trabajarlas de igual manera que las BC,
8. Pasar un parámetro más en cada método que identifique **CompanyId.** Esto se puede hacer con una aplicación recursiva de manera automática.- (4.1)

Para obtener la cadena de conexión se deberá llamar a un método de la clase BasesDAC estático el **CompanyId.** Finalmente la clase baseDAC obtendría la cadena(4.2)

**NOTA**: La ultima parece ser la más correcta y sencilla

Pasos siguientes

Para próximos desarrollos ya se debería tener en cuenta que los componentes deberían ser desarrollados de esta manera. Y también, habría que adaptar el generador de código y los wizard de visual sdudio para que se adapten a este requerimiento

Grafo de arquitectura con la solución recomendada.-

Front

Wrapper

REQ

REQ + CompanyId

DAC.Method1(…pCompanyId)

DAC.Method2(…pCompanyId)

BC : BaseBC(CompanyId)

BC : BaseBC(CompanyId)

SVC(Req)

REQ

Dispatcher

AppSetting

**3.2**

**2**

**1**

**BaseBC**

String pCompanyId

**3.1**

**4.1**

**BaseDAC**

ConfigManager

From AppSetting

**4.2**

From cnfgMannager