

How to

# Despliegue de plataforma Q-flow



*Versión: 2.0*

*Fecha de publicación 11-04-2011*

*Aplica a: Q-flow 3.0 y Q-flow 3.1*

## Índice

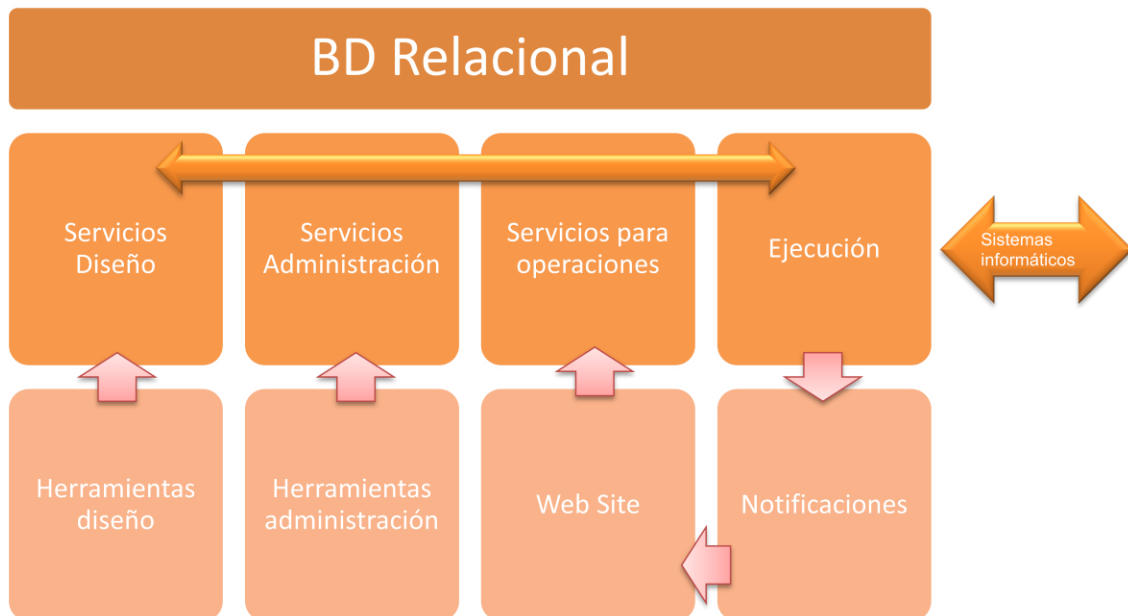
Diagrama de Servicios de Q-flow .....	3
Diagramas de Infraestructura de Q-flow 3.0 .....	4
Infraestructura general .....	4
Despliegue simple .....	5
Despliegue Standard .....	6
Despliegue Enterprise .....	7
Diagrama complete .....	8

## Diagrama de Servicios de Q-flow

Q-flow 3.0 ha sido desarrollado según una arquitectura SOA (Services Oriented Architecture) con lo cual existen una serie de servicios que agrupan funcionalidades similares entre sí y todos ellos conforman una capa llamada BackEnd. Los servicios del BackEnd son desacoplados entre sí y desacoplados a su vez de la capa de Persistencia, en donde se encuentra una base de datos relacional. Los servicios del BackEnd proporcionan a su vez, las funcionalidades disponibles en la capa llamada FrontEnd debido a que es donde interactúa el usuario.

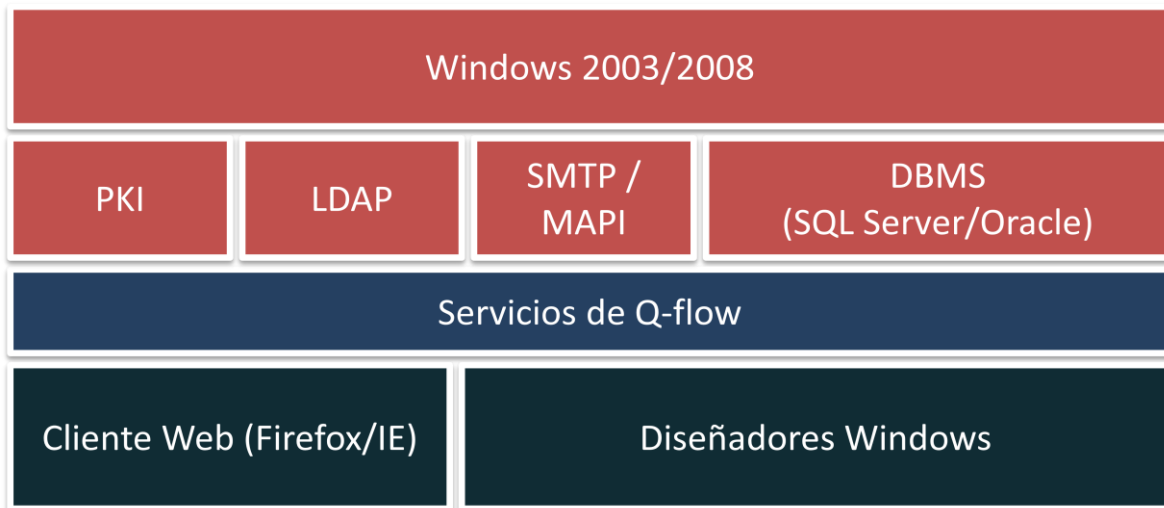
El FrontEnd también está construido de forma de ser totalmente desacoplado del BackEnd, entre ellos la comunicación se realiza mediante mensajes Xml que viajan a través de un canal, típicamente .Net Remoting/TCP. Esta capa está compuesta por el Diseñador de Procesos de Negocio, el Diseñador del Modelo Organizacional, Administrador de Procesos y el Sitio Web de Q-flow, componente principal, ya que es el que soporta todos los usuarios que trabajan con los flujos.

Cada uno de ellos posee a su vez en la capa de BackEnd un servicio homólogo. El BackEnd además posee el servicio de ejecución de flujos o “motor de workflow”, este servicio es el responsable por que los flujos avancen. Otros servicios que componen el BackEnd, son los de notificación, encargados de enviar mensajes a través de los diversos servidores de mensajería que pueden ser configurados.



## Diagramas de Infraestructura de Q-flow 3.0

### Infraestructura general



Tomando Windows 2003/2008 como Sistema operativo donde ejecuta Q-flow; cada una de las restantes cajas que figuran en el anterior diagrama, pueden instalarse en diferentes servidores o configurar diferentes esquemas hasta el más simple, donde todo está instalado en el mismo servidor. A continuación se detallan los principales esquemas.

## Despliegue simple

En este escenario todos los servicios de Q-flow ejecutan en un único servidor; si bien consume servicios de red como Base de datos, mensajería y directorio, lo ideal es que esos servicios se encuentren en otros servidores, pero es posible utilizar incluso la base de datos en el mismo servidor de Q-flow.

Este modelo es utilizado comúnmente como entorno de desarrollo.



Qflow Server

En un Q-flow Server ejecutan tanto los servicios de BackEnd como los de FrontEnd. Es un entorno simplificado usado para entornos de desarrollo



Servidor de BDD



Servidor de E-mail



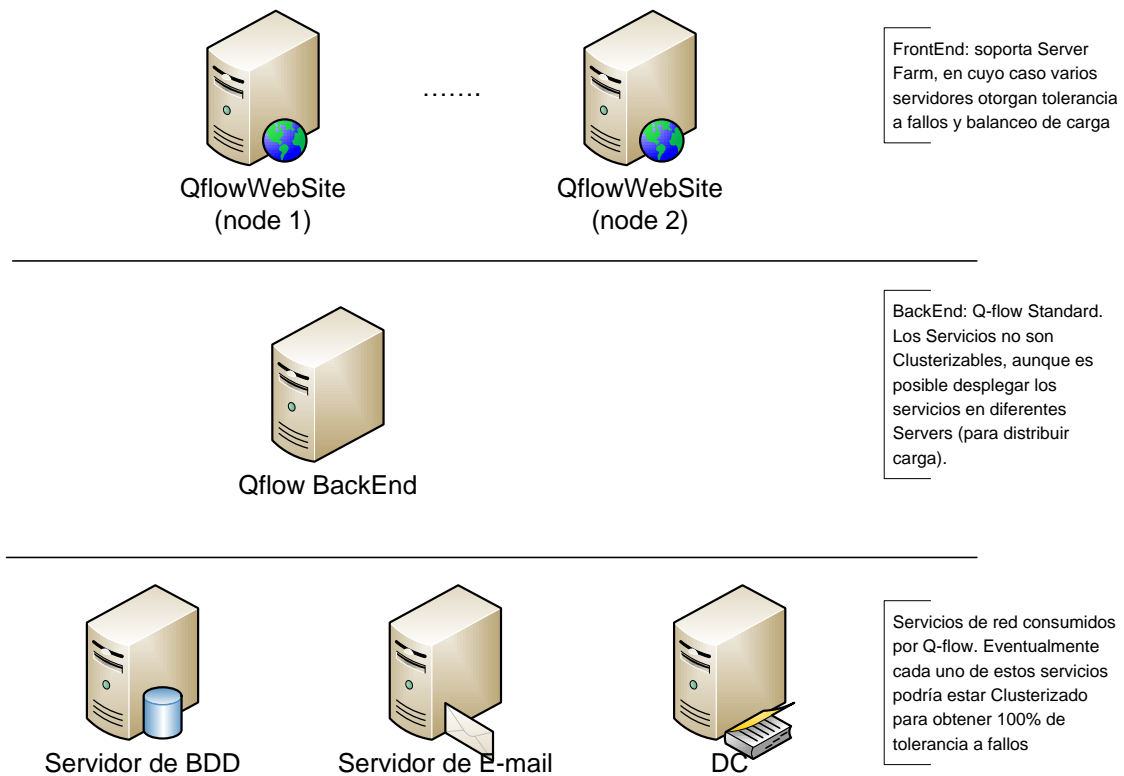
DC

Servicios de red consumidos por Q-flow. Eventualmente cada uno de estos servicios podría estar Clusterizado para obtener 100% de tolerancia a fallos

## Despliegue Standard

En este escenario los servicios de Q-flow se dividen en dos, BackEnd y FrontEnd. Los servicios de BackEnd, no son clusterizables por el propio Q-flow pero sí es posible montar un cluster de Windows para lograr el mismo objetivo. En cambio los servicios de FrontEnd si pueden colocarse en varios servidores, armando una granja utilizando NLB (Network Load Balancing), lo que le permitirá disponer de tolerancia a fallos de esta capa y balanceo de carga.

Este tipo de escenario es utilizado por medianas implementaciones de Q-flow, las cuales generalmente no cuentan con tolerancia a fallos en todas las capas.



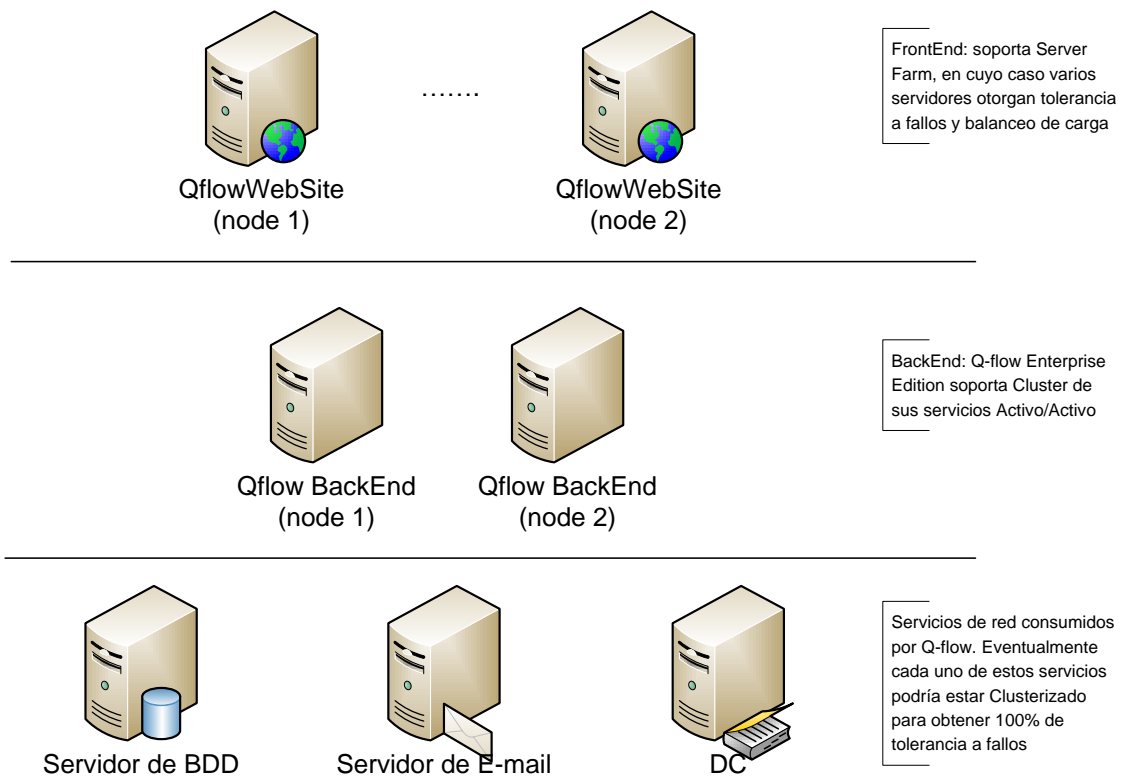
## Despliegue Enterprise

En este escenario la tolerancia a fallos se da en todas las capas. Es utilizado por implementaciones grandes que requieren alta disponibilidad.

Está compuesta por una capa llamada FrontEnd, en la cual se puede contar con varios servidores Web que proporcionan tolerancia a fallos y balanceo de carga. Y por una capa llamada BackEnd, la cual puede contar con varios servidores en modalidad de cluster Activo/Activo. Cabe destacar que el servicio del BackEnd que cuenta con balanceo de carga y tolerancia a fallos interno es el motor de workflows o *QflowEngineListener*. Los demás servicios que confirman el BackEnd deben colocarse en una granja de NLB de Windows para lograr similar efecto.

Este tipo de escenarios es típicamente utilizado por implementaciones grandes que requieran tolerancia a fallos en el ambiente de producción. Es común montar un escenario similar para el ambiente de Pre-Producción u homologación.

Este tipo de despliegue requiere una licencia especial de servidor de Q-flow llamada Q-flow 3.0 Enterprise Edition.



En todos los escenarios presentados, es posible que los servicios de Q-flow puedan ejecutar en servidores virtualizados.

## Diagrama completo

