Stringflow Server Architecture

Stringflow is a XMPP based server technology designed to power high traffic, interactive applications such as messaging applications, bidding infrastructures, gaming serves, chat-bots etc. Apart from core XMPP, server supports a long list of XEPs (the list can be found on website/appendix).

The server has following artifacts:-

* Server Build
* Java SDK
* Android SDK
* iOS SDK
* Java Script SDK

High-level Architecture

Below diagram explains very high level architecture of Stringflow server.

Session Manager

Disco Manager

Presence Manager

Connection Managers

Custom Component

Push Notification Manager

Media Store Manager

Server IO

IO Controller

Stringflow server uses a concept of Server Components; the whole server is nothing but a set of server component along with an administration layer around those components. Each server component is an independent entity which maintains its configurations, packet processing mechanism, packet filtering mechanism, storage and concurrency levels. There are three types of server components:-

* Core Components
* Packet Processors
* Custom Components

Core Components- Server components which are involved in core functioning of the server. Server IO, Connection Manager, Router, Session Manager etc are such components. Stringflow will load these components even if these components have been removed from the server configurations

Packet Processor- Each of the packet processors process certain packets. They may or may not generate another packet though. Some of these components can be removed from server configuration and Stringflow will not load these components. For example “Push Notification Manager” can be removed from the server configuration; in this case Stringflow will not generate push notifications

Custom Component- Developers can develop new server components and these components will be loaded by Stringflow runtime. These components generally contain some business logic and process certain packets and certain way. Custom component development process has been documented in “Stringflow Developer Guide”.