F2

The Open
Financial framework

F2

F2 is an open and free web integration framework for building financial services applications.

Using F2, firms can develop their own solutions that combine tools and content from multiple providers into a single, seamlessly integrated front end.

The essential components of the F2 framework are the Container, Apps, Context, and the Registry, each of which is described on the following pages.

F2.js is an open source JavaScript library available on GitHub. All relevant resources, including detailed documentation, code samples, and an API reference are available in the F2 Developer Center.

Problems solved

F2 solves several problems shared across the financial services industry.

Standardization

F2 provides a standard mechanism for integration across organizations, teams, and third parties, allowing companies to work together smoothly.

Efficiency

Using F2, developers can build once and deploy to any F2-enabled platform on any device, eliminating the need for duplicative work.

Agility

Unlike with a closed monolithic system, F2 allows platform owners to progressively make enhancements and migrate users and tools gradually as time and resources allow.

Benefits

F2 provides several easy wins for financial services firms and their vendors.

Reduced time to market

Using F2, developers can work off of a shared standard specification to quickly build Apps. In addition, firms can immediately deploy their products to one or many F2-enabled platforms, meaning a shorter time to market.

Quick iterations

F2 allows for multiple quick iterations based on user feedback or changing business needs. Using F2, firms can compete on features, make enhancements, or switch content providers with little risk.

Security

In an F2-enabled platform, Apps can communicate and pass information to each other without revealing sensitive user information to a third party provider.

Multi-vendor approach

F2 allows firms to choose the data and content they want from any single provider or multiple providers, and integrate it seamlessly into their solution. Below are just a few examples of the types of Apps that can benefit from the F2 standard.

Video

Streaming video sourced from a third party won't slow down page load times and frees the page owner from hosting costs. The provider can create one App and deploy it many times, either as a singular view or as part of an aggregated view.

Heat map

A heat map benefits greatly from F2's responsive design specification. Regardless of the provider, navigation and size adapt to the device.

Research

Page owners can include F2 Apps that contain research from preferred research providers; the research displayed can be customized based on user entitlements. A user can also select companies based on their portfolio or watchlist, or on the fly.

Container

The Container is a web page that serves as the foundation of any F2-enabled solution. The Container is where Apps reside, as well as the layer between the browser and Apps. If you have a website, you are one step away from having a Container.

Details

— Can be built and hosted by any party

— May take shape as entire website, or as wrappers around individual Apps

— Can be public or private (behind login)

— Container owner's branding, look & feel are represented here

— Supports the Apps and content chosen by the Container owner

— Responsible for:

  — Configuration and preferences

  — Entitlements and login details

  — Common or user-specific menus

  — Federated search

  — Inter-app communication via F2.js

  — Placement of Apps within the grid

Apps

Apps are synonymous with modules, widgets, and portlets. Think web-based HTML5 charts, portfolios, trade tickets, and screeners. An App is developed once, and can be used across multiple pages on multiple devices. Apps come in a few different flavors, including front-end Apps and data Apps (similar to an API).

Details

— Small web pages, consisting of HTML, CSS, JavaScript, and data that the Container owner is entitled to

— Data-only Apps can be created and used to power any front-end F2-enabled App

— Can be viewed either as a single App or as an aggregation of multiple Apps working together

— Can be public or private (behind login)

— Are called by the Container and passed relevant information (Context) which includes who the user is, his entitlements, and the App dimensions

— Browser-based, with no installation required

— Can be reused within different Containers and on any device that is F2-enabled

Context

Context is information about the user, such as what data he is entitled to and what searches he recently performed. The Container and Apps share Context with each other, so the user enjoys an integrated experience among Apps and the Apps smartly display information relevant to the user.

Details

— F2's in-browser, event-based, secure message system

— Both Container and App providers have control over inter-App communication

— Containers and Apps can publish and subscribe to messages

— Data including a user's portfolio positions, entitlements, and preferences could be passed via Context

— Context, represented as JSON, can easily be interpreted on both the client and the server
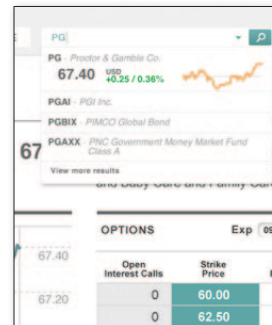
User enters an entity in the search box. In this case, Procter and Gamble.

```
CODE

F2.Events.emit(
    F2.Events.APP_SYMBOL_CHANGE,
    {
        symbol: "PG",
        name: "Procter & Gamble Co"
    }
);
```
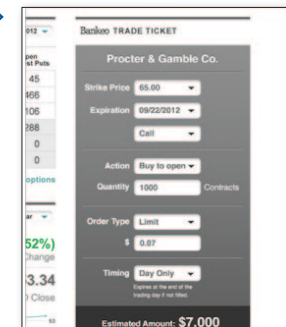

Via Context, all Apps that are programmed to listen to the search box automatically populate with Procter and Gamble-specific content.
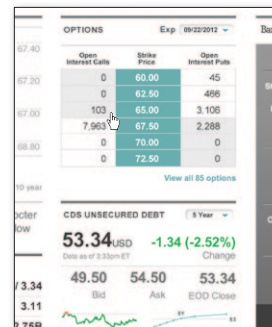
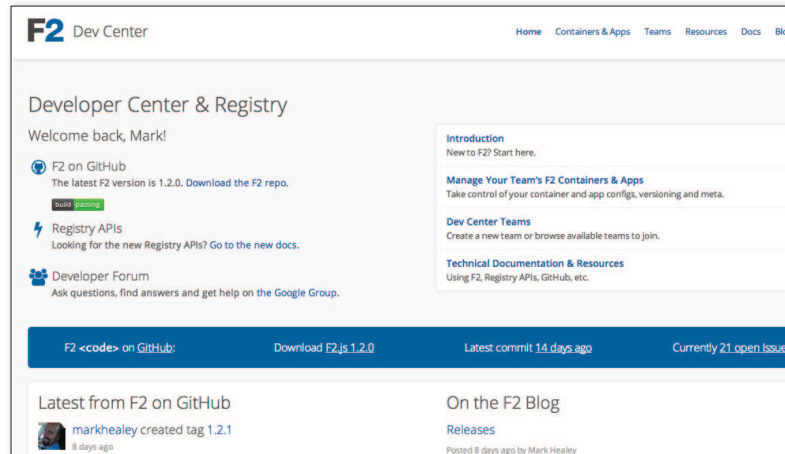*The user then selects an option from the options App.*

```
CODE

F2.Events.emit(
    "trade_ticket",
    {
        symbol: "PG",
        name: "Procter & Gamble Co",
        orderType: "Buy To Open",
        optionSymbol:"PG100416C0023"
    }
);
```


The trade ticket App is listening to the options App and automatically populates with the chosen symbol.

Developer Center & Registry

The Developer Center is where F2 developers register Containers and Apps, retrieve configurations, and access F2 resources. The Developer Center also allows a team of developers to collaboratively create, maintain, and deploy Apps.

Details

— Developer creates an F2 developer account that allows him to register and manage Containers and Apps

— Apps can be marked as public (which means they are discoverable by anyone with access to the Registry) or private (which means that they are not discoverable)

— Containers and Apps can be labeled with tags to enhance search

— Users can browse and preview Apps within an organization, and retrieve automated App configurations
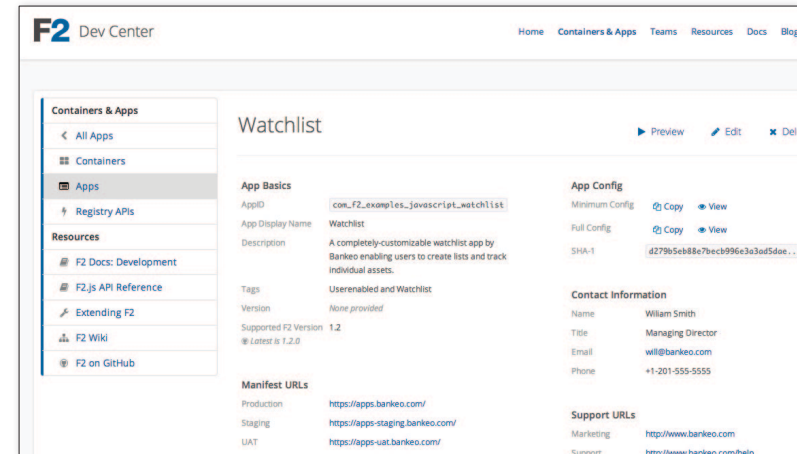
The Registry supports the configuration and distribution of F2-enabled solutions. Using the Registry database spec, firms can manage the Apps that work together to create an integrated user experience.

Details

— Open source database schema available on GitHub

— Stores F2 Containers and Apps registered in the Developer Center

— Export Container and App configurations using web services (authentication required)

— Allows for the development of F2 App stores by leveraging Registry-stored information

F2 to date

In 2012, we launched F2, providing a new way for financial firms to integrate tools and content from multiple providers into a single front-end. In 2013, the F2 community grew to include some of the financial industry's biggest firms. Collectively this community has written over a million lines of code and built thousands of apps, and grown into a collaborative ecosystem where members share ideas and best practices.

Highlights

F2 Advisory Board totals 14 firms

2013 Hackathon: Advisory Board firms gathered for a day of hacking, learning and collaborating

16 framework spec/code releases

1.2 million lines of F2 spec and code added or removed

Between 20 and 25 firms using F2 globally

2,500+ apps developed

2014 Summit: Advisory Board firms will meet in September 2014 to discuss the future of F2

Outreach

Finovate Conference, NYC in September '13

FinTech Hackathon, NYC in November '13

O'Reilly FluentJS Conference, San Francisco in March '14

Big Boulder, Boulder in June '14

Finovate, NYC in September '14

Stocktoberfest, San Diego in October '14

Steady stream of F2 introductions to many firms

Around the web

F2 blog
— 29 posts and counting
— 4,000 page views

F2 website
— 21,000+ sessions
— 57,000+ page views

F2 on GitHub
— 927 code changes
— 135 closed Issues
— 9 contributors, with dozens more behind the scenes
— hundreds of discussions

F2 Advisory Board

The evolution of F2 is overseen by an independent Advisory Board, comprised of representatives from leading financial services firms and content providers.

The Board is tasked with ensuring that F2 evolves to meet the needs of the industry. Board members are expected to promote the F2 spec within the broader financial technology community and contribute to the evolution of the F2 framework.

Currently 14 firms are represented on the Advisory Board.

F2 adoption

Our goal is to see F2 become the industry standard for building financial websites.

Markit On Demand, core maintainer of F2, displays its commitment to F2 by developing internal and client solutions that use the F2 framework, and through ongoing community education.

Advice, guidance, and thought leadership from the community are integral to the success of F2.

Additional information, as well as the latest version of the framework, is available at **www.OpenF2.org.**

More Information

Visit: OpenF2.org
Email: Info@OpenF2.org
Blog: Blog.OpenF2.org
Twitter: @OpenF2