System Overview

Facility Source (FS) provides facility maintenance optimization to its Clients nationwide. We use fmPilot 1.0 & fmPilot 2.0 as a proprietary software to facilitate every aspect of our service.

Both the systems offer our Clients a one stop solution for work order Management and Asset Tracking. These systems are used by our Associates, Clients and Service Provider (SP's).

Each user has permission based access, so they have access to functions and visibility within the fmPilot systems based on their requirement.

fmPilot has two parts: 1.0 and 2.0, and usage of these depends on the SP chosen to complete the work.

fmPilot: This system is used by our Clients to create and manage work order (WO).

fsElite: This app is used to track the SP and their activities on the assigned WO.

Internal API

There are several systems (mostly invisible to the end user e.g. client or SP) which exchange data with each other during the work flow of a WO for various purposes. For an example, markup and taxations process; the IFM Bridge; validation of invoice and quotes for pricing engine and Work Order queue (this is an external application). Internal APIs are the ones that enable the transmission of data between systems of this type. The Internal APIs collect some data and make it available to a certain processing engine(s) and collect the output finally storing it in the database. Further this output can be used by other types of APIs for other related purposes.

Work Order Pulling

The first required operation is pulling of existing WO to retrieve assigned work orders and update the existing work orders. This is an on-demand service that will respond to requests for data rather than an active push of new data, which would require SP's to construct their own services for reception. There are finite limits on the numbers of work orders that may be retrieved in a single pull to prevent any resource “hogging” by large queries.

Work Order Updates

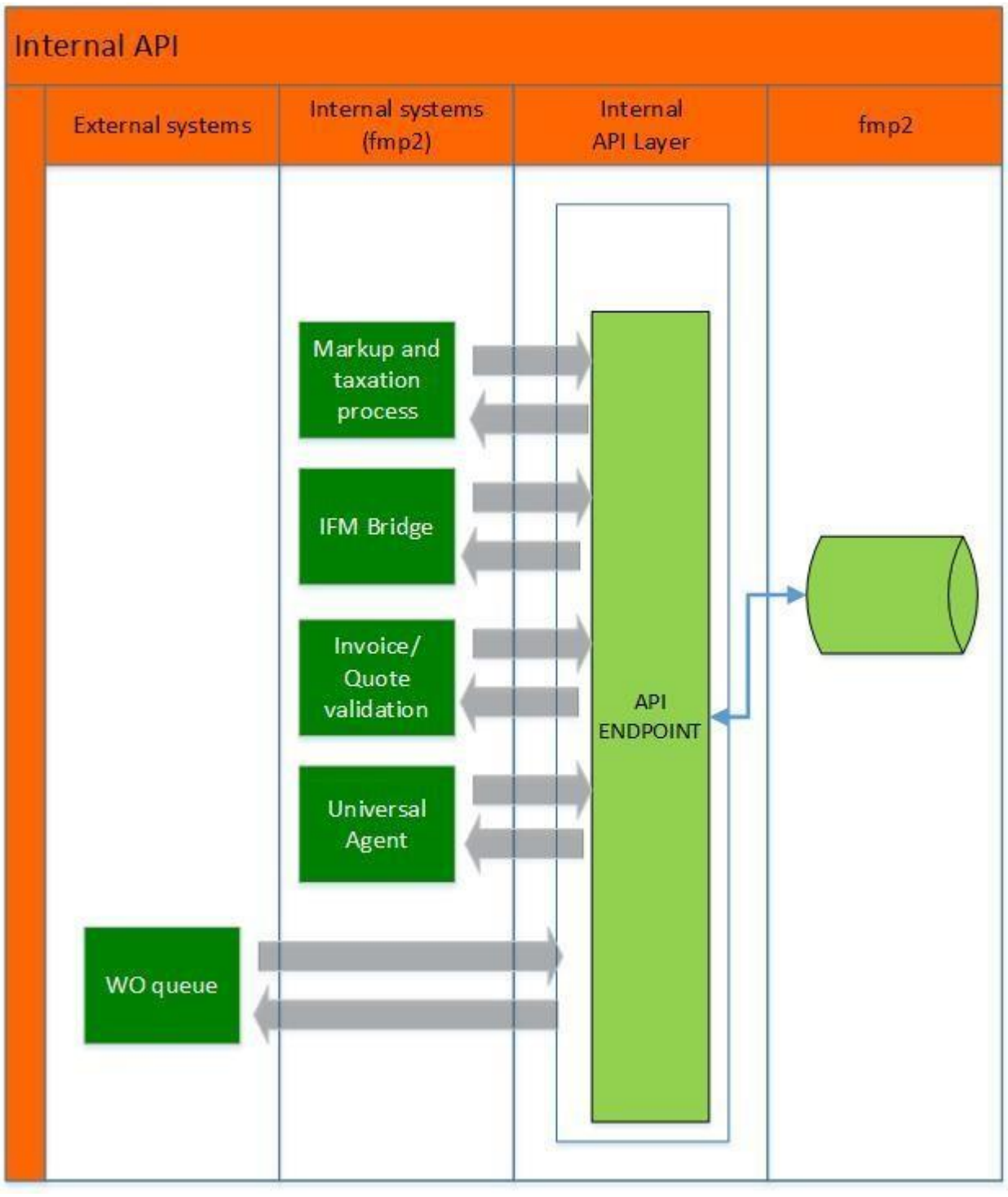
SP frequently have a need to modify the status of a WO, and often the status has been modified within a SP’s core systems and that data simply needs to be transmitted. This may be a fulfillment of a WO, acceptance of a WO, or suspense of a WO that is awaiting materials.

Each of these statuses has its own business logic within the application itself, so this will need to be mirrored within the service to ensure that all notifications, validations and processes are followed correctly. Additionally, all status updates undergo a general validation rule prior to transition-specific validations.

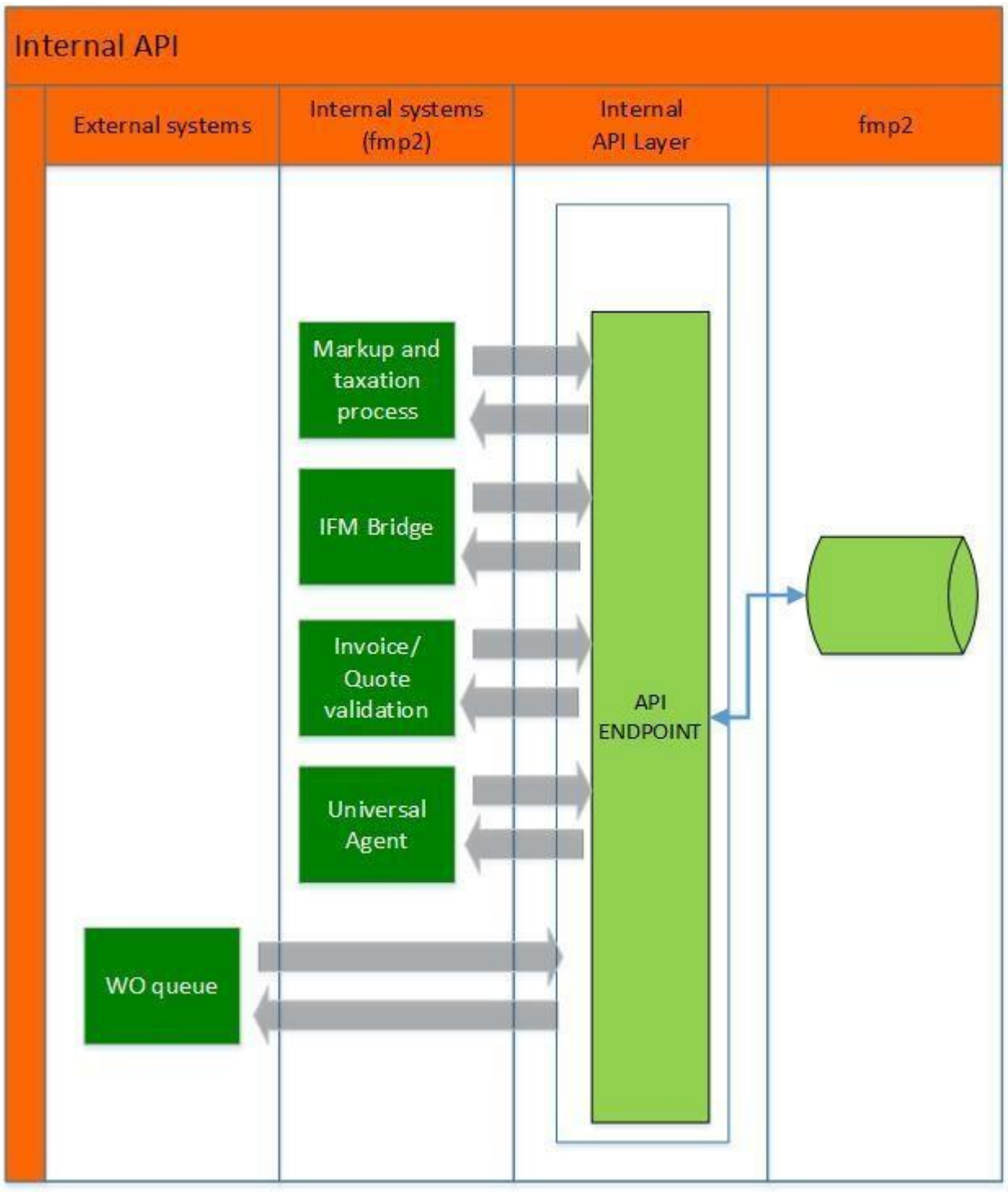
Work Order Comments and Notes

As WO evolve, technicians need to apply notes either for client usage, notes to continuing technicians, and other pertinent administrative information. A service is implemented that receives and annotates these comments relative to a particular WO. Pulling for comments are also implemented.

System Workflow



System Architecture



Implementation Consideration

Before you build an integration application or other client application, consider the data management, use limits, and communication issues explained in this section.

Web API Description

The system implements HTTP services by employing ASP.NET API Web API that can be consumed by a broad range of clients including browsers and handheld devices(smartphones and tablets). It supports variety of GET, PUT and POST request-response message system and provides the result in JSON/XML format.

Recommended Testing Tools

One of the simplest tools for testing is Postman. It is currently one of the most popular tools used in API testing. We will be referencing this tool mostly, but any API testing tool can be used with similar configuration and results.

Before going live, thoroughly test and debug your applications in the Sandbox Mode. You can use your favorite REST client to explore our API, but we recommend the Postman, or Fiddler.

Content Negotiation

With Web API content negotiation, the system returns data based on the client requests. If the client is requesting the data to be returned as JSON or XML, the Web API framework deals with the request type and returns the data appropriately based on the media type. By default Web API provides JSON based response.

API Overview

The Internal API is a RESTful and stateless API. It has predictable, resource-oriented URLs and accepts POST, GET, PUT, and DELETE requests. Our API uses standard HTTP response codes and returns JSON-encapsulated data.

API Version

The current stable version of the Internal API is **v1**

Environments

There are two environments for working with our API: Sandbox for testing and Production for live integrations.

App Registration

To use our API, you should have either a subscriber account with the Super Admin role or provider account with admin rights. You also need to register your application.

Authentication and Authorization

We use token authentication, yet secure and effective authentication and authorization

You must include the Authorization header in every API request. Unauthorized calls will fail.

Requests and Responses

Making requests are easy: our API accepts the most common HTTP verbs and has well-structured endpoint routes.

After a successful request, you will receive a JSON-formatted response as well as a standard HTTP status code.

Call Limits

There are no limits to how many calls you can make in a minute.

Sandbox Mode

We strongly recommend you to start in Sandbox Mode and move to the Production Environment only when your application is fully tested and ready to go live.

App Registration

Register your application to obtain calling client, acting domain and authentication token that are required for authentication and authorization.

To test an application in the Sandbox or Production Environment:

Currently staging environment is a sandbox environment.

To test API(s) use https://staging-api.fmpilot2.com/Internal/api/{{apiname}}

for e.g. to test clients domains execute - "https://staging-api.fmpilot2.com/Internal/api/WorkOrderQueue/WorkUsers"

Security

The system implements HTTP services by employing ASP.NET WebAPI that can be consumed by a broad range of clients including browsers and handheld devices(smartphones and tablets).

It supports variety of GET, PUT and POST request-response message system and provides the result in JSON or XML format.

With WebAPI content negotiation, the system returns data based on the client requests. If the client is requesting the data to be returned as JSON or XML, the WebAPI framework deals with the request type and returns the data appropriately based on the media type. By default, WebAPI provides JSON based response.

WebAPI is a trending technology. As we are exposing our WebAPI to the outside world, we should maintain security in WebAPI. It means a valid user can only access WebAPI, or else it will throw an unauthorization error.

Authentication

One of our primary concerns will be security and integrity of data and/or requests, For more details proceed to Authentication and Authorization.

Authentication

One of our primary concerns will be security and integrity of data and/or requests, so we will be establishing a special API authorization that will be used to perform an identity challenge along with domain to verify that the Internal Api are who they say they are.

Basic authentication is a simple authentication scheme built into the HTTP protocol. If a request requires authentication, the server returns 401 (Unauthorized). The response includes a WWW-Authenticate header, indicating the server supports Basic authentication.

WebAPI is a trending technology. As we are exposing our WebAPI to the outside world, we should maintain security in WebAPI. It means a valid user can only access WebAPI, or else it will throw an unauthorization error.

Error Handling

The API calls return error data that your client application can use to identify and resolve runtime errors. If an error occurs during the invocation of most API calls, then the API provides the following types of error handling:

For errors resulting from badly formed messages, failed authentication, or similar problems, the API returns a fault message with an associated Exception Code.

For most calls, if the error occurs because of a problem specific to the query, the API returns an Error.

Exception Handling

What happens if a WebAPI controller throws an uncaught exception? By default, most exceptions are translated into an HTTP response with status code 500, Internal Server Error. The HttpResponseException type is a special case. This exception returns any HTTP status code that you specify in the exception constructor. For example, the following method returns 404, Not Found, if the id parameter is not valid.

Sample:

```
public Product GetProduct(int id)
{
  Product item = repository.Get(id);
  if (item == null)
  {
    throw new HttpResponseException(HttpStatusCode.NotFound);
  }
return item;
}
```

In case of failure, the controller methods throw an HttpResponseException exception with the appropriate error message, which allows for a given HttpResponseMessage to be returned to the internal.

API Fault Elements in Common Authorization

Fault	Description
UNAUTHORIZED_MESSAGE	Please provide valid inputs for authentication token, calling client and acting domain.
UNAUTHORIZED_MAX_CALLS_MESSAGE	You have reached the max number of calls permitted by day.
NO_AUTH_REPO	No valid authorization provider could be validated for this request.

API Fault Elements in Comments

Fault	Description
NO_WO_NUMBER_OR_ID	The requested action requires a work order number or id.
NO_COMMENT_SUPPLIED	No comment was included to the request. A comment is required for this operation.
INSERT_COMMENT_ERROR	Error, one or more comments were not saved correctly.

API Fault Elements of BaseApi

Fault	Description
FILTER_ERROR	The provided filtering criteria is invalid. Please provide a valid one.
AUTHENTICATION_ERROR	The provided authentication/authorization header information is incorrect, please proceed to verify it.
INVALID_INTERNALFILTER	Invalid internal filter (CompoundFilterDTO).

API Fault Elements of IFMBridge

Fault	Description
RULE_FAILED_NO_MESSAGE	Business rules failed, but no message was received. Rules response code:
WORKORDER_NOT_FOUND	WorkOrder matching the Id or Number was not found.
VENDOR_NOT_IN_WHITELIST	WorkOrder vendor was not in the whitelist.
CLIENT_NOT_FOUND	Client matching the Id was not found.
SOURCE_STATUS_DOESNT_MATCH	Source({0}) workorder {1} is in status {2} not in status {3}, so the linked workorder may not be updated.

API Fault Elements in Markup

Fault	Description
NO_WO_NUMBER_OR_ID	The requested action requires a work order number or id.
INVALID_WO_NUMBER_OR_ID	The provided Work Order ID or Number doesn't match with a valid Work Order.
INVALID_STATUS	The current Work Order Status '{0}' is invalid.

API Fault Elements in Pricing

Fault	Description
RATE_TYPE_CANNOT_BE_EMPTY	Please provide a valid Rate Type.

Fault	Description
RATE_KIND_CANNOT_BE_EMPTY	Please provide a valid Rate Kind.
RATE_OPTION_CANNOT_BE_EMPTY	Please provide a valid Rate Option.

API Fault Elements in Quotes

Fault	Description
NO_WO_NUMBER_OR_ID	The requested action requires a work order number or id.
QUOTE_NOT_FOUND	No quotes could be found for the supplied Work Order.

API Fault Elements in WorkOrders

Fault	Description
NO_WO_NUMBER_OR_ID	The requested action requires a work order number or id.

API Fault Elements in IFMRequestMappingXML

Fault	Description
UNABLE_MAP_REQUEST	Unable to find mapped request code and request type.

Response Codes

Success Message(s)

Code	Text	Description
200	OK	The request was successful.
202	Accepted	The request has been accepted, but not yet processed.
203	Non-Authoritative Information	The request was successful, but the response data may be from a third party.

Redirection

Code	Text	Description
300	Multiple Choices	The request has more than one representation options, for example, different file formats. Select a preferred representation and redirect your request to the representation location.
301	Moved Permanently	The resource you requested has been permanently moved to a new location. Direct this and all future requests to the Uniform Resource Identifier (URI) specified in the Location response header.
302	Found	The resource you requested has been temporarily moved to a new location. Direct this request to the temporary URI specified in the Location response header, but continue using the original URI for future requests.
303	See Other	You can find the response to your request under another URI. Send a GET request to the URI specified in the Location response header.
304	Not Modified	The resource has not been modified since last requested, so there is no new data to return.
305	Use Proxy	Access the requested resource through the proxy provided in the Location response header.

Code	Text	Description
307	Temporary Redirect	The resource you requested has been temporarily moved to a new location. Direct this request to the temporary URI specified in the Location response header, but continue using the original URI for future requests. In contrast to code 302, you cannot change the request method when reissuing the original request.

Internal Error(s)

Code	Text	Description
400	Bad request	The request was not accepted due to bad syntax, missing parameters, insufficient data, etc.
401	Unauthorized	Missing or incorrect authentication credentials.
403	Forbidden	You are not authorized to request this resource, or the resource is unavailable for some reason.
404	Not Found	The request URI is incorrect, or the resource does not exist.
405	Method Not Allowed	The method specified in the Request-Line is not allowed for the resource identified by the Request-URI. The response must include an Allow header containing a list of valid methods for the requested resource.
406	Not Acceptable	This code is similar to 401 (Unauthorized), but indicates that the client must first authenticate itself with the proxy. The proxy must return a Proxy-Authenticate header field containing a challenge applicable to the proxy for the requested resource. The Internal may repeat the request with a suitable Proxy-Authorization header field. HTTP access authentication is explained in "HTTP Authentication: Basic and Digest Access Authentication".
407	Proxy Authentication Required	The format specified in the Accept header is not supported. Usually, the possible format is JSON. See Output Formats for details.
408	Request Timeout	The server has timed out while waiting for the request. You can repeat the request without modifications at any later time.

Code	Text	Description
409	Conflict	The server was not able to process the request because of a conflict. Study the response body to recognize the source of the problem. Resolve the conflict and reissue the request.
410	Gone	The resource you requested is no longer available. Do not request this resource again in the future.
411	Length Required	You have not stated the length of the content, which is required by the server. Add a valid Content-Length header and repeat the request.
412	Precondition Failed	The server could not meet one or more of the preconditions stated in the request headers.
413	Request Entity Too Large	The request is larger than the server is willing or able to process.
414	Request-URI Too Long	The URI provided is longer than the server is willing or able to interpret. You have probably used too many query-strings in a GET request. In this case, try to convert it into a POST request.
415	Unsupported Media Type	The request contains a media type that the server or resource does not support.
416	Requested Range Not Satisfiable	The server cannot provide the portion of the data that you specified in the Range request header. It is possible that the range is outside the size of the target data.
417	Expectation Failed	The server could not meet the expectation stated in the Expect request header.
421	Misdirected Request	The request was directed at a server that is not able to produce a response. This can be sent by a server that is not configured to produce responses for the combination of scheme and authority that are included in the request URI.
423	Locked	The resource that is being accessed is locked.

Code	Text	Description
424	Failed Dependency	The request failed due to failure of a previous request.
425	Too Early	Indicates that the server is unwilling to risk processing a request that might be replayed.
426	Upgrade Required	The server refuses to perform the request using the current protocol but might be willing to do so after the Internal upgrades to a different protocol. The server sends an Upgrade header in a 426 response to indicate the required protocol(s).
428	Precondition Required	Switch to a protocol stated in the Upgrade header and repeat the request.
429	Too Many Requests	The user has sent too many requests in a given amount of time ("rate limiting").
431	Request Header Fields Too Large	The server is unwilling to process the request because its header fields are too large. The request may be resubmitted after reducing the size of the request header fields.
451	Unavailable For Legal Reasons	The user requests an illegal resource, such as a web page censored by a government.

Server Error(s)

Code	Text	Description
500	Internal Server Error	Something went wrong, and the server was unable to complete your request. Should this problem persists.
503	Service Unavailable	The server is currently unavailable, or you have reached the throttling limit. Throttling - To keeps our API healthy and protect it from overuse, we set limits on the number of calls you can make per time interval. When developing your app, make sure to respect the API call limits.

Code	Text	Description
504	Gateway Timeout	The server, while acting as a gateway or proxy, has waited too long for a response from the upstream server.
505	HTTP Version Not Supported	The server does not support the HTTP version used in the request.
506	Variant Also Negotiates	The server has an internal configuration error: transparent content negotiation for the request results in a circular reference.
507	Insufficient Storage	The server has an internal configuration error: the chosen variant resource is configured to engage in transparent content negotiation itself, and is therefore not a proper end point in the negotiation process.
508	Loop Detected	The server detected an infinite loop while processing the request.
510	Not Extended	Further extensions to the request are required for the server to fulfill it.
511	Network Authentication Required	The 511 status code indicates that the Internal needs to authenticate to gain network access.

Object Basics

Generally speaking, API objects represent database tables that contain your organization's information. For example, the central object in the Facility Source (FS) data model represents accounts - companies and organizations involved with your business, such as customers, partners, and competitors. The term “record” describes a particular occurrence of an object. A record is analogous to a row in a database table. Objects can be standard objects, custom objects and external objects.

Applications work with only the objects that you are authorized to access. Programmatic access to objects are determined by the objects defined in your organization, your organization configuration, your user permissions and access settings (which are configured by your organization’s system administrator), your data sharing model, and other factors related specifically to the object.

Primitive Data Types

The API uses the following primitive data types:

Value	Details
Base64	Base 64-encoded binary data. Fields of this type are used for storing binary files in Attachment records, Document records, and Scontrol records.
boolean	Boolean fields have one of these values: true (or 1), or false (or 0).
Byte	A set of bits.
Date	Fields of this type contain date values, such as ActivityDate in the Event object.
DateTime	Fields of this type handle date/time values (timestamps), such as ActivityDateTime in the Event object or the CreatedDate, LastModifiedDate, or SystemModstamp in many objects.
Double	Fields of this type can contain fractional portions (digits to the right of the decimal place), such as ConversionRate in CurrencyType. Scale: Maximum number of digits to the right of the decimal place. Precision: Total number of digits, including those to the left and the right of the decimal place.
int	Fields of this type contain numbers with no fractional portion.
string	Fields that are of data type string contain text and some have length restrictions depending on the data being stored. The MailingStreet is 255 characters.
time	Fields of this type handle time values, such as FridayEndTime in the BusinessHours object. Development tools differ in the way that they handle time data.

External Information

Value	Details
ExternalAttachment	A set of enumeration See all enumeration list

List of ExternalAttachment

Name	Type
WorkOrderNumber	string
Id	integer
InvoiceId	integer
QuoteId	integer
Name	string
Path	string
URL	string
Username	string
CreatedDate	date
FormattedCreatedDate	string

Name	Type
SizeString	string
Latitude	string
Longitude	string

Quote Information

Value	Details
QuoteFragment	A set of enumeration See all enumeration list

List of QuoteFragment

Name	Type
Order	integer
SubmittedDate	date
RejectedDate	date
ApprovedDate	date
IncurredLaborCost	decimal number
IncurredPartsCost	decimal number

Name	Type
IncurredMaterialCost	decimal number
IncurredPartsAndOtherCost	decimal number
IncurredTotal	decimal number
EstimatedLaborCost	decimal number
EstimatedPartsCost	decimal number
EstimatedMaterialCost	decimal number
EstimatedPartsAndOtherCost	decimal number
EstimatedTotal	decimal number
SubTotal	decimal number
SalesTax	decimal number
VendorTotalTax	decimal number
Total	decimal number
TotalCost	decimal number

Name	Type
TotalLaborCost	decimal number
TotalPartsAndOtherCost	decimal number
TotalFreightCost	decimal number
TotalHandlingCost	decimal number
TotalMaterialCost	decimal number
TotalPartsCost	decimal number
TotalShippingCost	decimal number
Tax1ID	integer
Tax1Name	string
Tax1Value	decimal number
Tax2ID	integer
Tax2Name	string
Tax2Value	decimal number

Name	Type
Tax3ID	integer
Tax3Name	string
Tax3Value	decimal number
Tax4ID	integer
Tax4Name	string
Tax4Value	decimal number
Tax5ID	integer
Tax5Name	string
Tax5Value	decimal number
Id	integer
CreateDateTime	date
CreateUser	string
Description	string

Name	Type
LocationNumber	string
VendorCompanyName	string
VendorCode	string

WorkUserViewModel

Value	Details
WorkUserViewModel	A set of enumeration See all enumeration list

List of WorkUserViewModel

Name	Type
Id	integer
Name	string
Description	string
Exclude	boolean
TimeZoneRule	boolean
HoldOnTimeMinutes	integer

Name	Type
HoldOnTime	time interval
UserId	integer
UserName	string
FirstName	string
LastName	string
Role	string
RoleId	integer
IsPublished	boolean
RefreshPage	boolean

WorkUserListitem

Value	Details
WorkUserListitem	A set of enumeration See all enumeration list

List of WorkUserListitem

Name	Type
Id	integer
Name	string
Description	string
UserId	integer
UserName	string
FirstName	string
LastName	string
FullName	string
RoleId	integer
LastUpdated	date
IsPublished	boolean
Role	string

Environments

Internal API provides two environments: Sandbox Mode and Production.

Sandbox Mode

The Internal API Sandbox is a self-contained, virtual testing environment that mimics the live Internal API production environment. It provides a shielded space where you can initiate and watch your application process the requests you make to the Internal API without touching any live accounts.

Production

We strongly recommend you to start in Sandbox Mode and move to the Production Environment only when your application is fully tested and ready to go live. Production is an environment that contains real data and is designed for live integrations. Use this environment only for mature applications that are ready to go live.

Getting Started

Facility Source (FS) API has a host of different features, but the cornerstone of our platform is a work order .

About a Work Order

Think of a work order (WO) as a task issue. A WO details what happened and what should be done; it has a due date, assignee, and much more.

When subscribers need services like AC repair, plumbing etc, they create WO's that are assigned to certain providers. These providers solve the problem and “close” the WO.

Parameter Details

1. Request Parameter

To Get a WO, you need to pass some required information about the problem that occurred. In real life, you collect all these values using our API, but for a quick start we are providing you with the ready data.

Parameter	Value
WorkorderId	integer

2. Response Parameters

Parameter	Value
Id	integer
WorkOrderNumber	string
Type	string
WorkType	string

Parameter	Value
Status	string
StatusCode	string
Description	string
Priority	string
DateReported	date
DateModified	date
TargetDate	date
ScheduledStartDateTime	date
ScheduledCompleteDateTime	date
ActualStartDateTime	date
ActualCompleteDateTime	date
LatestInteractiveVoiceResponseIn	date
LatestInteractiveVoiceResponseOut	date

Parameter	Value
GLCode	string
RequestingContact	string
AlternateContact	string
RequestTypeId	integer
RequestType	string
DepartmentId	integer
Department	string
TradeId	integer
RequestCodeId	integer
RequestCode	string
ServiceLocationName	string
DNE	decimal number
WorkTypeGLCode	string

Parameter	Value
RequestGLCode	string
DispatchMatrixGLCode	string
AlternateWorkOrderNumber	string
IsETAMissed	boolean
MissedETAMessage	string
IsEmergency	boolean
IsVendorOnSite	boolean
HasAttachments	boolean
TargetCompleteDate	date
IsSnow	boolean
ProjectNumber	string
ProjectOther	string
CurrencyCode	string

Emails

API	Description
GET api/Emails/WorkOrders/{workOrderId}	No documentation available.
GET api/Emails/WorkOrders/{workOrderId}/AssociatedUsers	No documentation available.
GET api/Emails/WorkOrders/{workOrderId}/Recipients?searchTerm={searchTerm}	No documentation available.
GET api/Emails/WorkOrders/{workOrderId}/Recipients/Search?searchTerm={searchTerm}	No documentation available.
POST api/Emails/WorkOrders/{workOrderId}	No documentation available.
POST api/Emails?key={key}	No documentation available.

WorkOrders

API	Description
GET api/WorkOrders/{workorderId}	This endpoint retrieves all work order data from the source.
GET api/WorkOrders/{workOrderId}/Tracking	No documentation available.
POST api/WorkOrders/{workOrderId}/Activity/Viewed	No documentation available.
GET api/WorkOrders/{workorderId}/IVRLog	No documentation available.
PUT api/WorkOrders/{workorderId}/Schedule	No documentation available.
GET api/WorkOrders/Location/{locationId}	No documentation available.
GET api/WorkOrders/Vendor/{vendorId}	No documentation available.
POST api/WorkOrders?key={key}	No documentation available.

UniversalAgent

API	Description
GET api/UniversalAgent/Search?issueDescription={issueDescription}&locationId={locationId}&userId={userId}&clientId={clientId}&page={page}&pageSize={pageSize}&howIssueHappened={howIssueHappened}&whatsBroken={whatsBroken}&requestingContact={requestingContact}&serviceLocationId={serviceLocationId}&requestTypeId={requestTypeId}&departmentId={departmentId}&clientRoleId={clientRoleId}	No documentation available.
POST api/UniversalAgent?key={key}	No documentation available.

FinanceValidations

API	Description
POST api/FinanceValidations/ComparableInvoices	No documentation available.
POST api/FinanceValidations/ComparableInvoicesWithoutDetails	No documentation available.
POST api/FinanceValidations?key={key}	No documentation available.

WorkOrderQueue

API	Description
GET api/WorkOrderQueue/WorkUsers	No documentation available.
GET api/WorkOrderQueue/WorkUsers/{id}	No documentation available.
GET api/WorkOrderQueue/WorkUsers/ByUser/{id}	No documentation available.

API	Description
GET api/WorkOrderQueue/WorkUsers/IsPublished	No documentation available.
GET api/WorkOrderQueue/WorkUsers/ForAgents	No documentation available.
POST api/WorkOrderQueue/WorkUsers	No documentation available.
PUT api/WorkOrderQueue/WorkUsers/{id}	No documentation available.
DELETE api/WorkOrderQueue/WorkUsers	No documentation available.
GET api/WorkOrderQueue/WorkUsers/{id}/Preview	No documentation available.
GET api/WorkOrderQueue/WorkUsers/{id}/Export	No documentation available.
GET api/WorkOrderQueue/WorkUsers/{id}/CountAll	No documentation available.
GET api/WorkOrderQueue/WorkGroups	No documentation available.
GET api/WorkOrderQueue/WorkGroups/{id}?newName={newName}	No documentation available.
POST api/WorkOrderQueue/WorkGroups	No documentation available.
PUT api/WorkOrderQueue/WorkGroups/{id}	No documentation available.

API	Description
PUT api/WorkOrderQueue/WorkGroups/{id}/CopyUsers/{groupId}	No documentation available.
DELETE api/WorkOrderQueue/WorkGroups	No documentation available.
GET api/WorkOrderQueue/WorkGroups/{id}/Preview	No documentation available.
GET api/WorkOrderQueue/WorkGroups/{id}/Export	No documentation available.
GET api/WorkOrderQueue/WorkGroups/{id}/CountAll	No documentation available.
GET api/WorkOrderQueue/WorkOrderQueues	No documentation available.
GET api/WorkOrderQueue/WorkOrderQueues/{id}	No documentation available.
PUT api/WorkOrderQueue/WorkOrderQueues/{id}	No documentation available.
POST api/WorkOrderQueue/{id}	No documentation available.
DELETE api/WorkOrderQueue/WorkOrderQueues	No documentation available.
GET api/WorkOrderQueue/WorkOrderQueues/{id}/Preview	No documentation available.
GET api/WorkOrderQueue/WorkOrderQueues/{id}/Export	No documentation available.

API	Description
GET api/WorkOrderQueue/WorkOrderQueues/{id}/CountAll	No documentation available.
GET api/WorkOrderQueue/WorkOrder/{workOrderId}	No documentation available.
GET api/WorkOrderQueue/Disposition?fsPriority={fsPriority}&scheduleCompleteDateTime={scheduleCompleteDateTime}	No documentation available.
GET api/WorkOrderQueue/ClientExpectations?priority={priority}&fsPriority={fsPriority}&workType={workType}&workOrderId={workOrderId}	No documentation available.
GET api/WorkOrderQueue/NextWorkOrder	No documentation available.
GET api/WorkOrderQueue/Dashboard	No documentation available.
POST api/WorkOrderQueue/WorkOrderQueueHistory	No documentation available.
GET api/WorkOrderQueue/ODSRefresh	No documentation available.
GET api/WorkOrderQueue/WorkOrderQueueHistories	No documentation available.
POST api/WorkOrderQueue/CheckOut	No documentation available.
POST api/WorkOrderQueue/HoldOn	No documentation available.
POST api/WorkOrderQueue/WorkOrderQueueFMPTracking	No documentation available.

API	Description
GET api/WorkOrderQueue/WorkGroupDashboard	No documentation available.
POST api/WorkOrderQueue	No documentation available.
POST api/WorkOrderQueue?key={key}	No documentation available.

Markup

API	Description
GET api/Markup/MarkupContext?vendorId={vendorId}&clientId={clientId}&locationId={locationId}&tradeId={tradeId}&requestCodeId={requestCodeId}&requestTypeId={requestTypeId}	Retrieves a collection of Rate Cards joined with their RateType/Kind/Options (Overtime, Standard, etc..) that are available to the RateCards matched by the parameters vendorId, clientId, locationId, tradeId, requestCodeId, and requestTypeId.
PUT api/Markup/workOrders/{id}/invoice	No documentation available.
POST api/Markup?key={key}	No documentation available.

Invoices

API	Description
GET api/Invoices/WorkOrder/{workOrderId}	No documentation available.
POST api/Invoices?key={key}	No documentation available.

Pricing

API	Description
GET api/Pricing/LaborRate?vendorId={vendorId}&locationId={locationId}&tradeId={tradeId}&requestCodeId={requestCodeId}&requestTypeId={requestTypeId}&rateKind={rateKind}&rateOption={rateOption}	No documentation available.
GET api/Pricing/Rate?rateOption={rateOption}	No documentation available.
GET api/Pricing/RateKinds?rateTypeName={rateTypeName}&vendorId={vendorId}&locationId={locationId}&tradeId={tradeId}&requestCodeId={requestCodeId}&requestTypeId={requestTypeId}	Retrieves a collection of RateKinds (used to be called Skill Levels) that are available to the RateCards matched by the parameters vendorId, locationId, tradeId, requestCodeId, and requestTypeId.
GET api/Pricing/RateOptions?rateTypeName={rateTypeName}&rateKindName={rateKindName}&vendorId={vendorId}&locationId={locationId}&tradeId={tradeId}&requestCodeId={requestCodeId}&requestTypeId={requestTypeId}	Retrieves a collection of RateOptions (Overtime, Standard, etc..) that are available to the RateCards matched by the parameters

API	Description
	vendorId, locationId, tradeId, requestCodeId, and requestTypeId.
GET api/Pricing/RateContext?vendorId={vendorId}&locationId={locationId}&tradeId={tradeId}&requestCodeId={requestCodeId}&requestTypeId={requestTypeId}	Retrieves a collection of Rate Cards joined with their RateType/Kind/Options (Overtime, Standard, etc..) that are available to the RateCards matched by the parameters vendorId, clientId, locationId, tradeId, requestCodeId, and requestTypeId.
GET api/Pricing/TradeRateContext?vendorId={vendorId}&locationId={locationId}&tradeId={tradeId}	Retrieves a collection of Rate Cards joined with their RateType/Kind/Options (Overtime, Standard, etc..) that are available to the RateCards matched by the parameters vendorId, clientId, locationId, tradeId, requestCodeId, and requestTypeId.

API	Description
GET api/Pricing/TieredMarkUpContext?vendorId={vendorId}&clientId={clientId}&locationId={locationId}&tradeId={tradeId}&requestCodeId={requestCodeId}&requestTypeId={requestTypeId}	Retrieves a collection of Tiered MarkUp Cards joined with their RateType that are available matched by the parameters vendorId, clientId, locationId, tradeId, requestCodeId, and requestTypeId.
GET api/Pricing/MarkUpMapping?tradeId={tradeId}	No documentation available.
GET api/Pricing/LineItemMarkUpContext?vendorId={vendorId}&clientId={clientId}&locationId={locationId}&tradeId={tradeId}&requestCodeId={requestCodeId}&requestTypeId={requestTypeId}	Retrieves a collection of Tiered MarkUp Cards joined with their RateType that are available matched by the parameters vendorId, clientId, locationId, tradeId, requestCodeId, and requestTypeId.
GET api/Pricing/AvalaraConversionCodes?tradeId={tradeId}&requestTypeId={requestTypeId}&requestCodeId={requestCodeId}&lineItemTypeId={lineItemTypeId}&workTypeId={workTypeId}	No documentation available.
POST api/Pricing?key={key}	No documentation available.

IFMBridge

API	Description
POST api/IFMBridge/workOrders/{workOrderId}/create/{clientId}	No documentation available.
POST api/IFMBridge/workOrders/{workOrderId}/ClientCreate/{ifmId}	No documentation available.
PUT api/IFMBridge/workOrders/{workOrderIdOrNumber}/update	No documentation available.
PUT api/IFMBridge/workOrders/{workOrderIdOrNumber}/performStatusChange	No documentation available.
PUT api/IFMBridge/workOrders/{workOrderIdOrNumber}/activity/{activityType}	No documentation available.
POST api/IFMBridge/workOrders/{workOrderIdOrNumber}/comments	No documentation available.
PUT api/IFMBridge/workOrders/{workOrderIdOrNumber}/attachment/{attachmentId}	No documentation available.
DELETE api/IFMBridge/workOrders/{workOrderIdOrNumber}/attachment/{attachmentId}	No documentation available.
POST api/IFMBridge/workOrders/{workOrderIdOrNumber}/invoice/{invoiceId}	No documentation available.
PUT api/IFMBridge/workOrders/{workOrderIdOrNumber}/invoice/{invoiceId}	No documentation available.
POST api/IFMBridge/workOrders/{workOrderIdOrNumber}/quote/{quoteId}	No documentation available.
PUT api/IFMBridge/workOrders/{workOrderIdOrNumber}/quote/{quoteId}	No documentation available.
POST api/IFMBridge/workOrders/{workOrderIdOrNumber}/updateWorkOrderLink	No documentation available.
POST api/IFMBridge?key={key}	No documentation available.

Values

API	Description
GET api/Values/ClientLookup	No documentation available.
GET api/Values/TradeLookup?searchTerm={searchTerm}	No documentation available.
GET api/Values/VendorLookup?searchTerm={searchTerm}	No documentation available.

API	Description
GET api/Values/RequestTypeLookup?searchTerm={searchTerm}	No documentation available.
GET api/Values/RequestCodeLookup?searchTerm={searchTerm}	No documentation available.
GET api/Values/WorkTypeLookup	No documentation available.
GET api/Values/ZoneLookup?searchTerm={searchTerm}	No documentation available.
GET api/Values/StatusLookup	No documentation available.
GET api/Values/StateLookup	No documentation available.
GET api/Values/StoreCompanyCodeLookup?searchTerm={searchTerm}	No documentation available.
GET api/Values/VendorStatusLookup	No documentation available.
POST api/Values?key={key}	No documentation available.

Quotes

API	Description
GET api/Quotes/WorkOrder/{id}/{quoteOrder}	Gets the quotes associated to the specified workorder
POST api/Quotes?key={key}	No documentation available.

Comments

API	Description
PUT api/Comments/WorkOrder/{workorderId}	Inserts a comment associated to a particular Work Order
GET api/Comments/WorkOrder/{workorderId}	Get the list of Comments associated to a given Work Order
GET api/Comments/WorkOrder/{workOrderId}/Subject	No documentation available.
GET api/Comments/WorkOrder/{workOrderId}/CommentPreInsert	No documentation available.

API	Description
PUT api/Comments/WorkOrder/{workorderId}/Comments	Inserts a comment associated to a particular Work Order
POST api/Comments?key={key}	No documentation available.

Attachments

API	Description
GET api/Attachments/WorkOrder/{workOrderId}	Gets the list of Attachments associated to a Work Order
POST api/Attachments/WorkOrder/{id}	Allow the Submit / Post of attachments to the specified WorkOrder
PUT api/Attachments/{attachmentId}/Rename?newName={newName}	Snow App Renames an existing attachment
DELETE api/Attachments/{attachmentId}	Snow App Renames an existing attachment
GET api/Attachments/Attachment/{attachmentId}	No documentation available.
POST api/Attachments?key={key}	No documentation available.

BaseInternalApi

API	Description
POST api/BaseInternalApi?key={key}	No documentation available.

BaseUserInternalApi

API	Description
POST api/BaseUserInternalApi?key={key}	No documentation available.

.Net Sample

Add authentication information to request headers

GetHttpClient(), GetAsync(),GetAsyncCall() are defined in WebAPIProxy class
Sample:

```
private string baseAddress = “http://demo-api.fmpilot2.com/Internal/api/”

private const string authTokenKey = "authenticationToken";

private const string clientKey = "callingClient";

private const string domainKey = "actingDomain";



private HttpClient GetHttpClient(string authTokenValue, string clientValue, string domainValue, string contentType)

{

    HttpClient client = new HttpClient();

    // Set the Header values

    client.DefaultRequestHeaders.Accept.Clear();

    client.DefaultRequestHeaders.Add(authTokenKey, authTokenValue);

    client.DefaultRequestHeaders.Add(clientKey, clientValue);

    client.DefaultRequestHeaders.Add(domainKey, domainValue);

    client.DefaultRequestHeaders.Accept.Add(new MediaTypeWithQualityHeaderValue(contentType));



    // Set the base address

    client.BaseAddress = new Uri(baseAddress);

    return client;

}
```

Creating Result Class

Sample:

```
public class States

{

  public string Id { get; set; }

  public string Name { get; set; }

}
```

Executing StateLookup(GET api/Values/StateLookup)

Sample:

```
WebAPIProxy webAPIProxy = new WebAPIProxy();

WebApiResult List<States> result = webAPIProxy.GetAsync List<States> (authTokenValue, sourceDomainValue, domainValue, "Values/StateLookup", "application/json");

public WebApiResult<T> GetAsync<T>(string authTokenValue, string clientValue, string domainValue, string actionName, string contentType)

{

    WebApiResult<T> output = new WebApiResult<T>();
```

```
        try
        {
            var result = GetAsyncCall<T>(authTokenValue, clientValue, domainValue, actionName, contentType);
            result.Wait();
            return result.Result;
        }
        catch(Exception e)
        {
            output.StatusCode = System.Net.HttpStatusCode.InternalServerError;
            output.ErrorMessage = NoErrorMessage;
            return output;
        }
    }

    private async Task<WebApiResult <T>> GetAsyncCall<T>(string authTokenValue, string clientValue, string domainValue, string actionName,
string contentType)
    {
        Func<Task<WebApiResult<T>>> valueFactory = async () =>
        {
            WebApiResult<T> output = new WebApiResult<T>();
            HttpClient client = GetHttpClient(authTokenValue, clientValue, domainValue, contentType);
            HttpResponseMessage response = await client.GetAsync(actionName).ConfigureAwait(false);
            output.StatusCode = response.StatusCode;
            if (response.StatusCode == System.Net.HttpStatusCode.OK)
                output.Result = await response.Content.ReadAsAsync<T>();
            else
            {
                output.ErrorMessage = await response.Content.ReadAsStringAsync();
                if (string.IsNullOrWhiteSpace(output.ErrorMessage))
                {
                    output.ErrorMessage = NoErrorMessage;
                }
            }
            return output;
        }
    };
```

Abbreviations

List of Abbreviations:

Service Provider	SP
Work Order	WO
Facility Source	FS
Uniform Resource Identifier	URI
Application Program Interface	API
Information Technology	IT
Hypertext Transfer Protocol	HTTP
Secure Sockets Layer	SSL
Representational State Transfer	REST
JavaScript Object Notation	JSON
Uniform Resource Locator	URLs
eXtensible Markup Language	XML