API SDK Overview

for

Version 4.5

Part Number E62375-01

TOA Technologies 2014

Table of Content

This document contains proprietary and confidential information of TOA Technologies and shall not be reproduced or transferred to other documents, disclosed to others, or used for any other purpose other than that for which it is furnished, without the prior written consent of TOA Technologies. It shall be returned to TOA Technologies upon request. The trademark and logo of TOA Technologies are the exclusive property of TOA Technologies, and may not be used without permission. All other marks mentioned in this material are the property of their respective owners.

1 Introduction

1.1 Document Purpose

The document is intended to provide overall understanding of ETAdirect SDK documents.

1.2 Scope of the Document

The document provides basic overview of ETAdirect and describes overall SOAP and REST API idea, the use and applications related to each specific API. The document does not cover any front-end and back-end applications descriptions or detailed information on the specific API usage and transactions, as the information is provided in the correspondent SDKs.

1.3 Target Audience

The document was designed for those, who work with ETAdirect SDK documentation.

2 Application APIs

2.1 ETAdirect API Overview

ETAdirect APIs supports integration of ETAdirect with the corporate systems of a client company, enabling a wide variety of actions, from getting data into ETAdirect to integration into a client's existing mobile application.

2.2 API-related ETAdirect Entities

2.2.1 Users

Initially the system has to be filled with users and resources.

User is a person, group of people or software, accessing ETAdirect with a single login to the system. Each user in ETAdirect can manage only resources assigned to the user. User can be assigned several Security Profiles that define a set of permissions determining what the user can see and manage in the system.

Each time any SOAP-API transaction is run in ETAdirect, the user authentication is run. Basically, authentication is a process of verifying that the user is in fact who they claim to be. Authorization, or in other words, verification of the user's permissions within the system, occurs after successful authentication.

An Authentication check is described among common API structures of all the SOAP and REST APIs. More detailed description of the user processing is provided in the correspondent SDK of each specific API that processes the users.

2.2.2 Resources

When the system is implemented, it is populated with company specific resources.

Resource is one of the basic entities of ETAdirect which deals directly with activities.

There are several resource types in ETAdirect 4.5. The default types are – Group, Bucket, and Technician. At the same time, at the discretion of a company, more resource types can be created, while the default types can be removed.

Group can accumulate any resource type, including other groups, buckets, technicians and custom resource types. Technically speaking, Group is a resource with enabled 'Group of resources' feature and disabled 'Bucket' and 'Resource can execute activities' features.

Group can be a parent for group of resources, but cannot perform activities. Likewise, activities cannot be assigned directly to a group. An attempt to assign an activity to a group will implicate cloning of that activity, and assigning it to all technicians within the group (usually possible for internal activities). In this case, the activity to be assigned should have 'Allow mass activity' feature enabled.

Bucket can be a parent folder for any resource. Bucket is a resource with the 'Group of resources' and the 'Bucket' features enabled. Bucket can be compared to a dispatcher, who assigns tasks to technicians. It can be assigned activities, but cannot perform them. Only customer activities can be assigned directly to a bucket, while internal activities cannot.

Technician is a child entity of any group resource (e.g. bucket or group). Technician is a resource with the 'Resource can execute activities' feature enabled and the 'Bucket' features disabled. Technician performs the assigned activities and cannot include any sub-resources.
Specific activities are assigned to resources (if applicable) and resources are assigned to users, and only these users can manage the resource and its activities.
More detailed description of the resource processing is provided in SDK of each specific API that processes the resources.

2.2.3 Activities

Activities can be added to ETAdirect and then manually or automatically allocated among resources. Activity is a time-consuming action. By default, there are three types of activities: internal, customer, and team work. However, activity types are configurable. New activity types can be created, while the default types can be removed, if needed. More detailed description of the activity processing is provided in the correspondent SDK of each specific API that processes the activities.

2.2.4 Properties

Property is a variable associated with some ETAdirect entity (resource, user, activity, inventory, etc.) This is much like a field of an object in most programming languages. Property has a name, visibility, data representation, relations to entity and modification parameters. Therefore, if there is a property of an entity in ETAdirect, all such entities have this property, and all relevant API functions can read this property, write to it and make conditional decisions based on the property value (as long as correspondent permissions are granted to a user). And, correspondingly, all screens and functions, having access to the entity, can as well operate its properties (according to visibility restrictions).

Properties for ETAdirect entities can also be updated. A more detailed description of the property processing is provided in SDK of each specific API that processes the properties.

2.2.5 Inventories

Inventory is any equipment which is installed or deinstalled during an activity performance. Inventory could be virtually anything that is used in a job: from consumable material to a set of instruments, or devices. Inventory can belong to any resource or customer. Basically inventory is a property of an activity, but inventory in turn, can have own properties (e.g. type of inventory). A more detailed description of the inventory processing is provided in SDK of each specific API that processes the inventories.

2.2.6 Required Inventory

Required inventory is the inventory necessary to complete a certain activity. If any required inventory is defined for an activity, such required inventory is regarded as one of the criteria of activity assignment to resources. The required inventory is checked against the resource's inventory to see whether the resource's inventory is sufficient to complete the activity. If the resource has no required inventory in their pool, the activity will not be assigned to such resource. A more detailed description of the required inventory processing is provided in SDK of each specific API that processes the required inventory.

2.2.7 Work Skills

One of the properties defined for a resource is a set of work skills that the resource obtains and qualification level for each skill. A work skill is a task that a resource is qualified to perform. Work skills are assigned to technicians based on their training and knowledge. The names of the work skills may vary in different companies and industries.

Along with that, a set of rules is defined that enables automatic calculation of the work skills of each activity, their required and preferable levels. An activity should be assigned to a resource that obtains all of the work skills necessary to perform an activity with qualification level not less than the required level for the activity. A more detailed description of the work skills processing is provided in the correspondent SDK of each specific API that processes the work skills.

2.2.8 Work Zones

One of the properties defined for a resource is a set of work zones. A work zone is a defined geographical area where a resource can perform their service. A temporary work zone is a work zone

that is not a resource's typical work zone, but that is assigned to the resource for a period of time. Temporary work zones will override a resource's regular work zones.

Along with that, a set of rules is defined that enables automatic calculation of the work zone of each activity. An activity should be assigned to a resource that can perform work in the work zone of the activity. A more detailed description of the work zones processing is provided in the correspondent SDK of each specific API that processes the work zones.

2.2.9 Message Scenarios

ETAdirect Notification Module provides ability to create and trigger message scenarios. A Message Scenario is a collection of one or more Message Steps, defining message content, recipient, delivery protocol, and delivery business rules.

A Message Scenario defines a set of rules that tells how to do a specific notification. These rules are triggered by an internal event or certain conditions (Cancel, Complete, Change, etc).

Message Scenarios enable the following actions to be executed (by default):

- Sending an email
- Making an outbound voice call
- Changing a property of a technician, an activity or a user
- Sending a message to an external system

More actions can be configured according to a company's requirements. A more detailed description of the message processing is provided in the correspondent SDK of each specific API that processes messages.

2.2.10 Service Request

Service request is a message generated as the result of the 'send Service Request' operation and assigned to a specific entity in ETAdirect. A more detailed description of the service requests processing is provided in the correspondent SDK of each specific API that processes the service requests.

2.2.11 Route

Route is a list of activities assigned to a resource for a specific date, or a list of non-scheduled activities assigned to a resource. A route may contain zero or more activities.

One workday of one technician defines a route. Any references to the 'date' of the route mean the date of the workday start (e.g. if the resource works overnight).

Initially the route is formed in accordance with the resource calendar for every active resource with the working time scheduled for the date. The route of a resource can be filled with activities. When a user to which such resource is assigned has logged into the system**,** the user can manage the route. A more detailed description of the route processing is provided in the correspondent SDK of each specific API that processes the routes.

2.3 SDK Documents

In order to provide the client or a third-party integrator with the information and requirements for integration and interaction with TOA's APIs on the TOA part (if needed), the SDK documents are

provided to the client.

Each SDK document is divided in several sections. Most documents contain the following sections:

#	Description	Details
1	Introduction	Purpose and scope of the document
		Target audience
		Glossary
2	API overview	API general description
		Supplementary terms and definitions
		List of methods used by the API
3	Entities description	Description of common entities and structures processed by the API
4	Detailed methods description	Method name and aim
		Request and response parameters
		Request and response examples
5	Transaction errors	Error codes and error message examples
6	Version to version alterations	Description of changes implemented to the API in the described ETAdirect version compared to the previous versions

2.4 ETAdirect API List

The following APIs are available for ETAdirect. The APIs below are listed in the alphabetical order:

2.4.1 Activity Management API

ETAdirect Activity Management Application Programming Interface can be used throughout the activity lifecycle and enables integration of the ETAdirect activity management functionality in any software regardless of the platform or technology used, providing efficient tool for dynamic management and update of activity-related properties. Basic entities processed with the Activity Management API are activities and activity properties.

The Activity Management API provides methods to manage routes and activities, creating and processing them, and changing their details, namely:

- Retrieve properties of the activities in the specified resource's route for a specified day, as well as start, restart, or end the route
- Create activities with the specified properties assigned to the specified resource. Manage the resource's route by changing activity statuses and their start and end times
- Alter activity details applicable to a given activity regardless of its status
- Retrieve specified activity details and work skills
- Define, retrieve, and delete required, preferred, and forbidden resources to perform the specific activity
- Define, retrieve, and remove specified dependencies between two specified activities
- Define a file property, retrieve details of the specified property, and delete the file property with the specified label
- Define, retrieve, and delete required inventory for (from) the activity.

Detailed information on the methods used by Activity Management API can be found in the corresponding Activity Management SDK.

2.4.2 Capacity Management API

The function of the Capacity Management API is to:

- Extract all data available in the Quota View of ETAdirect
- Set or update the quota parameters
- Retrieve, set, or update the time when the quota is to be closed automatically
- Determine quota available for an activity to be booked and provide this information to the system where this booking is actually performed

Detailed information on the methods used by Capacity Management API can be found in the corresponding Capacity Management SDK.

2.4.3 External Trigger API

The External Trigger API is designed to trig a message scenario from an external system.

External Trigger API provides the ability to:

- Find an activity in ETAdirect based on a key field (e.g. customer phone)
- Trigger a message scenario from an external system to generate a message for this activity
- Set final status for this generated message, optionally executing subsequent steps of the message scenario, and optionally changing properties of an activity and/or activity status.

Detailed information on the methods used by External Trigger API can be found in the corresponding External Trigger SDK.

2.4.4 Positioning API

Positioning API supports integration of the ETAdirect system with external GPS-based systems, providing position tracking of GPS-enabled devices.

Positioning API allows to:

- Enable the ETAdirect system to obtain information about last known position of objects specified by identifiers and history of their movements
- Enable the external system to update information about the last known position of objects specified by identifier

Detailed information on the methods used by Positioning API can be found in the corresponding GPS SDK.

2.4.5 History API

History API serves as an advanced means of retrieving history logs and their details for further use by external applications.

History API uses one method to retrieve history records of changes to the following entities of ETAdirect:

- Route
- Activity
- Activity link
- Resource preference
- Required inventory
- Inventory
- Service request

Detailed information on the methods used by History API can be found in the corresponding History SDK.

2.4.6 Inbound API

The Inbound Interface is used to import data from a client external system to ETAdirect.

Inbound Interface is implemented as SOAP function for the following purposes:

- Set activities for a specific day for all resources or resource groups in ETAdirect
- Add new activities to ETAdirect
- Update, reassign, reschedule, and cancel activities in ETAdirect

- Change an activity status in ETAdirect
- Delete activities from ETAdirect
- Set inventory for resources and activities in ETAdirect
- Update or delete specific inventory in ETAdirect

  The upload can differ in the object uploaded: activities or resource inventories; and in the scale of upload: full or incremental.

Detailed information on the methods used by Inbound API can be found in the corresponding Inbound Interface SDK.

2.4.7 Mobile Client API

The Mobile Client Interface is a set of low-level functions that allow full control over activities, service requests and inventory in ETAdirect and allows a third-party to create a mobile field application that enables the functionality to:

- Retrieve properties of a specified user
- Retrieve properties of specified resource's route for a specified day; start, restart, or end specified resource's route
- Create activities with the specified properties assigned to the specified resource. Manage the resource's route by changing activity statuses and their start and end times
- Alter activity details applicable to a given activity regardless of its status
- Retrieve specified activity details and work skills
- Define, retrieve, and delete required, preferred, and forbidden resources to perform the specific activity
- Define, retrieve, and remove specified dependencies between two specified activities
- Define a file property, retrieve details of the specified property, and delete the file property with the specified label
- Define, retrieve, and delete required inventory for the activity
- Create and delete inventory assigned to the resource or activity; retrieve inventory details of the specified resource activity
- Install, deinstall, exchange, and update inventory
- Create and retrieve data on service requests assigned to a specified resource, activity, and inventory in ETAdirect
- Retrieve all properties of the specified service request.

Detailed information on the methods used by Mobile Client API can be found in the corresponding Mobile Client SDK.

2.4.8 Outbound API

Outbound API is used for interaction between the ETAdirect message engine and external Client Application. Client Application is a software that needs to be developed in order to integrate ETAdirect with external system(s). Client Application SOAP Service implements the following operations called

by ETAdirect:

- Send messages to Client Application
- Check if the message is still being processed
- Remove message from the agent internal queue, if the need in such message no longer present
- Optionally changing properties of an activity and/or activity status.

Detailed information on the methods used by Outbound API can be found in the corresponding Outbound Interface SDK.

2.4.9 Parts Catalog API

Parts Catalog API provides a set of functions used to manage the ETAdirect Parts Catalog. They are as follows:

- Create a new catalog
- Start and upload transaction for an existing catalog
- Upload data to an existing catalog
- Close an upload transaction for an existing catalog
- Abort the current transaction and delete all data uploaded during such transaction
- Search for existing catalogs
- Upload item weights to an existing catalog
- Delete item weights from an existing catalog
- Update inventory types or the item types in the catalog
- Delete all catalog data
- Get the list of transactions for an existing catalog

Detailed information on the methods used by Parts Catalog API can be found in the corresponding Parts Catalog SDK.

2.4.10 Resource Management API

Resource Management API is designed as a means of managing resources and users. With the methods of the Resource Management API it is possible to carry out the following processes:

- Add a new user to ETAdirect, define user properties, resources and profiles assigned to the user
- Update and delete an existing user
- Retrieve data about the specified existing user
- Retrieve data on the specified properties of all users assigned to the specified resource and its child resources
- Add a new resource to ETAdirect, define the resource properties and users the resource is assigned to

- Update an existing resource, retrieve specified data of a specified existing resource
- Retrieve the specified data on all child resources of the specified resource
- Retrieve details and define the working time, shifts, schedules and non-working time of a specific resource for a specific date
- Define and retrieve work zones and work zone details of a specific resource for a specific date
- Define, retrieve, and delete work skills of a specific resource
- Create or update resources' locations, delete existing locations
- Retrieve locations assigned to the specified resource
- Assign 'start', 'end' and 'home zone center' locations to resources or to remove the existing location assignments from resources
- Retrieve and delete 'start', 'end', and 'home zone center' locations assigned to the specified resources.

Detailed information on the methods used by Resource Management API can be found in the corresponding Resource Management SDK.

2.4.11 SelfCare Functionality

SelfCare functionality is used to support the integration of ETAdirect with the corporate web-site of the ETAdirect client company that renders services to a customer. SelfCare functionality supports the ability to:

- Retrieve the list of activities with the specified value in the specified field for the specified time period
- Alter the specified activity properties in ETAdirect
- Retrieve file property details from ETAdirect
- Retrieve information of a specified existing resource.

Detailed information on the methods used by SelfCare Functionality can be found in the corresponding SelfCare Interface SDK.

2.4.12 Location API

The Location API provides a simple and structured way to communicate with ETAdirect service. This API allows to remotely update ETAdirect Location-Based Services to change their storages with resource data from the third-party providers that is needed to search for locations. This API provides a complete set of methods to monitor the location / traces of resources the client uses for ETAdirect, namely:

- Update the service with the current location of resource
- Retrieve the last location which was set for the resource
- Retrieve the custom properties which were set with the attributes parameter for the last location of the resource
- Return the history of positioning for the given resource during the specific period of time
- Filter the history by applying the algorithm reducing the number of points in a curve, that is

approximated by a series of points

- Retrieve the information about the location of resources within the area defined by the longitude, latitude, and radius parameters.

Detailed information on the methods used by Location API can be found in the corresponding SmartLocation SDK.

2.4.13 Voice API

Voice API is used to initiate the outgoing calls to the customers and notify them about their upcoming appointments. Its functionality is considerably simpler than the one of the Outbound API for this purpose, due to the fact that a lot of the functionality has already been implemented on the ETAdirect side. They are as follows:

- Prioritization of pending calls waiting in the queue
- Controlling of a maximum number of parallel outbound calls
- Controlling of a maximum number of parallel call transfers
- Using different servers for different types of calls
- Adding a necessary prefix to a phone number to be dialed
- Validating a time period to be delivered to customer.

Details on Voice Integration with ETAdirect can be found in the corresponding ETAdirect Voice Integration manual.

3 Implementation Prerequisites

Implementation is the installation of ETAdirect system and its further configuration in accordance with a specific client's specifications, business rules, and requirements. Prior to implementation the following prerequisites have to be provided:

Prerequisite	Details
ETAdirect system	the system has to be configured and run in compliance with the client's specification
Network environment	the environment has to be configured in order to get access to ETAdirect system: VPN created, firewalls configured, etc.
Connectivity testing	network connectivity tests have to be performed: connection to the right port, request of the wsdl
URLs, ports, user logins and passwords	should be received from TOA by the customer and, for the Outbound API, should be received from the customer by TOA
Permissions and access details	have to be defined in the system as described in the section below

3.1 Permissions and Access Details

In a course of integration, 'user' entities are created. The 'user' entity is used for authentication and authorization purposes. For each user certain security and visibility settings are defined. The security settings define if the user is permitted to access a certain part of the system, and visibility settings define if the user can/must define or receive certain properties.

3.1.1 Security Settings

The security settings define if a user is permitted to access a certain part of the system. Security settings are defined as a set of Security Profiles, permissions of which are summed up. In other words, a user is allowed to access any part of the system that is permitted for any of its Security Profiles.

Following, is the list of APIs and methods, access to which is controlled by permissions:

- Activity Management API and its methods (Permissions → API → Mobile Client)
- Capacity Management API and its methods (Permissions → API → Get Capacity, Get Quota Close Time, Get Quota Data, Set Quota, Set Quota Close Time)
- Inbound API – has an only method (Permissions → API→ Inbound Interface)
- Mobile Client API and its methods (Permissions → API → Mobile Client)
- Outbound API (has the only method that is processed on the ETAdirect side: Permissions → API → Set message status)
- Parts Catalog API and its methods (Permissions → API → Parts Catalog)
- Resource Management API and its methods (Permissions → API → Resource management)
- External Trigger API (Permissions → API)

- History API (Permissions → API → History API)
- SelfCare Functionality (Permissions → API → Mobile client (Search Activities, Update activity, Get File) and (Permissions → API → Resource Management → Get resource)

Depending on specific client objectives, security settings provide a variety of access level options.

For example, there can be one Security Profile created to control access to all the APIs and methods (not the most secure option; may typically suit smaller companies). At the same time, there can be a specific Security Profile created for each particular Interface (best-practice). If required, a Security Profile can even restrict the access to a set of specified methods within a single API.

The Security Profile creation is available to implementation and support teams, that can set up a corresponding Security Profile or Security Profiles upon a client's request.

3.1.2 Visibility Settings

The visibility settings define whether a user is required to process certain properties.

If an Interface and its methods are permitted for the user by security settings, the user may initiate transactions in the Interface (send requests and receive responses). The elements of such requests and/or responses are properties of the entities processed in ETAdirect.

API Profile defines the list of properties that are available for the user within the Interface and the visibility rules for those properties.

A property can be set to hidden and will not be seen in any way by the user. On the other hand, visible properties can be mandatory for the request to be processed correctly, or optional.

In the SDK documentation for correspondent APIs, the following structure for defining property visibility is used:

Optional: the user can see the property and can optionally manage it. The 'Required' column contains 'No' for such property.

Mandatory: the user can see the property and must define it. The 'Required' column contains 'Yes' for such property.

– If the transaction contains an invalid mandatory property, the request is rejected with a corresponding error

– If the transaction has no mandatory property, the request is rejected with a corresponding error.

API Profiles can be configured by going to Company Settings → API Configuration and selecting the needed API profile. Upon clicking on the Layout link, Context layout list will appear where the following can be configured:

- **Activity API** – for properties assigned to an activity (Add Activity/Activity details, Soap set user)
- **Inbound API** – for properties assigned to an activity and inventory (Add/Edit Activity, Add/Edit inventory, Resource Info)
- **Mobile Client API** – for properties assigned to an activity, inventory and service request (Add activity/Activity details, Soap set user, Mobile client request, Mobile client inventory)
- **Resource Management API** – for properties assigned to a resource (Soap set user, Mobile

Client resource)

- **SelfCare Functionality** uses the same contexts as Activity Management API and Resource Management API
- **For the rest of APIs** all the fields and properties are visible (as long as the API is permitted by Security settings. The API profile creation is available to implementation and support teams, which should set up a corresponding API profile or API profiles upon a client's request.

3.2 System Requirements

ETAdirect Interfaces work via HTTPS protocol. As such, they require an environment that supports SOAP 1.1 and HTTPS protocols:(see http://www.w3.org/TR/2000/NOTE-SOAP-20000508/). Actual hardware and operating system platforms are usually not a constraining factor. ETAdirect APIs can be used with a variety of technologies, including but not limited to Java, .Net, and C/C++, on both Windows or UNIX platforms.

4 Technical Support

TOA ETAdirect professionals are available for technical support concerning all questions and troubleshooting related to the development of TOA-compliant applications.

Development support is available during customer business hours to TOA customers during the SDK active development phase and is designed to support the customer through their development cycle.