ALPHAREN CORE-Integrator (ARINT) System

Product features

Table of contents:

General features

- Open API full compliant
- Integrations, Microservices, SOA and ESB in Python
- high availability load-balancer
- hot-deployment and hot-reconfiguration - deploy with no downtime
- Browser-based GUI, CLI
- API - easy to use and customize

For an overview of rhe product please see.

Standards compliance

The following list presents some public (most usual) standards, protocols, data stores, formats, and so on, *that ARINT system can use and is compliant with*.

The actual list is larger and practically *any kind of specific interface* can be assured by services as long as is written the corresponding code.

Standards update

Also verify your system version and update it as new standards can be included in official package releases.

> The order of items is not relevant (meaning that it does not pursue a specific objective). Also classification made is not *an exact one* as some items can be categorized in more places. If someone know exactly what standard is looking for, a traversal of the entire list is best option.

- Protocols:
  - REST
  - SOAP
  - FTP
  - SFTP
  - LDAP
  - Active Directory
  - WebSockets
- Industry standards:
  - HL7 (healthcare - data exchange)
  - RBAC (IT - Role Based Access Control)
  - Swift (banking - Society for Worldwide Interbank Financial Telecommunication)
- Business systems:
  - SAP
  - Odoo
- Mail and messaging protocol and systems:
  - SMTP
  - IMAP
  - Telegram
  - JMS
  - Twilio
  - Slack
- Data languages and formats;
  - OpenAPI
  - SQL
- Databases and broker systems:
  - MongoDB
  - Redis
  - Memcached
  - Cassandra
  - Kafka

- Search systems:
  - ElasticSearch
  - Solr (Apache)
- File oriented stores and depots:
  - Amazon S3
- Queue based communication systems:
  - AMQP
  - IBM MQ
  - ZeroMQ
  - JMS (Java Message Service)
- Security and protection:
  - Vault
  - all cryptography standard algorithms

Services level issues and features

In writing an ARINT services there are some frequent and repeating issues. An ARINT service address them by offering built-in solutions and ***let developer focus on business aspects***. Addressed issues are:

- How do I connect to ARINT own resources ? How do I send user entered credentials to ARINT?
- How do I connect to external systems or databases to store or extract data?
- How do I convert and / or map messages from one format to another?
- How do I automate my work that it is repeatable in most cases (for example across environments like `development`, `test`, `production`)?
- How can I focus on my job instead of thinking of trivial low-level details?
- How do I debug my code?
- How can I reuse the skills and knowledge that I obtained?
- How do I manage the complexity of having to integrate at least several systems and dozens or hundreds of APIs?

All these questions find answers especially in "Development documentation", Development Overview being a good start.

Frequent service type examples

Here are some examples of what kind of services can be accomplished by ARINT:

Most popular

Technologies most commonly used in API integrations: *REST | SOAP | Scheduler | Pub/sub | SFTP | WebSockets*

Databases and message queues

How to access commonly used data sources: *SQL | MongoDB | Redis | AMQP | IBM MQ | ElasticSearch*

Business apps

Integrating with 3rd party CRM and ERP software: *Microsoft 365 | Salesforce | Odoo*

Health care interoperability

Integrating health systems meaningfully: *HL7 FHIR | Banking Security Systems*

E-mail

How to send and receive emails: *IMAP | SMTP | Microsoft 365*

Shell commands

Turning shell commands into API services: *Shell commands | SSH | PowerShell*

File integrations

Integrating systems using continuous or batch file transfer: *File transfer | SFTP | FTP*

Requests, responses and data models

Convenient access to request and response data: *Requests | Responses | Data models | OpenAPI*

Cloud integrations

Integrating with popular cloud providers: *AWS S3 | Jira | Confluence | WordPress*

LDAP and OAuth

Integrating with external security providers: *LDAP and Active Directory | OAuth and REST*

These are only some kind of type of supplementary services that can be written in ARINT. More details about services will be found in Development - Service anatomy.

ARINT own objects and components

Besides the features offered as external systems integration, the ARINT has its internal objects and components (that will be detailed in other documents). These are:

- **System database** containing all ARINT meta-information about its objects
- **Business database** containing all business entities and objects with their effective data
- **Master data** containing all "cross instances" master data objects (more exactly master data objects that are agnostic to business entities and objects content)

A detailed description of *System Data & Objects*.

Last update: August 26, 2023