DIGITAL INTEGRATION HUB

Moving from a discrete array of task-specific tools to an integrated set of functionalities

1 INTRODUCTION

The challenges posed by digital business transformation require a radical change in the way organizations deal with. Designing and implementing an integration strategy for the digital era's requirements for agility and adaptability, along with fast and pervasive integration, involve addressing technological and organizational issues

In most cases, organizations will have to:

- Renovate their integration platform strategy by moving from a discrete array of task-specific tools to an integrated set of functionalities
- Reshape their organizational model by moving traditional centralized integration teams from a "software factory" model to an empowerment approach to support hybrid integration platform-enabled.

The document describes our understanding about digital integration and implementation approach.

2 WHAT IS DIGITAL INTEGRATION HUB?

Digital Integration HUB (DIH) is generally advanced application architecture as a set of data stores, APIs, utilities, tools, rules, defining how to approach data persistent layers an that aggregates multiple back-end system of record data sources into a low-latency and scale-out, high-performance data store. DIH is also enabler for flexible reporting, data analysis and machine learning.

2.1 MAIN DIH IMPLEMENTATION BENEFITS

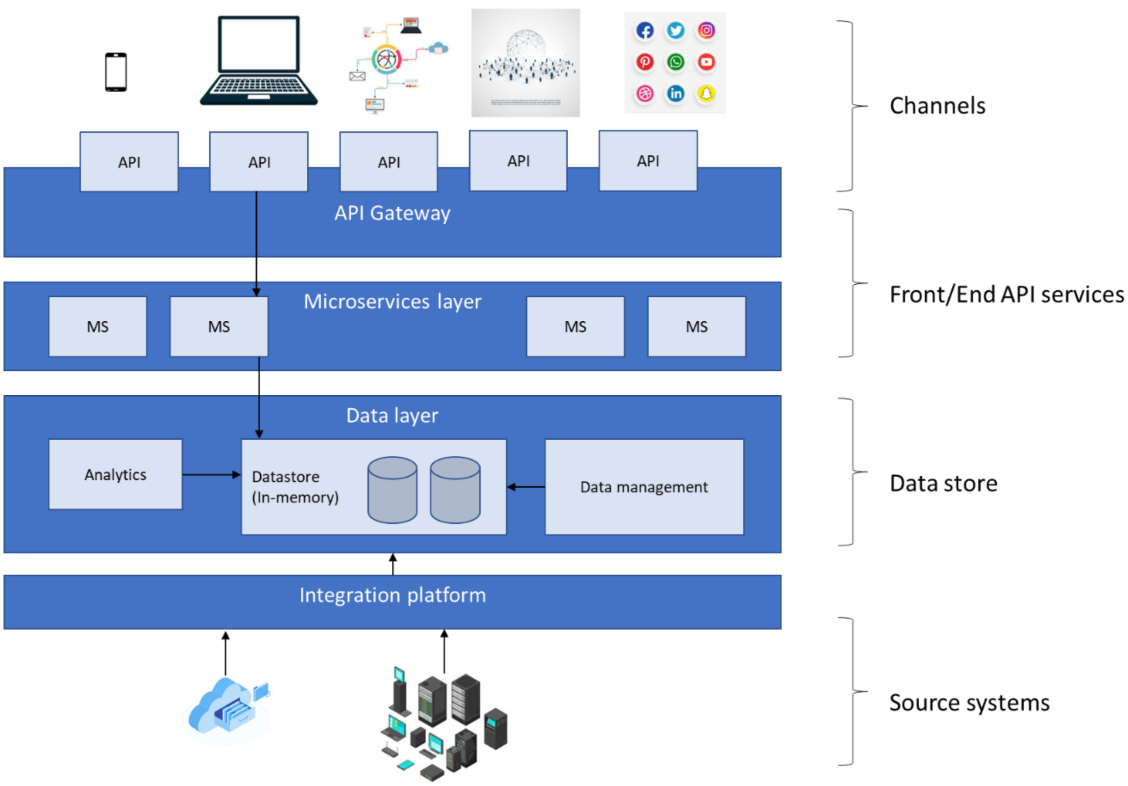
DIH basically allow company for fast building of new products and services, cause you never know what we will sell and bill the customer for. At the same time, the whole architecture should allow for stable user friendly and reliable environment. And also:

- Decoupling the back-end data sources from the front-end API services to allow multi-channel communication.
- Protect systems of record from excessive, low-value API workloads
- Enable front-end APIs to access multiple back-ends
- Maintain an up-to-date, consolidated picture of fast changing data
- Keep cost of integration tasks on reasonable low level.
- Provide a responsive, low-latency, 24/7 API experience
- Capability to use synchronous communication via API, asynchronous data streaming via streaming platform and data replication.

3 DIH CONCEPT ACCORDING TO GARTNER

By Gartner definition DIH is: “an advanced application architecture that aggregates multiple back-end system of record data sources into a low-latency and scale-out, high-performance data store”.

Figure 1: Digital Integration HUB by Gartner



4 DIH KEY FEATURES

4.1 THE HIGH PERFORMANCE DATASTORE

Look at in-memory computing technology for high scale and low latency

- Holds a consolidate copy of the back-end data (data replication).
- Single view of customers, products, prices, itineraries, students and patients,
- Synchronized with one or multiple system of record applications and data sources.
- Low-latency/scale-out/high-availability

4.2 THE (HYBRID) INTEGRATION PLATFORM

Adopt a suitable mix of integration technologies (one may not be sufficient!). Keeps the high-performance data store and back-ends in synchro.

- Unidirectional:
    - Back-end to HP data store
- Bidirectional:
    - Back-end to HP data store
    - HP data store to backends
- Data load and data cache from core / legacy systems and synchronisation with high performed front ends
- Manage File Transfer (MTF):
    - Powerful automated file transfer solution
    - Detailed auditing and reporting tools
    - All with the streamlined user management, transfer flexibility and intelligent security

4.3 API GATEWAY

- Centralised single point of management, orchestration and security for all internal and external integration (internal and external API GTW)

4.4 ANALYTICS

Add analytical capabilities to increase the business value of DIH. Data in the high-performance data store can also be used for analytics:

- Search
- Data visualization
- Predictive analytics
- Optional. At times, the primary adoption driver

5 DIH IMPLEMENTATION – GSW DEVELOPMENT SUCCESFUL STORY

GSW development (GSW) has successfully prepared, implemented and launch of the DIH solution from scratch for retail integrated mobility provider in Europe market. All technologies, mentioned in DIH examples could be evaluated and changed according to customer situation and preference.

5.1 MAIN OBJECTIVES OF IMPLEMENATTION

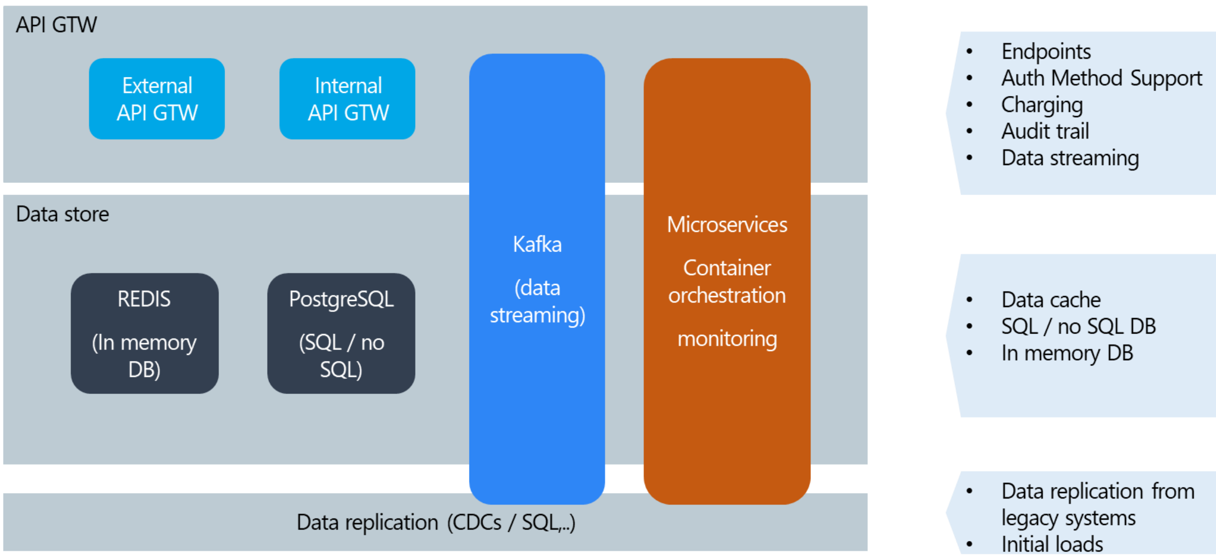
Main project / DIH Implementation targets/principles were specified by customer:

- Scalability of solution – for peak performance
- Cost optimal – low/no licence costs and low maintenance costs
- Cloud agnostic – concept should support all major cloud providers
- Flexible for further development – broadly used SW development languages, configurable
- Container-based deployment for all DIH components
- GIT as version control system to provide single source of truth
- Infrastructure as code via Terraform to provision and manage any infrastructure
- Fully automated CI/CD pipeline via Azure DevOps

5.2 DIH GSW – LOGICAL ARCHITECTURE

Based on our experience with DIH platform implementation, we defined logical architecture based on integration, databases and replication components:

Figure 2: DIH logical architecture



5.3 DIH SOLUTION - KEY FEATURES

5.3.1 DATA REPLICATION

- Solution is based on change data capture (CDC) design patterns
- Support online replication from multiple data sources. Mostly from server databases to DIH platform data store (actually realised by *PostgreSQL*).
- Realized by custom developed replication component and for example *StreamSets Data Collector* (open source app)
- Replicated changes can be also published via data streaming APIs (*Apache Kafka*) for online event triggering, analytics etc.

5.3.2 DATA STREAMING

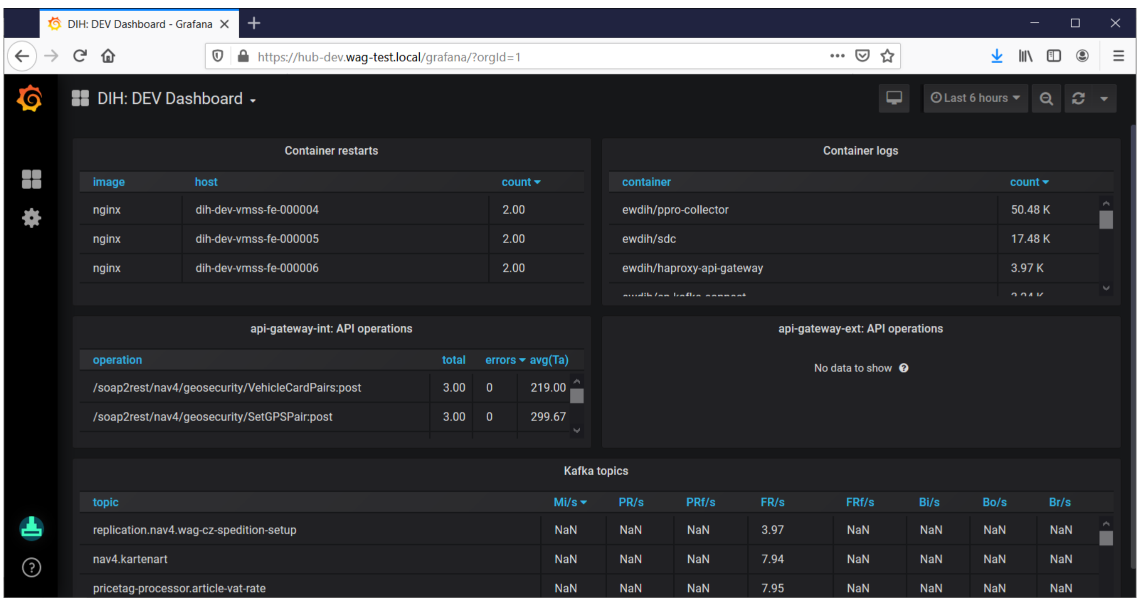
- Confluent Platform/Apache Kafka is part of DIH platform
- Publish-subscribe model
- Topics
- Producers publish data to topics

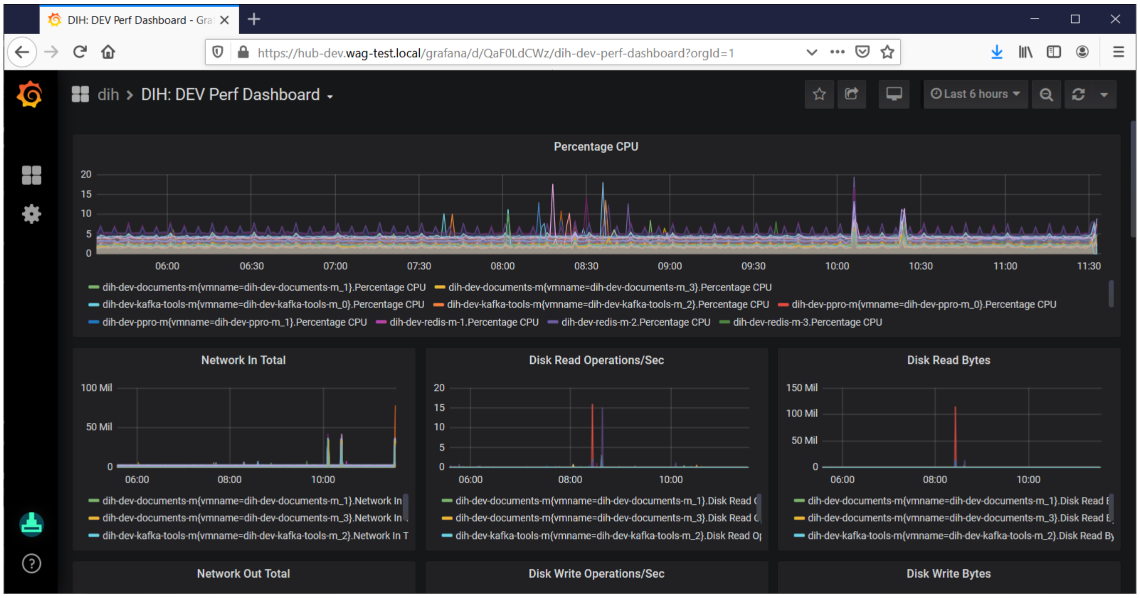
- Consumers/Consumer groups retrieve data from topics
- Secure access via SASLSSL protocol (preferred)
- Fast access via SASL_PLAINTEXT protocol (only from secure networks)

5.3.3 MONITORING

- Logs and operation metrics are aggregated to Azure Monitor‘s Log Analytics Workspace
- Each DIH environment has dynamically generated Operations and Performance Dashboards

Figure 3: Monitoring dashboard example





5.3.4 MANAGEMENT OF SECRETS

- Each DIH environment has separate set of secrets.
- Management of secrets implemented on top of:
  - *Mozilla SOPS* an open-source tool from Mozilla allowing to create and edit encrypted files.
  - *Azure Key Vault* a cloud-based tool for securely storing and accessing secrets. A secret is anything that you want to tightly control access to, such as API keys, passwords, or certificates. A vault is a logical group of secrets.

5.3.5 API MANAGEMENT

Essential component of API management is an API Gateway implemented on top of HAProxy, Lua scripting & Swagger UI

Each DIH environment has 2 independent API Gateways reachable on well-known URL endpoints:

- Internal API GW for APIs that SHOULD NEVER BE reachable from Internet
- External API GW for APIs that COULD BE reachable from Internet

DIH API management comprises of following tasks:

- **Discovery** - All published API definitions are compiled to a single Swagger / OpenAPI definition.

- **Publication** - APIs must be defined by machine-readable specification.
    - Currently Swagger 2.X and OpenAPI 3.X.Y specifications are supported.
    - Common URL structure for published API specification serves as a contract between DIH/API Gateway and published system.

- **Security**:
    - DIH API Gateway currently supports following authentication mechanisms:
        - SSL/TLS & mTLS
        - OAuth 2.0 / OpenID Connect
        - HTTP basic authentication (for legacy purposes)
    - Each API GW can be dynamically configured to trust any number of OAuth 2.0 authentication providers.
    - At the moment two authentication providers are configured on each DIH API Gateway:
        - CIAM (SSO) as provider of Customer entities
        - AzureAD as provider of employee / internal identities
    - DIH API Gateway supports following authorization mechanisms:
        - Scope-based access control (SBAC)
        - Role-based access control (RBAC)
        - Group-based access control (GBAC)
    - Authorization requirements are specified directly in security section of API specification provided by backend systems.

- **Rate-limiting** - API rate-limiting is a mechanism how DIH API Gateway is able to protect backend systems from overload or denial-of-service attacks. DIH API Gateway uses fairly advanced generic cell rate algorithm. Rate-limiting requirements are also specified in API specification provided by backend system.

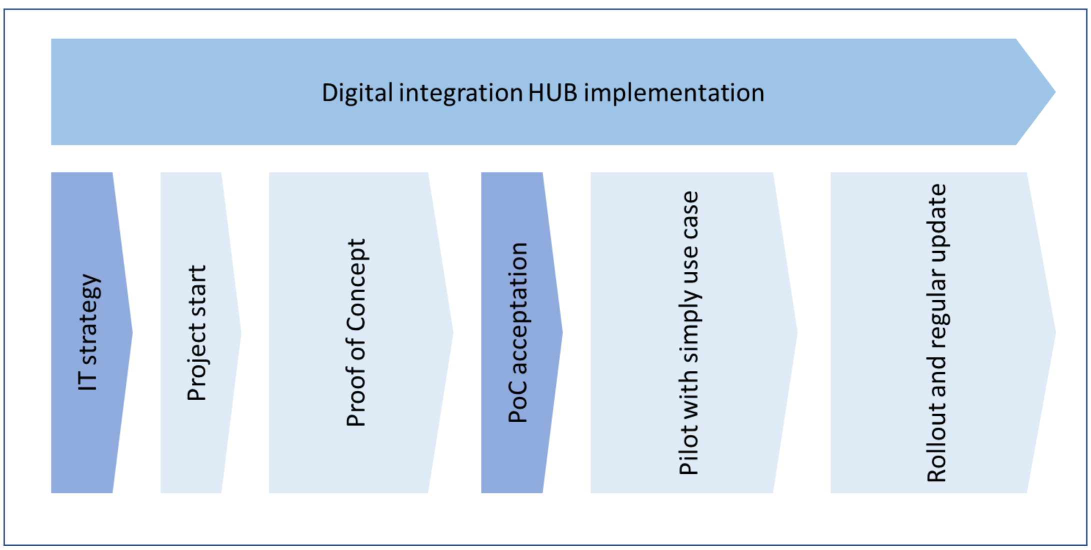
6 DIH IMPLEMENTATION - OUR IMPLEMENTATION APPROACH

DIH implementation is based on Iterative approach separated on five basic stages:

- Project start
- Proof of Concept preparation
- Proof of Concept acceptation
- Pilot
- Rollout and regular update

Recommendation for project start is prepared company IT strategy with defined requirements and patterns for digital integration and digital channels.

Figure 4: DIH implementation - basic stages



6.1 PROJECT START

Project initiation phase is focused on successful project start with project team mobilisation, project scope and targets definition, project plan preparation, risk management process definition and Kick off meeting execution.

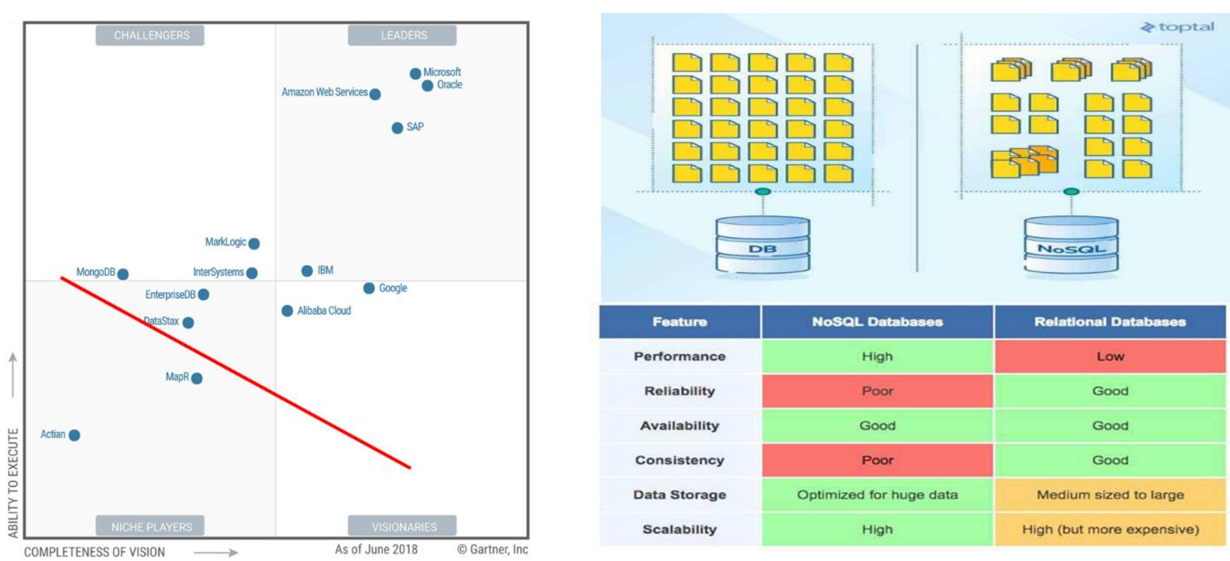
6.2 PROOF OF CONCEPT PREPARATION

Proof of concept stage would show whether an DIH concept is feasible from a technology standpoint. All (or main part of) future technology components will be evaluated and tested based on defined metrics.

Components/tools are selected in first iteration based on Magic Quadrants studies (Gartner or similar sources). Based on strengths and cautions analysis shortlist of components is prepared for detailed evaluation and testing.

IT architects will collaborate across project teams to prepare To-Be concept on regular basis including regular review of concept by IT architecture board.

Figure 5: Example of Magic quadrant for DB tool selection



Feature	NoSQL Databases	Relational Databases
Performance	High	Low
Reliability	Poor	Good
Availability	Good	Good
Consistency	Poor	Good
Data Storage	Optimized for huge data	Medium sized to large
Scalability	High	High (but more expensive)

Selected tools are tested into testing infrastructure to demonstrate functionality and determining whether DIH architecture idea can actually be built (at all or at reasonable cost), or finding the most effective technologies to use in the DIH development.

Basic documents prepared during PoC stage:

- Basic draft of DIH architecture principles

- DIH components/tools comparison and evaluation
- Design principles and integration patterns for existing applications
- Design principles and patterns for new applications
- DIH patterns
- Microservices patterns

6.3 PROOF OF CONCEPT ACCEPTATION

All project outputs from PoC stage are going to final review by Architecture board and acceptation process by company management.

6.4 PILOT

Pilot is an initial business requirement small-scale implementation that is used to prove the viability of a DIH PoC idea.

We will select simple business functionality (use case) which will be implemented into DIH platform to demonstrate business capability of concept.

Implementation could be on no-production environment (TEST, UAT,..) or including PROD pilot verification.

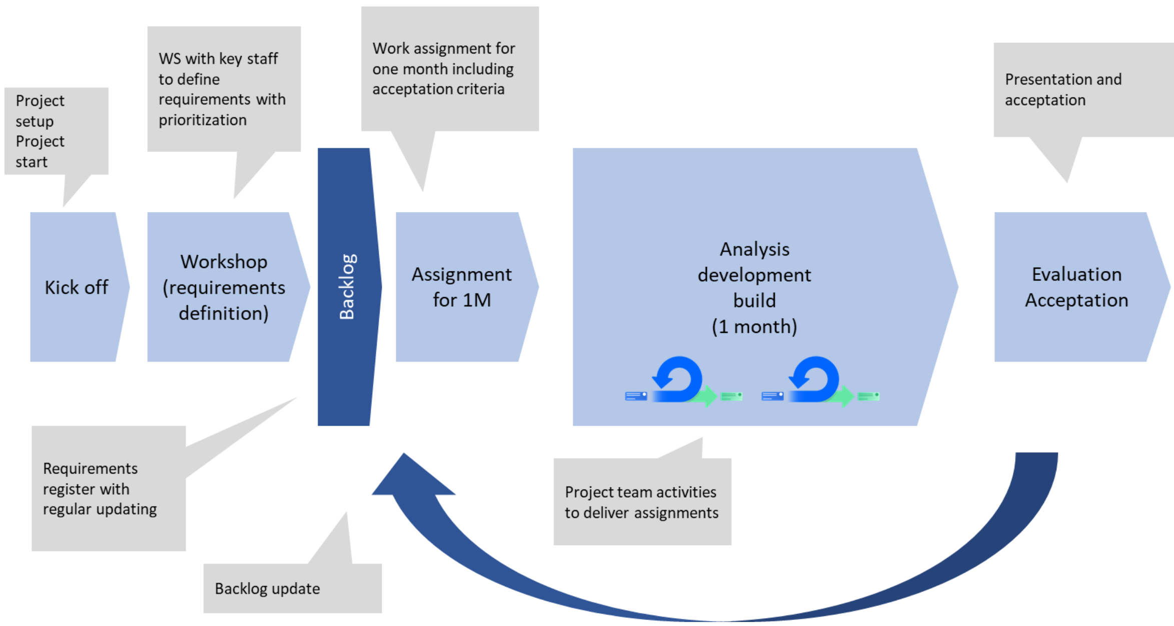
6.5 ROLLOUT AND REGULAR UPDATE

Final implementation stage is rollout of DIH platform into production, high availability support and regular enhancement and maintenance.

7 GSW AGILE APPROACH

GSW development team works generally in monthly iterations. Each iteration is based on two SCRUM sprints. Scope of work is defined and accepted before monthly iteration. Finalised iteration is accepted after functionality demonstration or documents review finalisation.

Figure 6: GSW SCRUM approach



8 GSW DEVELOPMENT REFERENCES

With Digital Integration Hub we gain control and capability to integrate not only state-of-the-art systems and applications but we are able to scale up legacy systems like MS Navision 4 (our heavily customized ERP, CRM and transaction processing system). Utilizing Digital Integration Hub we have capability to define and expose APIs to every WAG customer, partner or internal application and these entities can consume it in a secured way.

Digital Integration Hub also significantly contributed to time to market reduction. We are now able to integrate new application or system in days instead of months.

Radek Moc: Chief Technology Officer, board member

W.A.G. payment solutions, a.s.