



PERIGEE
SOFTWARE



Integration Implementation

W H I T E P A P E R

Table of Content

01	Overview
02	Why Should You Be Concerned About implementation?
04	Systems Integration Terms To Know
06	Types of Integrations
07	Best Practices
09	Summary
10	What is Perigee?



Overview

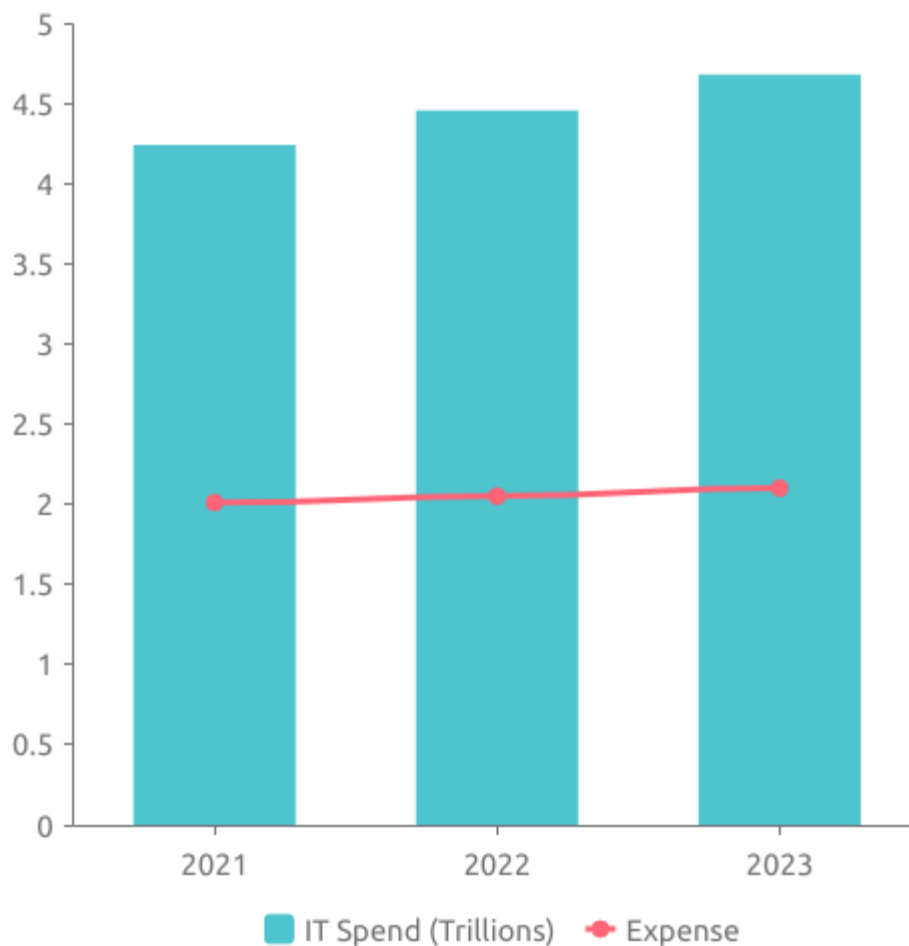
Ensuring resilient integrations and robust implementations can result in substantial yearly savings for your organization.

Adopting a simplistic strategy of just ***"making the integration function"*** will, in the long run, be more costly compared to initially crafting a sound and solid implementation.

Why Should You Be Concerned About implementation?

The lack of effective data integration, substandard coding practices, and inadequate implementation are believed to have contributed to a **staggering expenditure of over 2 trillion USD in 2020.** ¹

With global IT expenditure projected to reach 4.6 trillion in 2023 ², it becomes crucial to comprehend why a considerable segment of your budget is being utilized to address issues that originated within the IT sector itself.

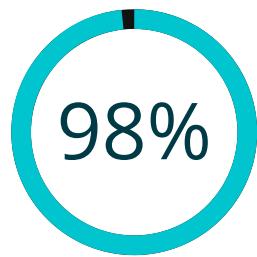


² <https://www.gartner.com/en/newsroom/press-releases/2022-01-18-gartner-forecasts-worldwide-it-spending-to-grow-five-point-1-percent-in-2022>

¹ <https://www.it-cisq.org/cisq-files/pdf/CPSQ-2020-report.pdf>

Why Should You Be Concerned About implementation?

A significant factor contributing to this remarkably high expenditure is that current solutions often fail to meet long-term requirements. Current "drag and drop" or "do-it-yourself" projects typically require rewriting anywhere between 2 and 10 times, leading to a staggering 98% probability of the integration needing to be reconstructed in the future.



Rewrite Integration



Avg. Number of Rewrites

" When they do use integration tools, IT decision makers often find that existing integration tools fall short of where they need to be in terms of usability, adaptability and speed to market.

Rebuilding integrations also is a common task. More than half of CIOs and 45% of system architects and developers said in the past 12 months they had to rebuild integrations for existing key business applications six to 10 times, while 98% of respondents have rebuilt integrations for existing key business applications in the past 12 months. " ¹

¹ <https://www.prnewswire.com/news-releases/majority-of-it-leaders-say-enterprise-integration-is-a-business-imperative-in-2022-but-only-7-have-an-established-strategy-according-to-digibee-report-301599524.html>

Systems Integration Terms To Know

Integration

- Integration refers to the method of enabling two systems to effectively transfer data or communicate with each other.
- Integrations are not exclusively for third-party vendor connections; they can also encompass in-network communications within your own system infrastructure.

Implementation

Implementation involves the translation of a business requirement into a working model through coding. This process can significantly differ based on various factors. A successful implementation is not solely determined by the chosen solution, but also by the tools used, the technological prerequisites, the skill level of the developers, the time restrictions, and the planning against potential failure points. All these elements significantly influence the effectiveness of an implementation.

API

An Application Programming Interface (API) represents a set of rules and protocols established for communication between different software systems. It acts as a contract, defining how data can be exchanged between systems, thereby enabling developers to interface with third-party services in a standardized manner. APIs essentially provide the guidelines for pushing and pulling data between different systems.



Systems Integration Terms To Know

Enterprise Service Bus

Enterprise Service Bus (ESB). The ESB integration leverages an integration infrastructure functioning as a central hub to interconnect various systems and applications. It promotes smooth interaction and data transfer among different components within an enterprise framework.

Message-Oriented Middleware

Message-Oriented Middleware (MOM). The MOM integration encourages asynchronous interaction and data transfer among systems through messaging. It supports reliable and flexible integration between distributed applications by handling message queues and guaranteeing message delivery.

Extract, Transform, Load

Extract, Transform, Load (ETL). The ETL integration process includes extracting data from one or multiple sources, modifying it to conform to the destination system's criteria, and subsequently loading it into the target system. It is typically utilized for activities such as data warehousing, data migration, and synchronization.



Types of Integrations:



API



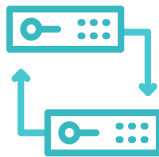
Database



ESB



ETL



Realtime



MoM



Best Practices: Integration Implementation

Integration and data applications necessitate a dedicated toolset and require engagement from across the business to define operational needs. Importantly, they also need a clear understanding of the requirements during system failures or outages to ensure business continuity and data integrity.

The principle is straightforward: The less emphasis your integration toolset places on disaster recovery, the more time you will inevitably spend mitigating the impacts of such disasters.



Focus on Failure

Merely focusing on task completion as a measure of success can lead to a rewrite as soon as critical business operations start to falter. The majority of your time should be dedicated to collaborating with the business and development team to strategize responses for potential critical faults at every layer of your integration.

Best Practices: Mitigating Loss

Data loss prevention cannot be accomplished by a single magic solution. Crafting smart designs, utilizing talented individuals, and equipping them with top-tier tools is an essential path towards achieving this goal.



Fault Tolerant Design

Creating an application with a failure-aware design, using tools that enable fault-tolerant recovery, will guarantee the smooth operation of your integration, even during challenging times.



Exceptional Recovery Plan

The recovery process should be initiated at the application level, by enabling the application to restart following a crash, resume its operations, and complete any pending transactions



Graceful Shutdown

The practice of graceful shutdown, which allows an individual task to preserve its state and conclude its operation before ending is crucial. Implementing procedures to respond appropriately to shutdown requests is essential in mitigating data loss.



Concise Direct Reporting

There will be instances where your application encounters an irreparable fault, and all disaster recovery strategies, retry mechanisms, and fault-tolerant designs have been depleted. In such situations, it is imperative to create a precise and straightforward report detailing the steps needed for human intervention to resolve the issue. Lengthy discussions involving the entire IT department are resource-draining and can be avoided with clear instructions.

Summary

The consequences of poorly integrated applications are vast and almost incalculable. The time and resources invested in rectifying issues, communicating failures and patches to clients, addressing system outages, and holding company-wide meetings to troubleshoot, all accumulate. The potential loss of clients due to system faults further amplifies this cost.

To avoid channeling a significant part of your budget into repairing poorly implemented integrations, consider the following:

- Engineer your systems with a capacity to handle failure.
- Employ talented developers and allot sufficient time for them to create robust integrations; the return on investment here is unfathomable.
- Upgrade to superior development toolsets that facilitate disaster recovery and fault tolerance.
- Have well-structured disaster recovery plans starting at the application level.

These steps can help create a solid integration strategy that minimizes disruptions and maximizes efficiency.

What is Perigee?

"According to our research, best-in-class tools are the primary driver of Developer Velocity. " ²

We created Perigee out of the necessity for a solution that could achieve the high performance and fault tolerance critical for a successful integration application, something missing from existing market offerings. Many existing platforms emphasize ease of implementation, often overlooking real-world scenarios such as intermittent connectivity, server faults, crashes, and failures.

Perigee represents a "best-in-class" toolset that enables a significant acceleration in development. Your organization stands to gain from this velocity enhancement as we've addressed the most common data exchange, integration, and processing patterns, resolving each issue meticulously. The result is the most robust, fault-tolerant, and resilient toolkit available in the market today.

Schedule a demo and download the free trial today at:

<https://perigee.software>

¹ <https://www.mckinsey.com/industries/technology-media-and-telecommunications/our-insights/developer-velocity-how-software-excellence-fuels-business-performance>