The **CEMLI** Framework

Configuration, **E**xtension, **M**odification, **L**ocalization, and **I**ntegration

Hi! This is **Demasy**

Ahmed El-Demasy is a Technology Architect with over 15 years of experience in Oracle technologies, applications, and the cloud. I specialize in implementing, upgrading, and customizing Oracle EBS applications.

Let's **Start!**

Contents

- CEMLI Overview
- CEMLI History
- Types of Customizations
- CEMLI Complexity Guideline
- CEMLI Benefits
- Applying CEMLI
- Applying Guidelines
- Conclusion

CEMLI CONTENTS

CEMLI Overview

#demasy

Oracle CEMLI is an acronym for the five customizations that can be made to an Oracle application: **C**onfiguration, **E**xtension, **M**odification, **L**ocalization, and **I**ntegration.

The CEMLI framework provides a structured approach to customizing Oracle applications to meet an organization's business requirements. It offers guidelines for managing customizations to the application while maintaining its integrity and stability.

The CEMLI framework provides guidelines for customizing Oracle applications, documenting customizations, and testing before deployment. By following the CEMLI framework, organizations can reduce the risk of system disruptions, make upgrades more effortless, and better align their Oracle applications with business requirements. The framework is designed to provide a standard approach to customizing Oracle applications, enabling organizations to make necessary changes without disrupting the underlying systems.

CEMLI OVERVIEW

The CEMLI framework benefits industries with unique business requirements that out-of-the-box Oracle applications may not meet. Every type of customization is distinct and requires a different approach, skill set, and level of effort. The CEMLI framework provides guidelines for managing each customization type while maintaining consistency and repeatability.

CEMLI OVERVIEW

CEMLI History

Oracle Corporation introduced the CEMLI framework in the late 2000s to help organizations manage customizations to their Oracle applications. Before its introduction, customizations to Oracle applications were often made ad hoc, leading to a fragmented and unstable system that was difficult to maintain and upgrade. The CEMLI framework provided guidelines for managing customizations and emphasized the importance of following a structured approach, focusing on the five types of customizations: Configuration, Extension, Modification, Localization, and Integration.

The CEMLI framework was initially defined in Oracle's E-Business Suite and was later expanded to include other Oracle applications, such as PeopleSoft, JD Edwards, and Siebel CRM. Today, the CEMLI framework is widely used by organizations worldwide that use Oracle applications, and it has become a best practice in the industry for managing customizations to enterprise software.

CEMLI HISTORY

The history of **Oracle CEMLI** dates back to the early days of Oracle's development of enterprise software applications. The need for customizations to meet specific business needs increasing as the Oracle software suite grew more extensive and complex. To address this need, Oracle introduced the CEMLI framework, which has since become an essential tool in Oracle's application development and customization process.

The CEMLI framework continues to evolve as new technologies emerge and Oracle expands its software offerings. It is a vital tool for organizations that need to customize their Oracle applications to meet their unique business requirements while minimizing the risk of system disruptions and maintaining the integrity of their core applications.

CEMLI HISTORY
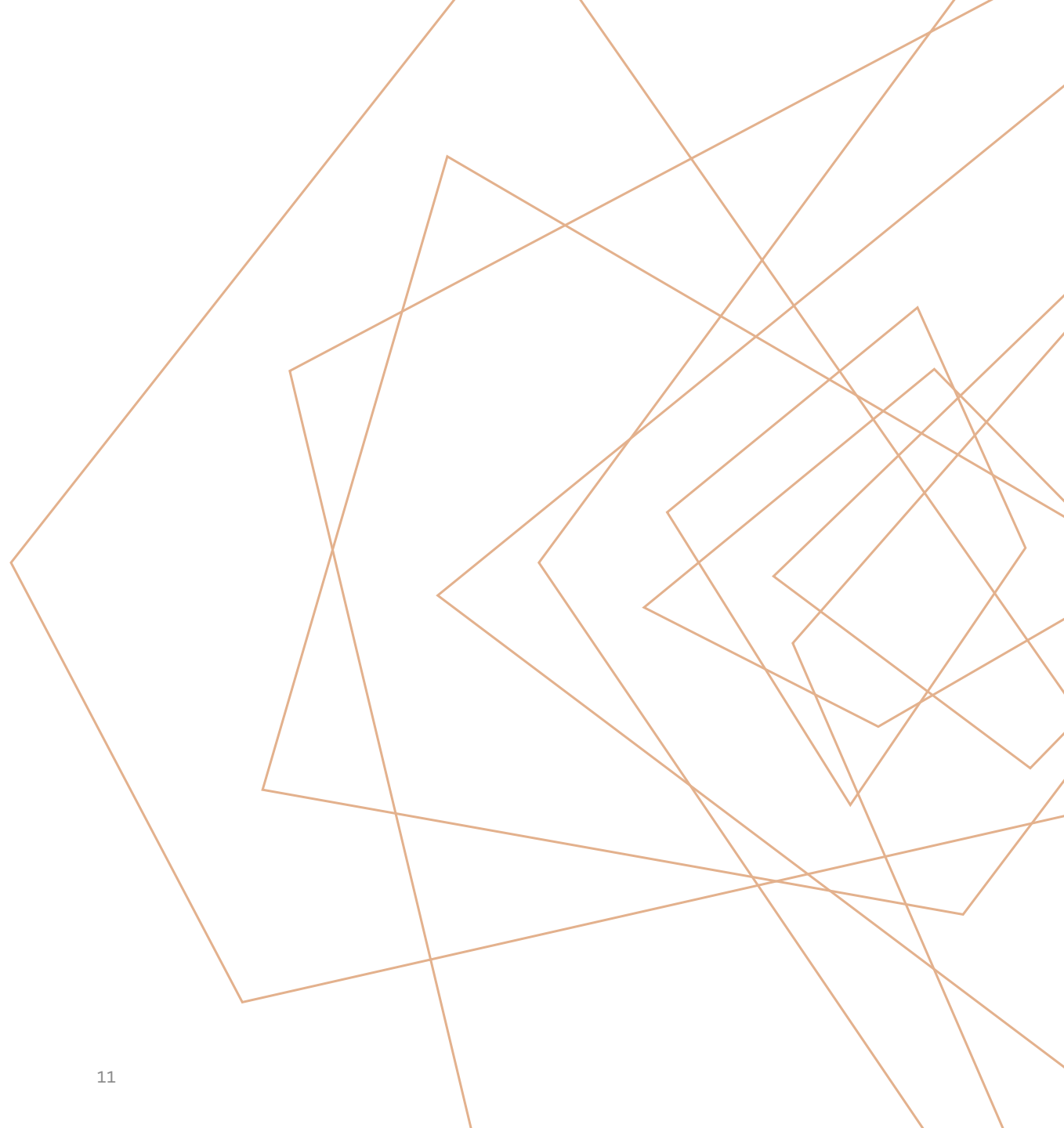
Types of Customizations

The CEMLI framework encompasses five types of customizations that can be made to Oracle applications. These types are:

The CEMLI framework provides a structured approach to managing customizations to Oracle applications. By following the framework's guidelines, organizations can ensure that their customizations are aligned with their business requirements and are appropriately integrated with the core application.

The CEMLI framework
- **C**onfiguration
- **E**xtension
- **M**odification
- **L**ocalization
- **I**ntegration

TYPES OF
CUSTOMIZATIONS

Configuration

Configuration in the CEMLI Framework refers to adjusting an Oracle application's settings and parameters to align with an organization's business requirements. This type of customization typically involves setting up user roles and permissions, defining approval workflows, and configuring accounting rules. It focuses on modifying an application's setup or configuration settings to meet specific business needs, such as changing default language settings or customizing the user interface. Using the Configuration approach, organizations can tailor the application to their specific requirements while minimizing the risk of disrupting the underlying system's stability and integrity.

TYPES OF
CUSTOMIZATIONS

Configuration | Examples

- Setting up a new responsibility in Oracle E-Business Suite
- Configuring a new printer for printing invoices in Oracle Fusion Applications
- Modifying the default currency in Oracle Cloud ERP
- Setting up a new employee in Oracle HRMS with a new job and position.
- Defining a new organization in Oracle Inventory Management to manage inventory at a new location.
- Configuring a new payment term in Oracle Payables to support payment for a new vendor.

TYPES OF
CUSTOMIZATIONS

Extension

The extension part of the CEMLI framework involves adding new features or functions to an Oracle application to meet an organization's specific business requirements. This typically involves developing new code by modifying or creating new code from scratch.

Extensions can be implemented using various development tools like Oracle Application Framework (OAF), Oracle Forms, or Oracle Reports. Extensions can add new functionality to an application, such as custom workflows, new reports, or additional fields to capture data.

Extensions made using the CEMLI framework are designed to be maintainable and upgradeable, meaning that they can be easily modified or updated when new releases of the Oracle application become available. This ensures that customizations do not become obsolete and can continue to provide value to the organization over time.

TYPES OF
CUSTOMIZATIONS

Extension | Examples

- Adding a custom report to Oracle E-Business Suite to display order information in a specific format
- Developing a custom web page in Oracle Application Express to allow customers to submit service requests
- Creating a custom workflow in Oracle Cloud HCM to automate the onboarding process for new employees
- Adding a new custom report in Oracle General Ledger to display financial information in a specific format.
- Developing a custom form in Oracle Forms to capture additional information about customer orders.
- Creating a custom workflow in Oracle Workflow to automate the approval process for purchase requisitions.

TYPES OF
CUSTOMIZATIONS

Modification

Modification is altering or customizing an existing Oracle application's functionality to meet an organization's specific requirements. It involves changing the application's underlying code, database structure, or user interface. Modifications may include adding new fields to a form, creating new reports, or changing the behavior of existing functions. It is important to note that modifications may impact the application's future upgrades, making it essential to carefully evaluate the impact of any changes before implementing them.

TYPES OF
CUSTOMIZATIONS

Modification | Examples

- Modifying the standard invoice validation process in Oracle E-Business Suite to include additional checks.
- Changing the default inventory valuation method in Oracle Fusion Applications to align with specific accounting rules.
- Adding custom code to Oracle Cloud SCM to automate order fulfillment.
- Changing the default invoice numbering sequence in Oracle Receivables to meet specific business requirements.
- Modifying the standard sales order validation process in Oracle Order Management to include additional checks.
- Adding custom code to Oracle Purchasing to automate the vendor selection process.

TYPES OF
CUSTOMIZATIONS

Localization

The Localization aspect of the CEMLI framework involves customizing an Oracle application to meet specific local or regional requirements. This may include adding support for local languages or currencies, complying with local tax regulations, and adapting the application to local business practices. By localizing an Oracle application, organizations can ensure that the software meets their specific needs and that their employees can use it effectively, regardless of location.

TYPES OF
CUSTOMIZATIONS

Localization | Examples

- Configuring Oracle E-Business Suite to support local tax laws in India.
- Modifying Oracle Cloud HCM to display dates in the DD/MM/YYYY format for a European customer.
- Translating the user interface of Oracle Fusion Applications into Spanish for a Latin American market

TYPES OF
CUSTOMIZATIONS

Integration

The Integration part of the CEMLI framework involves connecting an Oracle application with other software systems to enable data exchange and process automation. This may include integrating with third-party software, legacy systems, or other Oracle applications.

Integration customizations can range from simple data transfers between systems to complex workflows involving multiple applications. This may involve developing interfaces using standard technologies such as web services or creating custom code to integrate with specific systems.

TYPES OF
CUSTOMIZATIONS

Integration

The goal of integration customizations is to improve the efficiency of business processes by automating data exchange and eliminating manual data entry. It can also improve data accuracy and consistency by enabling real-time updates between systems.

TYPES OF
CUSTOMIZATIONS
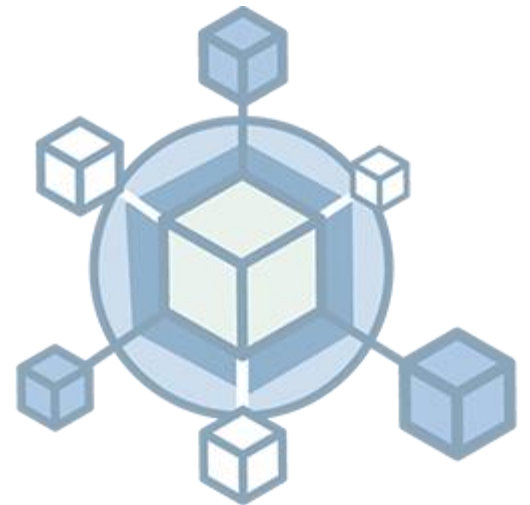
Integration | Examples

- Integrating Oracle E-Business Suite with a third-party CRM system to share customer data
- Connecting Oracle Cloud ERP with an external payment gateway for processing credit card payments
- Integrating Oracle Cloud HCM with a background check provider to automate employee verification.
- Integrating Oracle E-Business Suite with a third-party shipping carrier to automate shipping.
- Connecting Oracle Purchasing with an external vendor management system to share vendor data.

TYPES OF
CUSTOMIZATIONS

Conclusion

Each of these customizations requires a different approach and skill set. The CEMLI framework provides guidelines for managing each type of customization while maintaining consistency and repeatability across the application. By following the CEMLI framework, organizations can make necessary changes to their Oracle applications without disrupting the underlying system's stability and integrity.

TYPES OF
CUSTOMIZATIONS

CEMLI Complexity
Guideline

The CEMLI Complexity Guideline

The Complexity Guideline is a key aspect of the Oracle CEMLI framework, which defines a set of criteria for classifying customizations based on their complexity. The guideline helps organizations assess the effort required to implement customization and plan for the necessary resources and budget accordingly.

The Complexity Guideline defines three levels of complexity: **Simple**, **Moderate**, and **Complex**.

CEMLI COMPLEXITY

Simple

Customizations that involve minimal changes to Oracle standard functionality do not require significant development effort and can be easily implemented using standard configuration tools.

Example: -
- Adding a new responsibility to a user profile
- Enabling a new feature or module that is not yet in use
- Changing a message or prompt on a form
- Adding a new value to a list of values (LOV)
- Changing a descriptive flex-field label or value set

CEMLI COMPLEXITY

Moderate

Customizations that require some level of programming effort and changes to Oracle standard functionality but do not significantly alter the underlying application's behavior or data model.

Example: -

- Creating a new custom concurrent program
- Developing a custom workflow process
- Modifying a seeded report to meet specific business needs.
- Integrating with a third-party system using web services
- Developing a new custom form with moderate complexity

CEMLI COMPLEXITY

Complex

Customizations that involve significant changes to Oracle standard functionality and require extensive programming effort, potentially impacting the application's behavior, data model, or performance.

Example: -

- Modifying core application functionality, such as changing the way the system calculates tax
- Developing a new custom module or application
- Upgrading a heavily customized system to a new release of Oracle E-Business Suite
- Customizing the application's data model, such as adding new tables or modifying existing ones
- Developing complex integrations with multiple systems or complex business rules

CEMLI COMPLEXITY

Conclusion

Using the Complexity Guideline, organizations can better plan and prioritize customizations, allocate resources appropriately, and manage the associated risks and costs. It also helps organizations maintain the stability and reliability of the Oracle E-Business Suite while ensuring that customizations meet business requirements.

CEMLI COMPLEXITY

CEMLI Benefits

Benefits of the CEMLI Framework

The CEMLI framework provides a structured approach to customizing Oracle applications to meet an organization's business requirements. By following this framework, organizations can benefit in several ways:

- Reduced risk of system disruptions
- Easier upgrades
- Better alignment with business requirements
- Improved collaboration and communication

CEMLI BENEFITS

Reduced risk of system disruptions

The CEMLI framework provides guidelines for customizing Oracle applications, documenting customizations, and testing before deployment. By following the CEMLI framework, organizations can reduce the risk of system disruptions caused by untested customizations.

CEMLI BENEFITS

Easier upgrades

The CEMLI framework provides a standard approach to customizing Oracle applications that is consistent and repeatable. This consistency and repeatability make it easier for organizations to upgrade their Oracle applications. They can quickly identify and migrate customizations to the new version of the application, reducing the time and effort required to upgrade.

CEMLI BENEFITS

Better alignment with business requirements

The CEMLI framework provides guidelines for customizing Oracle applications to meet an organization's specific needs while maintaining the core application's integrity and stability. This alignment ensures that the customized Oracle applications meet the business requirements and deliver the expected value to the organization.

CEMLI BENEFITS

Improved collaboration and communication

The CEMLI framework provides a common language and structure for discussing and documenting customizations. This common language and structure improve collaboration and communication between the technical and business teams. They can better understand the customizations and their impact on the system.

CEMLI BENEFITS

Conclusion

Overall, **the CEMLI framework** enables organizations to effectively manage customizations to their Oracle applications. By providing a structured approach to customizing and documenting customizations, organizations can reduce the risk of system disruptions, make upgrades more effortless, better align their Oracle applications with business requirements, and improve collaboration and communication between technical and business teams.

CEMLI BENEFITS

Applying **CEMLI**

Applying the CEMLI Framework

The CEMLI framework provides a structured approach to customizing Oracle applications, but its successful implementation requires careful planning and execution. Here are the steps to applying the CEMLI framework:
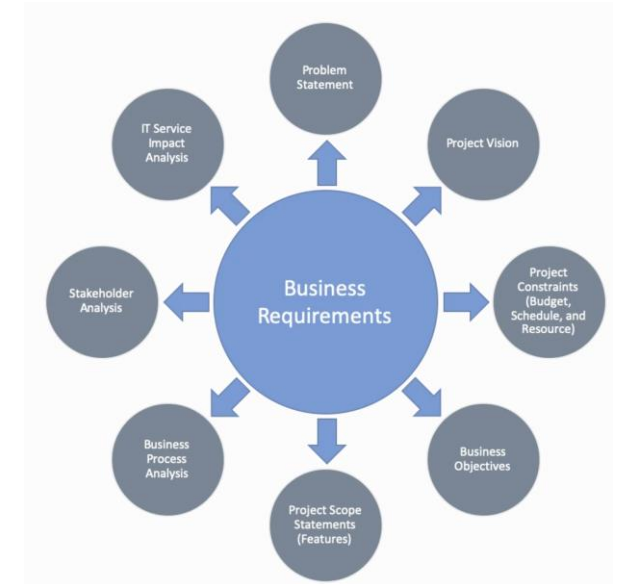
- Assess business requirements
- Determine customization type
- Assess complexity level
- Develop a customization plan
- Maintain and upgrade
- Evaluate effectiveness

APPLYING **CEMLI**

Assess business requirements

The first step is to assess the Oracle application's business requirements. This assessment should include a review of the current system's strengths and weaknesses, identifying gaps between the design and business requirements, and identifying areas where customizations are required.
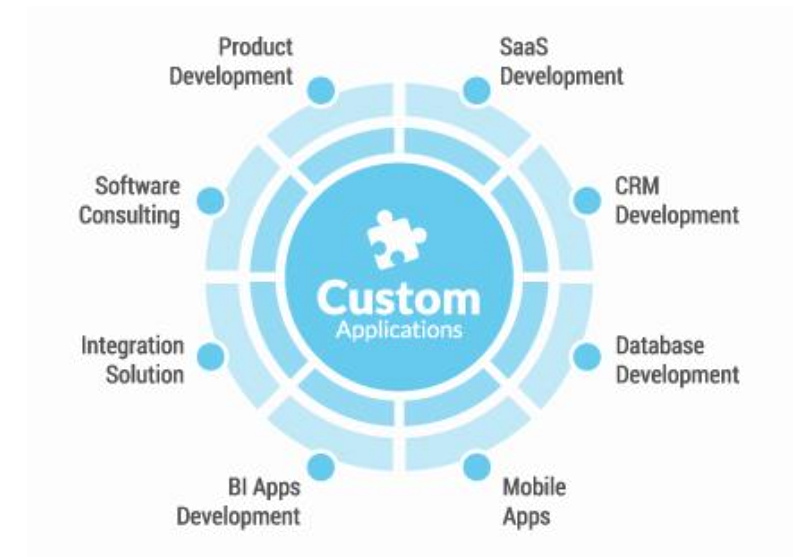

APPLYING **CEMLI**

Determine customization type

Once the business requirements have been assessed, determine which customization types are necessary. For example, these could be **Configuration**, **Extension**, **Modification**, **Localization**, and **Integration**.
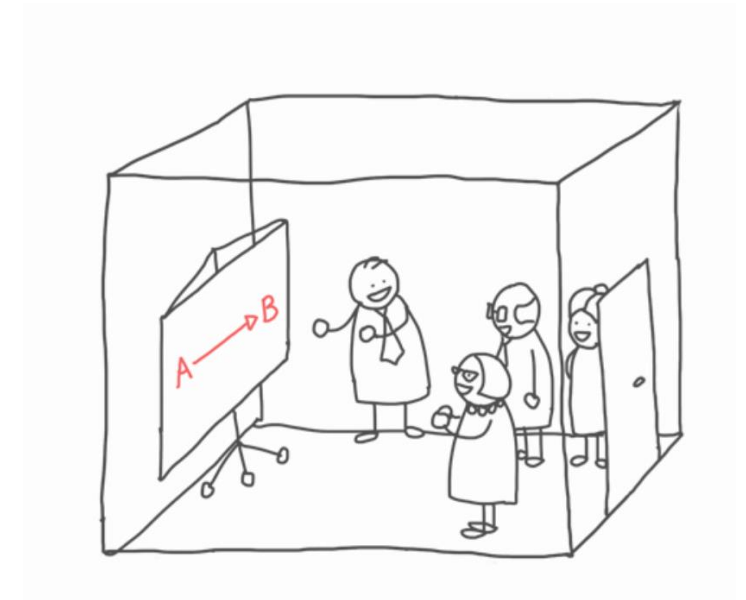

APPLYING **CEMLI**

Assess complexity level

Determine the complexity level of the customization. The complexity level could be **simple**, **moderate**, or **complex**.

APPLYING **CEMLI**

Develop a customization plan

Develop a customization plan including the details, such as the scope, objectives, schedule, resources, and budget.

APPLYING **CEMLI**

Implement customization

Implement the customization according to the plan. This implementation should include proper documentation, testing, and training.

APPLYING **CEMLI**

Maintain and upgrade

After implementing the customization, maintain it, and upgrade it as necessary. This step includes monitoring and managing the customizations to ensure they remain aligned with business requirements and compatible with future upgrades.

APPLYING **CEMLI**

Evaluate effectiveness

Evaluate the effectiveness of the customization in meeting business requirements. This evaluation should include a review of the customization's impact on the system's performance, stability, and security.

APPLYING **CEMLI**

Conclusion

By following these steps, organizations can effectively apply the **CEMLI framework** to customize Oracle applications to meet their business requirements. The key to successful implementation is careful planning, proper documentation, and ongoing maintenance and evaluation.

APPLYING **CEMLI**

Applying Guidelines

Applying Guidelines

Customizing Oracle applications using the CEMLI framework requires following specific guidelines to ensure that customizations are implemented correctly and do not cause disruptions to the core application. Here are some guidelines to consider:

- Documentation
- Testing
- Version control
- Adherence to coding standards
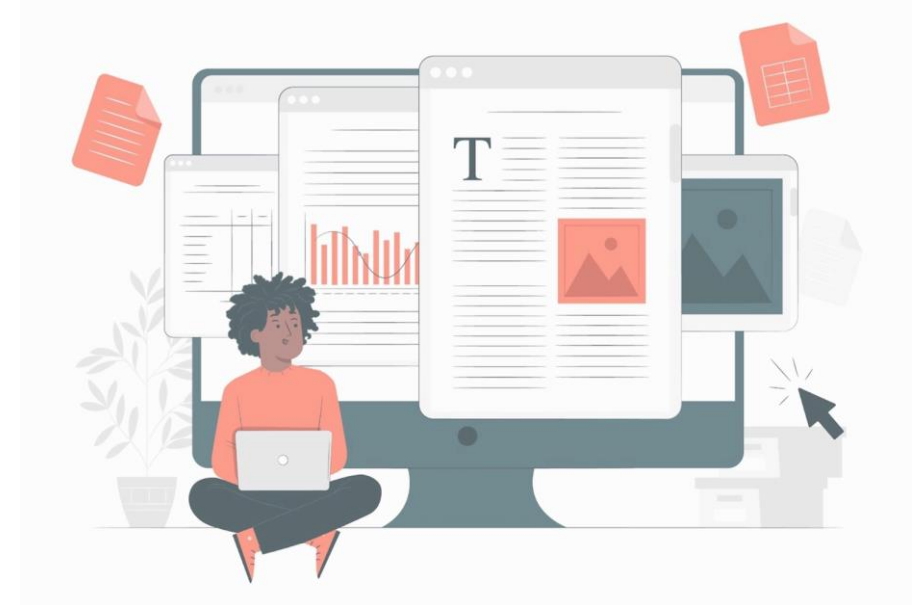- Impact analysis
- Change Management

APPLYING GUIDELINES

Documentation

Proper documentation is essential for tracking customizations and ensuring they can be easily managed and maintained. Documentation should include a detailed description of the customization, its objectives, scope, and impact on the system.
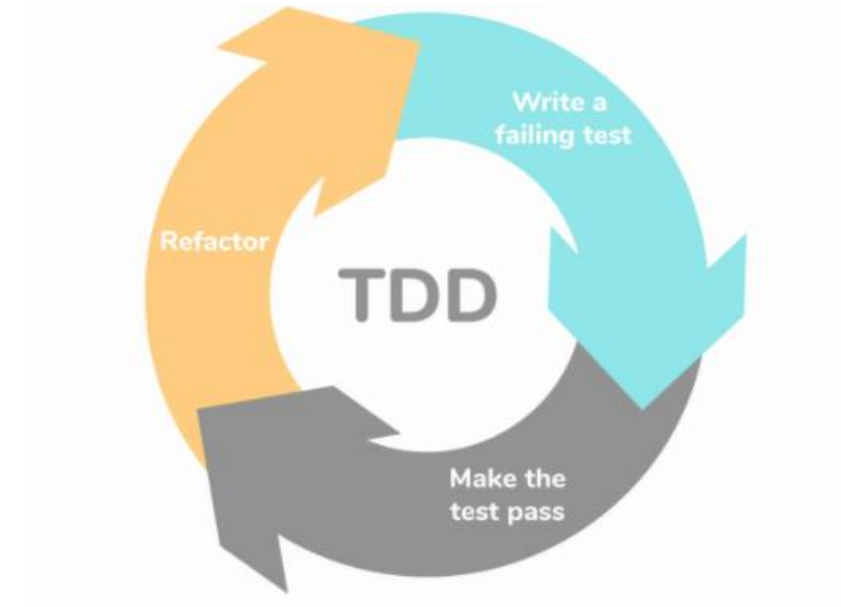
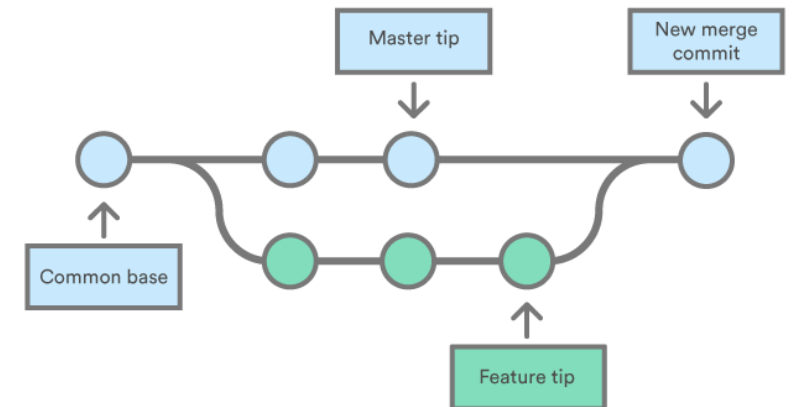APPLYING GUIDELINES

Testing

Testing ensures that customizations work as expected and do not negatively impact system performance. It is essential to perform thorough testing before deploying customizations to the production environment.

APPLYING GUIDELINES

Version Control

Version control ensures that customizations are accurately tracked, managed, and updated over time. Proper version control procedures should be implemented to ensure that changes are appropriately tracked, reviewed, and approved.
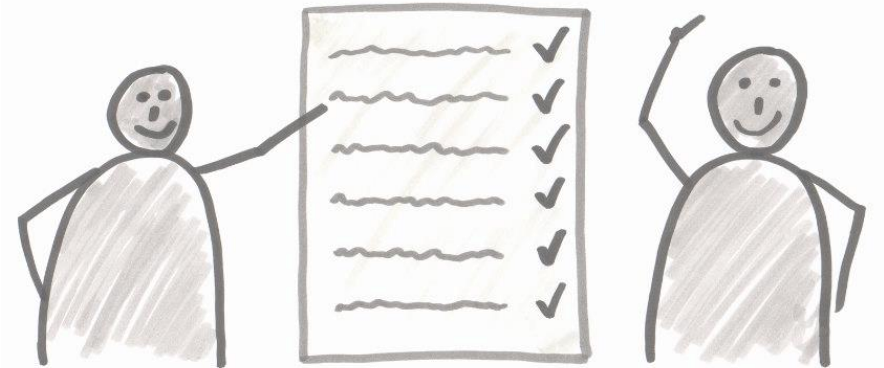

APPLYING GUIDELINES

Adherence to coding standards

Adherence to coding standards ensures that customizations are consistent and follow best practices, making it easier to maintain and update customizations in the future.

APPLYING GUIDELINES

Impact analysis

Conduct an impact analysis to identify the potential impact of customizations on the system, including any dependencies or integrations.

APPLYING GUIDELINES

Change Management

Implement proper change management procedures to ensure that customizations are reviewed, approved, and tested before deployment to the production environment.

APPLYING GUIDELINES

Conclusion

By following these guidelines, organizations can reduce the risk of system disruptions and ensure that customizations are implemented correctly, aligned with business requirements, and maintainable over time.

APPLYING GUIDELINES

Conclusion

#demasy

CEMLI Conclusion

The CEMLI framework is a structured approach to customizing Oracle applications, consisting of five types of customizations: **C**onfiguration, **E**xtension, **M**odification, **L**ocalization, and **I**ntegration. Organizations can apply the CEMLI framework to tailor Oracle applications to their needs while maintaining the core application's integrity and stability.

The CEMLI framework's benefits include reducing the risk of system disruptions, making upgrades more straightforward, and better aligning Oracle applications with business requirements. Organizations can also benefit from increased consistency and repeatability in customizations, allowing them to make necessary changes without disrupting the underlying systems.

CONCLUSION

CEMLI Conclusion

To apply the **CEMLI framework**, organizations should follow guidelines for each customization type, document customizations, and test them thoroughly before deployment. Best practices include prioritizing configurations over modifications, minimizing the use of custom code, and maintaining compatibility with future upgrades.

Challenges in applying the **CEMLI framework** include determining the appropriate complexity level for customizations, ensuring compatibility with third-party applications, and managing the customizations over time. However, by following best practices and leveraging the benefits of the CEMLI framework, organizations can achieve a tailored Oracle application that meets their unique business requirements while maintaining its stability and integrity.

CONCLUSION

Thank You!